VAAL UNIVERSITY OF TECHNOLOGY

Faculty of Applied and Computer Sciences

AN INNOVATIVE INTERNET OF THINGS SOLUTION TO CONTROL REAL-LIFE AUTONOMOUS VEHICLES

Roger L Wahl

Doctor Philosophy Information and CommunicationTechnology

20423985

Research project and thesis (AIRPP6A) submitted in complete fulfilment of the requirements for the Doctor Technologiae in Information Technology degree in the Department of Information and Communication Technology at the Vaal University of Technology

Promoter: Prof A Jordaan

Co-Promoter: Dr A Joubert

June 2021

DECLARATION

I, Roger Wahl, declare that the contents of this thesis represent my own unaided work, and that the thesis has not previously been submitted for academic examination towards any qualification. Furthermore, the contents of this thesis represent my own opinions and not necessarily those of the Vaal University of Technology.

Signed **Date**

ACKNOWLEDGEMENTS

Appreciation is extended to the tenacious autonomous automotive industry and researchers in the field on whose shoulders I have stood.

Furthermore, I wish to express sincere gratitude to the following persons whose assistance was instrumental in my completion of this document:

- Professor Annelie Jordaan for her unwavering encouragement and guidance
- Dr André Joubert for his knowledge and supervision
- Staff from the Faculty of Computer Science of the Vaal University of Computer Science in sanctioning this research project
- My family for their love and support

ABSTRACT

This research was initiated because of a global increase in congestion on roads and the consequent increase in the rate of fatalities on both national and international roads. Annually, 1.3 million people are killed on the roads globally, and millions are injured. It was estimated that 2.4 million people will be killed in road traffic accidents annually by 2030, and in South Africa, over 14 000 deaths were reported in 2016. A study undertaken by the American Automobile Association Foundation for Traffic Safety (AAAFTS), established in 1947 to conduct research and address growing highway safety issues, found that motorcar accidents, on average, cost the United States $300 billion per annum. In the same vain, the World Health Organisation (WHO) asserted in their 2013 Global Status Safety Report on Road Safety that by 2020, traffic accidents would become the third leading cause of death globally. In this organisation's 2015 report, South Africa was listed as having one of the highest road fatality rates in the world, averaging 27 out of 100 000 people.

Cognisance of these statistics that describe wanton loss of life and serious economic implications, among other reasons, led to the development of autonomous vehicles (AVs), such as Google and Uber's driverless taxis and Tesla's autonomous vehicle. Companies have invested in self-driving prototypes, and they bolster this investment with continuous research to rectify imperfections in the technologies and to enable the implementation of AVs on conventional roads. This research aimed to address issues surrounding the systems communication concept, and focused on a novel method of the routing facet of AVs by exploring the mechanisms of the virtual system of packet switching and by applying these same principles to *route* autonomous vehicles. This implies that automated vehicles depart from a source address and arrive at a pre-determined destination address in a manner analogous to packet switching technology in computer networking, where a data packet is allotted a source and destination address as it traverses the Open Systems Interconnection (OSI) model for open system interconnection prior to dissemination through the network.

This research aimed to develop an IoT model that reduces road congestion by means of a cost effective and reliable method of routing AVs and lessen dependency on vehicle-to-vehicle (V2V) communication with their heavy and costly sensor equipment and GPS, all of which under certain conditions malfunction. At the same time, as safety remains the foremost concern, the concept aimed to reduce the human factor to a considerable degree. The researcher demonstrated this by designing a computer-simulated Internet of Things (IoT) model of the concept.

Experimental research in the form of a computer simulation was adopted as the most appropriate research approach. A prototype was developed containing the algorithms that simulated the theoretical model of IoT vehicular technology. The merits of the constructed prototype were analysed and discussed, and the results obtained from the implementation exercise were shared. Analysis was conducted to verify arguments on assumptions to clarify the theory, and the outcome of the research (an IoT model encompassing vehicular wireless technologies) shows how the basic concept of packet switching can be assimilated as an effective mechanism to route large-scale autonomous vehicles within the IoT milieu, culminating in an effective commuter operating system.

Controlled routing will invariably save the traveller time, provide independence to those who cannot drive, and decrease the greenhouse effect, whilst the packet switching characteristic offers greater overall security. In addition, the implications of this research will require a workforce to supplement new growth opportunities.

Keywords: Internet of Things (IoT), computer simulation, machine learning (ML), artificial intelligence (AI), LiDAR, packet switching, routers

PUBLICATIONS

The conference abstract and poster indicated below were prepared and presented as progress was made in the preparation and completion of this thesis. All the work was conducted by Roger Wahl under supervision of Prof Annelie Jordaan and Dr André Joubert.

Wahl, R., Jordaan, A. & Joubert, A. (2020). Digitalisation for self-regulated vehicles via the Internet of Things. *Poster presentation*. NEMISA Digital Skills Summit and Research Colloquium, 11-13 March 2020, Birchwoord, Johannesburg (Annexure D).

Wahl, R., Jordaan, A. & Joubert, A. (2020). Digitalisation for self-regulated vehicles via the Internet of Things. *Proceedings.* NEMISA Digital Skills Summit and Research Colloquium, 11-13 March 2020, Birchwoord, Johannesburg (Annexure E).

The following publication of the researcher was used as the basis for conducting this research.

Wahl, R. L. (2013). *The study of packet switched technology incorporated into large-scale magnetic levitating vehicles.* MTech Dissertation, Vaal University of Technology, South Africa.

TABLE OF CONTENTS

DECLARATION I
ACKNOWLEDGEMENTS III
ABSTRACT IV
PUBLICATIONS VI
LIST OF FIGURES XII
LIST OF TABLES XIV
GLOSSARY OF TERMS XV
PROLOGUE 1

CHAPTER 1: INTRODUCTION 11
1.1 Introduction 11
1.2 Keywords 12
1.3 Background to the research 12
1.3.1 Internet of Things 14
1.3.2 Rationale and motivation 17
1.4 Problem statement 17
1.5 Research aim 18
1.6 Research questions 18
1.6.1 Primary research question (PRQ) 18
1.6.2 Secondary research questions (SRQs) 18
1.7 Research objectives 18
1.7.1 Primary research objective (PRO) 18
1.7.2 Secondary research objectives (SROs) 19
1.8 Research design and methodology 19
1.8.1 Strategy of enquiry and philosophy 19
1.8.2 Methodology 20
1.8.2.1 Simulation 23
1.9 Validity and reliability of data 25
1.9.1 Validity 25
1.9.2 Reliability of data 26
1.10 Delimitations 27
1.11 Ethics 27
1.12 Contribution of the research 28
1.13 Chapter outline 29
1.14 Summary 29

CHAPTER 2: LITERATURE REVIEW ... 30
2.1 Background ... 30
2.2 Technologies encompassing packet switching ... 34
2.2.1 Packet switching mechanisms used by routers ... 39
2.2.1.1 Process switching/software switching ... 39
2.2.1.2 Interrupt context switching ... 40
2.2.1.3 Network load balancing ... 43
2.2.2 Router operating stages ... 44
2.2.3 Routing mechanisms ... 45
2.2.4 Routing protocols ... 46
2.2.4.1 Distance vector routing ... 46
2.2.4.2 Link State Routing Protocol ... 48
2.2.4.3 Summary of differences between distance vector and link-state protocols ... 50
2.2.4.4 Internet Protocol Version 6 (IPv6) ... 51
2.2.4.5 5G communications ... 51
2.3 Evolution of Artificial Intelligence (AI) and the Internet of Things (IoT) ... 51
2.3.1 Artificial Intelligence (AI) ... 52
2.3.2 Machine learning (ML) ... 54
2.3.2.1 Light Detection and Ranging (LiDAR) ... 56
2.3.2.2 VHF Omni-directional Radio Range (VOR) ... 60
2.3.3 Internet of Things (IoT) ... 60
2.3.3.1 Cloud Computing ... 64
2.3.3.2 IoT security ... 65
2.4 The impact of AI and IoT on society ... 65
2.5 The potential impact of self-driving vehicles on society ... 67
2.6 Previous studies conducted on routing mechanisms ... 69
2.6.1 Vehicular Ad Hoc Networks (VANETs) ... 69
2.6.2 Other research on routing mechanisms ... 71
2.7 Theory defining the study ... 78
2.7.1 Design theory ... 78
2.7.1.1 Seminal work ... 78
2.7.1.2 Definition ... 79
2.7.1.3 Characteristics ... 79
2.7.1.4 Advantages ... 79
2.7.1.5 Disadvantages ... 79
2.7.1.6 Extent of application to this study ... 79
2.7.1.7 Limitation of theory relative to this study ... 80
2.7.2 Systems theory ... 80

2.7.2.1 Seminal work 80
2.7.2.2 Definition 81
2.7.2.3 Characteristics 81
2.7.2.4 Advantages 82
2.7.2.5 Disadvantages 82
2.7.2.6 Extent of application to this study 82
2.7.2.7 Limitation of theory relative to this study 83
2.8 Summary 83

CHAPTER 3: RESEARCH DESIGN AND METHODOLOGY 85
3.1 Introduction 85
3.2 Real-system problem defined 85
3.3 Build a model 86
3.3.1 Step 1: Deciding on static, discrete, continuous or combined discrete-continuous modelling 87
3.3.2 Step 2: Elements that drive the system 91
3.3.3 Step 3: Determine the entities that should represent the system elements 91
3.3.4 Step 4: Determine the level of detail needed to describe the system components 92
3.3.5 Step 5: Determine the graphic requirements of the model 92
3.3.6 Step 6: Identify the areas that utilise special control logic 93
3.3.7 Step 7: Determine how to collect statistics in the model and communicate results 93
3.4 Model system 94
3.5 Conclusion 97

CHAPTER 4: RESEARCH AND DESIGN OUTCOMES 98
4.1 Introduction 98
4.2 Building and setting up a network in Psimulator 99
4.3 Perform simulation 103
4.3.1 Destination/location component 104
4.3.2 House component 105
4.3.3 Metropolis component 105
4.3.4 Router component 105
4.3.5 Switch component 106
4.3.6 Link component 106
4.3.7 Observation 106
4.3.7.1 Broadcast domain 106
4.3.7.2 Timestamp 107
4.3.7.3 Security 107

4.4 Simulation results ... 107
4.4.1 Routing outcomes of experiment and events ... 107
4.4.1.1 Recording events ... 109
4.4.1.2 Component events ... 111
4.4.1.3 Event sequence ... 111
4.4.1.4 Component identifiers ... 111
4.4.1.5 Timestamp ... 112
4.5.1 Failed case study ... 115
4.5.2 Replication results ... 115
4.5.3 Failed case observation ... 116
4.6 Construct theories ... 117
4.7 Theoretical predictions ... 118
4.8 Analysis ... 119
4.9 Develop prototype ... 120
4.10 Validation and verification ... 120
4.11 Conclusion ... 122

CHAPTER 5: MODEL INTERPRETATION ... 123
5.1 Introduction ... 123
5.2 Theoretical background to model ... 123
5.3 Outline of simulation objective ... 124
5.4 Overview ... 128
5.5 Conclusion ... 128

CHAPTER 6: FINDINGS, RECOMMENDATIONS AND CONCLUSION ... 129
6.1 Introduction ... 129
6.2 Overview of thesis ... 129
6.3 Findings ... 130
6.3.1 Findings of Secondary Research Objective 1 ... 130
6.3.2 Findings of Secondary Research Objective 2 ... 132
6.3.3 Findings of Secondary Research Objective 3 ... 133
6.3.4 Findings of Secondary Research Objective 4 ... 134
6.3.5 Findings of Secondary Research Objective 5 ... 135
6.3.6 Findings of Secondary Research Objective 6 ... 136
6.3.7 Findings of Primary Research Objective ... 137
6.4 Contribution ... 137
6.5 Recommendations for further research ... 138
6.6 Conclusion ... 138

REFERENCE LIST **140**
ANNEXURE A: CONSOLE LOGS OF ROUTER EVENTS **158**
ANNEXURE B: COMPREHENSIVE LOGS OF CAPTURED EVENTS **165**
ANNEXURE C: COMPREHENSIVE LOGS OF REPLICATION TESTING **172**
ANNEXURE D: CONFERENCE PROCEEDINGS **176**
ANNEXURE E: CONFERENCE POSTER **179**

LIST OF FIGURES

Figure 1.1: Model-prompting trend for the IoT 15
Figure 1.2: Process of constructing the proposed computer model 22
Figure 1.3: Basic components and function of Psimulator 23
Figure 1.4: Drill down on proposed simulation process 25
Figure 1.5: Conceptual and operational validation and verification in simulation 26
Figure 2.1: Graphical representation of South African road fatalities 2008 to 2017 31
Figure 2.2: Demographic breakdown by the Centre for Automotive Research 33
Figure 2.3: Datagram connectionless packet switching 36
Figure 2.4: Virtual circuit connection-oriented packet switching 36
Figure 2.5: Ports on hub or switch in the same broadcast domain and ports on router 37
Figure 2.6: Virtual circuit packet network 39
Figure 2.7: Routing table for virtual-circuit packet switching network 39
Figure 2.8: Process switching flow 40
Figure 2.9: Interrupt context switching 41
Figure 2.10: Structure of a binary tree format 41
Figure 2.11: Optimum switching multiway tree 42
Figure 2.12: CEF switching flow 43
Figure 2.13: Algorithm and relative figure for load balancing approach 44
Figure 2.14: System services 45
Figure 2.15: Demonstration of RIP functioning 47
Figure 2.16: Network topology of successor route and feasible 48
Figure 2.17: OSPF area function 49
Figure 2.18: Link state function of OSPF 50
Figure 2.19: Perceptron (neuron) 54
Figure 2.20: Different sensors used in an autonomous vehicle 57
Figure 2.21: Principle of LiDAR distance measurement 58
Figure 2.22: Gartner's 2017 Hype Cycle for emerging technologies 62
Figure 2.23: Design architecture of cyber-physical systems-enabled manufacturing system 63
Figure 2.24: ECUs exchange in-vehicle messages and control the SIDVN 76
Figure 2.25: Node segments 77
Figure 3.1: Preliminary process (highlighted) of constructing proposed computer model 86
Figure 3.2: Network diagram of basic nodes of Psimulator2 system 94
Figure 4.1: Second stage (highlighted) of proposed computer model 98
Figure 4.2: Unmodified Psimulator network topology from available simulator routing components 100

Figure 4.3: Interface from simulation with configured node IP address details depicting UI properties ... 101
Figure 4.4: Sector of console window (Figure 4.2) of Cisco router from available simulator routing components displaying routing table info and interface status ... 101
Figure 4.5: Start-up screen of Psimulator network emulator for commencing simulation experiment ... 102
Figure 4.6: Accessible animated envelopes for simulation indicating colour-coded data transmitted ... 103
Figure 4.7: Psimulator customised legend for illustrating the experiment in this research ... 104
Figure 4.8: Properties configuration user interface from the experimental simulation ... 105
Figure 4.9: Experiment network topology portrayed during simulation ... 108
Figure 4.10: Routing simulation of autonomous vehicle ... 109
Figure 4.11: Graphical log of events in simulated routing of autonomous vehicles ... 114
Figure 4.12: Replication test sampling ... 114
Figure 5.1: Diagram representing a concept consolidation model ... 124
Figure 5.2: Platform and Design constituents of the consolidation model ... 126
Figure 5.3: Configuration and Implementation constituents of the consolidation model ... 127
Figure 5.4: Packet Switching constituent of the consolidation model ... 127

LIST OF TABLES

Table 1.1: Outline of chapters 29
Table 2.1: Comparative hierarchical functions of seven-layer Open Systems Interconnection (OSI) model and four-layer Transmission Control Protocol model 35
Table 2.2: Summary of differences between distance vector and link state protocols 50
Table 2.3: Merging IT and OT under differing technical standards 66
Table 3.1: Tools representing system elements 96
Table 4.1: Console logs of captured events 110
Table 4.2: Extract of log created by the emulator used for replaying the simulator 112
Table 4.3: Output data for case study 2 of the experiment 115
Table 4.4: Failed case study depicting unreachable host/destination 116

GLOSSARY OF TERMS

Abbreviation	Full Term
ABS	Antilock Braking System
ABR	Area Border Router
AD	Advertised Distance
A-EV(s)	Electric Autonomous Vehicles
AGPS	Assist Global Positioning System
AI	Artificial Intelligence
AIB	Adjacency Information Base
ANN	Artificial Neural Network
AODV	Ad hoc on Demand Vector
ARP	Address Resolution Protocol
AS	Autonomous System
ASBR	Autonomous System Border Router
ASIC	Application-Specific Integrated Circuit
AV	Autonomous Vehicle
BGP	Border Gateway Protocol
CACC	Cooperative Adaptive Cruise Controller
CAN	Controller Area Network
CEF	Cisco Express Forwarding
CO2	Carbon Dioxide
CPU	Central Power Unit
DES	Discrete Event Simulation
DNS	Domain Naming Server
DOT	States Departments of Transportation
DSDV	Destination-Sequencing Distance Vector
DSRC	Dedicated Short Range Communication
ECU	Electrical Control Unit
EGP	Exterior Gateway Protocols
EHCV	Extra Heavy Commercial Vehicles
EIGRP	Enhanced Interior Gateway Routing Protocol
EIoT	Enterprise Internet of Things
FCC	Federal Communications Commission
FD	Feasible Distance
FIB	Forwarding Information Base

Abbreviation	Full Term
GIS	Geographic Information Systems
GPS	Global Positioning System
GUI	Graphical User Interface
HGV(s)	Heavy Goods Vehicles
ICMP	Internet Control Message Protocol
IBM	International Business Machines
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGP	Interior Gateway Protocol
IOS	Internet Operating System
IOS XR	Internet Operating System train of Cisco systems employed
IIoT	Industrial Internet of Things
IoT	Internet of Things
IoV	Internet of Vehicles
IP	Internet Protocol
IPv4	Internet Protocol version 4
IS	Information Systems
ISDT	Information Systems Design Theory
IS-IS	Intermediate System-Intermediate System
IT	Information Technology
ITS	Intelligent Transportation Systems
LAN	Local Area Network
LiDAR	Light Detection and Radar Ranging
LIVN(s)	Legacy In-vehicle Network(s)
LSA(s)	Link-State Advertisement(s)
LSP	Link State Packet
LSR	Link State Request
LSU(s)	Link-State Update(s)
MAC	Media Access Control
ML	Machine Learning
MLP	Multi-Layer Perceptron
NHTSA	National Highway Traffic Safety Administration
NIC	Network Interface Card
NPU	Network Processing Unit
OS	Open Systems

Abbreviation	Full Term
OSI	Open Systems Interconnection
OSPF	Open Shortest Path First
OSS	Open-Source Software
OT	Operations Technology
PERT	Programme Evaluation Review Technique
QoS	Quality of Service
RADAR	Radio Detection and Ranging
RAM	Random Access Memory
RD	Reported Distance
RFID	Radio Frequency Identification
RIB	Routing Information Base (routing table)
RIP	Routing Information Protocol
RTMC	Road Traffic Management Corporation
SDIVN(s)	Software Defined In-Vehicle Network(s)
SDN	Software Defined Networking
SID	Segment ID Identifier
SOA	Service-Oriented Architecture
SPRING	Source Packet Routing in Networking
TaaS	Transport-as-a-Service
TCP	Transmission Control Protocol
TSC	Transport Systems Catapult
TTL	Time-To-Live
UDP	User Datagram Protocol
UI	User Interface
URI	Uniform Resource Identifier
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
VANET(s)	Vehicular Ad Hoc Network(s)
VC	Virtual Circuit
VCI	Virtual Circuit Identifier
VHF	Very High Frequency
VLSM	Variable Length Subnet Mask
VOR	VHF Omni-Directional Radio Range
WAN	Wide Area Network
WAVE	Wireless Access in Vehicular Environment

Abbreviation	Full Term
WSNs	Wireless Sensor Networks
XML	Extendable Mark-up Language

PROLOGUE

"Any skilled engineer can take control remotely of any connected 'thing'. Society has not yet realised the incredible scenarios this capability creates."
– André Kudelski, Chairman and CEO of Kudelski Group

ANECDOTAL SCENARIO

Ethics Review Boards and AI Self-Driving Cars

– Dr Lance Eliot, the AI Trends Insider

"As a driver of a car, you are continually making judgments that involve life-or-death matters. We don't tend to think explicitly about this aspect of driving and take it for granted most of the time. Whenever there is a car accident, the topic comes up about what the driver did or did not do, and any aspects of how judgment came to play in the accident usually comes to light.

Suppose you are driving down a street at nighttime. You have your radio on. It has been a long hard day at work and you are heading home for the evening. How well are you paying attention to the driving task? Perhaps your thoughts are focused on a difficult problem at work that you are hopeful of solving. The radio is meanwhile tuned to a talk show and it covers a topic of keen interest to you.

You normally take the main highway to get home, but tonight you opted to use a less common road that you hope has little traffic and will allow you to get home faster. The speed limit is 45 miles per hour, and you are doing about 55 mph. Going over the speed limit on this particular road happens all the time and going just 10 mph over the speed limit is actually not much of an excess in comparison to what other drivers do.

Suddenly, via your headlight beams, you see what might be a figure in the road up ahead. There's not a crosswalk nearby and so you weren't anticipating that any pedestrians would be in the roadway. You weren't looking for pedestrians, plus with your thoughts on the problems at work and with your somewhat rapt listening to the radio talk show, it all added up that you didn't notice the shadowy figure at first.

Your mind races as to whether it really is a person or not. The roadside lighting in this area is rather poor. You have only a few seconds of time to decide what to do. Should you slam on the brakes? But, if so, there is a car behind you and they might ram into your car. Plus, perhaps by slamming on the brakes you might lose control of the car and not be able to manoeuvre it. You could instead try to swing wide, out of your lane, and do so in a somewhat frantic manner under the belief that the shadowy figure is headed in the other direction. You might just skirt the figure by going to the left, if you can swerve just enough and if the shadowy figure continues to move to the right.

Swinging over into the other lane isn't so easy though. There is another car in that lane. You might cause the other driver to react and they might then swerve into the median. You could maybe try to go to your right, up on the sidewalk, doing so to avoid the figure in the street. But, it is so dark that you aren't sure if there might be anyone on the sidewalk and besides the idea of driving on the sidewalk seems almost crazy, really just a desperate last resort to avoid hitting the figure in the street.

This is a relatively realistic scenario and one that any of us could encounter.

Let's analyse the situation.

The driver is faced with a rather untoward dilemma. There might or might not be a pedestrian in the path of their car. Whatever is in the path, the driver only has a few seconds to decide what it is and what action to take.

If the driver opts to use their brakes, it could lead to the car behind the driver doing a rear-ender and it might injure or kill the human occupants in either or both cars.

If the driver opts to swing into the next lane to the left, it could lead to the car in that lane becoming concerned and possibly veering into the median, which could injure or kill the human occupants, and might careen further into traffic and injuring or killing other humans in nearby cars.

If the driver opts to drive up onto the sidewalk in hopes of avoiding the figure in the street, there might be pedestrians there that could get injured or killed, plus the driver might generally lose control of the car and the driver gets injured or killed too.

If the driver decides to stay the course and continue forward, they will potentially hit the shadowy figure. This might injure or kill the figure, assuming it is a human, and the driver might also get injured or killed in the process of striking the figure.

Is there a proper and precise equation or some form of calculus that we can use to identify what the correct course of action is?

I don't think so.

Suppose you had time to try and develop some kind of calculation, what would it consist of? You might try to find out the ages of the various 'participants' such as the driver of the car, the driver of the car behind the dilemma facing car, the driver of the car in the next lane over, etc. Maybe you could say that the older the driver the more they have lived their lives and so the less they count in terms of whether to be on the one that might take the brunt of the situation. In other words, you might say that moving into the lane to the left is the 'better' option because the driver in that car is the oldest of those involved and thus has already lived their life.

Some would say that your use of age in this manner is outrageous and absolutely wrong. You might instead try to calculate the societal value of each participant, somehow trying to encompass what they do and how they are helping our society. Or, maybe you come up with some other factors to try and weigh the value of their human lives.

You might instead just decide to use probabilities regarding the various actions involved. If the approach of slamming your brakes has a 30% chance of injury or death, while if you swing into the next lane there is a 60% chance of injury or death, perhaps you should go with the brakes option since it has the lower probability of an adverse outcome.

These analytic methods could be handy and yet it seems rather trying that any of us could individually come up with an agreeable set of equations or formulas to cover such circumstances for us and others. As far as we all know, the method used by today's human drivers is the nebulous notion of 'human judgment.' None of us can really say whether our brains do some kind of mathematical calculation, nor can we explain directly why we did something. We can rationalise what we did by offering an explanation, but the explanation itself might have little to do with what really happened inside our heads.

Explanations are provided as a means to try and turn our mental aspects into something that can be elaborated to other people. Usually, our explanations are intended to suggest a logical means of how we arrived at a decision. No one can though definitively say or prove that their mind actually carried out the logical steps offered. Instead, the explanation is a post-reflected aspect that might match to what our minds considered, or it might be a completely concocted aspect.

Suppose the driver in this case decides to go ahead run into the shadowy figure. Did they do so after carefully considering all of the other options?

The driver might after-the-fact claim they considered the various options, but perhaps they did and maybe they did not. It could be that the after-the-fact explanation is an attempt to rationalise what took place. The driver might not want to seem as though they just mindlessly rammed into the shadowy figure, and as such, provide instead an elaborated indication of the other options, which might allude to the notion that the driver tried to find a means to avoid the incident, even though maybe they just froze-up or maybe didn't even notice the shadowy figure beforehand at all.

Ponder for a moment the number of times that each of us as car drivers make these kinds of spur of the moment decisions, doing so in real-time, in order to try and avoid causing some kind of car incident that might injure or kill others. It's not just limited to those occasions when you get into a car accident. You undoubtedly have lots of situations that fortunately don't lead to an accident per se, and yet you had to make some tough decisions anyway.

In this case of the driver, suppose it turns out that the shadowy figure was actually a large tumbleweed that was blowing across the street. If the driver opted to plough ahead and into the tumbleweed, perhaps it led to no car accident. The driver just kept going. Meanwhile, the car behind also kept going, and the car in the lane to the left kept going. None of them are injured or killed. Yet, there was a split second or so when a decision might have been made that could have led to their injury or death. No one would have likely recorded this non-event and no explanation or rationalization was sought or tendered.

I'd like to suggest that with the millions of cars on our roads on a daily basis, we are all involved in millions upon millions of such judgment calls, continually, and those of us in the cars, either as drivers or passengers, are subject to the outcomes of those judgments. So too are the pedestrians nearby to wherever cars are driving.

It is actually a bit staggering that we don't have more car accidents. With this many people and they are all making those millions upon millions of judgments, it is somewhat a miracle that their judgments are good enough and sound enough that we don't experience even more car incidents and more injuries and deaths accordingly.

I hope this doesn't scare you from getting into your car. Also, I hope that this discussion hasn't been overly macabre or ghastly. As I suggested earlier, the reliance on human judgement permeates our car driving and determines life-and-death matters. We don't usually overtly consider this aspect in our daily driving and tend to take it in stride.

What does this have to do with AI self-driving cars?

AI Self-Driving Cars Will Need to Make Life-or-Death Judgements

At the Cybernetic AI Self-Driving Car Institute, we are developing AI software for self-driving cars. One crucial aspect to the AI of self-driving cars is the need for the AI to make 'judgments' about driving situations, ones that involve life-and-death matters.

I've had some AI developers tell me that there isn't a need for the AI to make such judgments. When I ask why the AI does not need to do so, the answer is that the AI won't get itself into such predicaments.

I am flabbergasted that someone could have such a belief. In the scenario that I just described, I would assert that the AI could readily have gotten itself into exactly the same predicament that I had indicated the human driver was involved in.

Some might say that the AI would not be distracted by the radio playing and would not be thinking about problems at work. Okay, let's subtract that entirely from the scenario. Some might say that the self-driving car would not be driving over the speed limit. I'd tend to debate that aspect, but anyway, let's go ahead and assume that the self-driving car was doing the 45-mph speed limit.

We still have the situation of the car approaching the shadowy figure and need to consider the matter of a car behind the self-driving car and the car that is to the left of the self-driving car, all being done in real-time, with just a few seconds to decide, and with the balance of people's lives at stake.

If you were to suggest that the self-driving car would be better able to detect the shadowy figure because the self-driving car has not only cameras but also radar, sonic, and perhaps LIDAR capabilities, I'd say that yes there is a chance of having a more robust indication, but in practical terms those sensors won't guarantee you that you have a better detection. Anyone that knows much about those sensors would concede that you can still have an imperfect indication of what is ahead of the self-driving car. There are many factors that can limit the capabilities of those sensors…

Some would say that the self-driving car would make sure to have sufficient distance between it and other cars so that it could have the needed stopping distance unimpeded. I don't quite see how that is feasible per se. If the car behind you is on your tail, how do you ensure that there is sufficient stopping distance without getting rear-ended by that other car?

The answer usually is that the other car is being driven by a human and the 'stupid' human has not allowed for the proper stopping distance. Therefore, the problem now is that we have a human driver, which if we just remove all of the pesky human drivers and have only AI self-driving cars, we would not need to be concerned with cars being too close on our tails.

This will require me to take you on a related tangent about the nature of self-driving cars.

There are varying levels of AI self-driving cars. The topmost level is considered Level 5. A Level 5 self-driving car is one that is being driven by the AI and there is no human driver involved. For the design of Level 5 self-driving cars, the automakers are even removing the gas pedal, brake pedal, and steering wheel, since those are contraptions used by human drivers. The Level 5 self-driving car is not being driven by a human and nor is there an expectation that a human driver will be present in the self-driving car. It's all on the shoulders of the AI to drive the car.

For self-driving cars less than a Level 5, there must be a human driver present in the car. The human driver is currently considered the responsible party for the acts of the car. The AI and the human driver are co-sharing the driving task. In spite of this co-sharing, the human is supposed to remain fully immersed into the driving task and be ready at all times to perform the driving task. I've repeatedly warned about the dangers of this co-sharing arrangement and predicted it will produce many untoward results…

Let's focus herein on the true Level 5 self-driving car. Much of the comments apply to the less than Level 5 self-driving cars too, but the fully autonomous AI self-driving car will receive the most attention in this discussion.

Here [are] the usual steps involved in the AI driving task:

- Sensor data collection and interpretation
- Sensor fusion
- Virtual world model updating
- AI action planning
- Car controls command issuance

Another key aspect of AI self-driving cars is that they will be driving on our roadways in the midst of human driven cars too. There are some pundits of AI self-driving cars that continually refer to a utopian world in which there are only AI self-driving cars on the public roads. Currently there are about 250+ million conventional cars in the United States alone, and those cars are not going to magically disappear or become true Level 5 AI self-driving cars overnight.

Indeed, the use of human driven cars will last for many years, likely many decades, and the advent of AI self-driving cars will occur while there are still human driven cars on the roads. This is a crucial point since this means that the AI of self-driving cars needs to be able to contend with not just other AI self-driving cars, but also contend with human driven cars. It is easy to envision a simplistic and rather unrealistic world in which all AI self-driving cars are politely interacting with each other and being civil about roadway interactions. That's not what is going to be happening for the foreseeable future. AI self-driving cars and human driven cars will need to be able to cope with each other. Period…

Returning then to the matter at hand of the scenario about the driver and the shadowy figure in the roadway, we need to dispense with the notion that the cars around the self-driving car will be only AI self-driving cars. Realistically, there will be a mix of human driven cars and AI self-driving cars.

I say this to clarify that the scenario I've painted remains the same, namely the AI is faced with the matter of having to try and determine whether to hit the brakes but might get rear-ended, or swing into the next lane but might cause the other driver to veer into the median, or the AI might drive onto the sidewalk but maybe harm pedestrians, or the AI might continue straight ahead and potentially plow into the shadowy figure.

As mentioned, there are AI developers that claim that an AI self-driving car would not let itself get into such a predicament, but there doesn't seem to be any realistic world in which the AI could have magically avoided this situation and many other such situations. I'm putting a stake in the ground and will unabashedly say that there are going to be unavoidable crashes that AI self-driving cars will need to confront (and, of course, there will be avoidable crashes too, for which hopefully the AI will be astute enough to avoid).

I've stated many times that there are crucial ethical decisions or judgments that the AI will need to make when driving a self-driving car. I don't believe you can hide behind the matter by saying that the AI will never get itself into a situation involving an ethical decision or judgment. Saying this belies the very act of driving a car. Anyone developing an AI self-driving car that seems to think that the AI won't get itself mired into such situations has their head in the sand, and worse too they are developing an AI system that cannot presumably handle the real-world driving tasks that the AI will face…

For the moment, please go with me on the notion that the AI will need to cope with ethical decisions or judgments as part of the driving task. If that is indeed the case that the AI will need to deal with the matter, the question then becomes how it will do so. You might suggest that the AI needs to use common sense reasoning.

Common Sense Reasoning for AI Self-Driving Cars Not Available

As humans, we seem to have an ability of being able to use common sense about the world around us. We somehow know that a chair is a chair and that the sky is blue. We also presumably use common sense to decide when to slam on our brakes in the car versus swerving into another lane. Well, sad to report that we don't yet have any true semblance of common-sense reasoning for AI systems, and so let's count out for now the 'solution' that we could just plug-in common sense reasoning and have dealt with the ethical choices matter swiftly by doing so….

You might say that the AI should use Machine Learning (ML) to figure out how to cope with these ethics related decisions. Are you suggesting that we let AI self-driving cars drive around and sometimes they hit and injure or kill someone, and sometimes they don't, and by the collection of such driving instances that somehow over time the ML 'learns' which approach to take in these dicey situations? This seems impracticable. I would wager that most of us would not want to be one of the humans injured or killed during the thousands of such instances that the ML needed to collect to be able to find patterns and 'learn' from the experiences…

In short, the better approach would be to explicitly design, develop, test, and field the ethical decision making or judgment aspects into the AI.

Thus, since we don't have available as yet any kind of automated common-sense reasoning, and since relying upon ML to somehow miraculously over time figure out what to do (during which grave results are apt to occur), it would seem prudent to overtly tackle the problem and devise a system capability for the AI to rely upon.

If we do nothing, the AI will be unable to adequately perform when such moments arise, and the result will be likely random chances of the self-driving car either managing to avoid an incident or getting involved in an incident and doing so without any explainable rhyme or reason for it. I don't think we want self-driving cars to become clueless rockets of potential destruction.

Now, assuming that indeed the appropriate approach would be to devise a system component for this purpose of ethical decision making, this raises a slew of technological and societal considerations.

Should this be left to the automakers and tech firms to devise on their own, each independently creating such system components? This would seem somewhat questionable. If you have brand X self-driving car driving around and it is going to decide one way as to how to ascertain whether to proceed forward toward the potential pedestrian or weave or hit the brakes, and there is brand Y self-driving car that decides another way, it would be potentially confusing for the public at large as to what to expect from the AI of these self-driving cars.

Besides the aspect that each of the automakers or tech firms would need to reinvent the wheel, as it were, in terms of trying to come up with a viable approach, it would seem more consistent and transparent if some overarching approach were used. This too would deal with the potential thorny aspect that involves the crux of how the decisions are being made.

The thorny aspect involves how to decide what the 'best' course of action might be in these ethical dilemmas. I had earlier asked whether humans use some kind of mental calculus to determine which choice to make. Do humans weigh each factor? Do they consider whether age is important of those that might get injured or killed? How do humans do this? We can't say for sure how humans do it.

This makes trying to have an AI system do something similar a problematic issue. It would be handy to know how humans make such decisions and thus we could just pattern the AI to do the same. I've had some AI developers that tell me that all this will take is to ask people how they decide, and then essentially 'program' this into the system. As pointed out earlier, the rationalizations that people provide are not necessarily how they truly decide, and we are not even close as yet to being able to probe into the mind to discover how people really make such decisions.

Perhaps this takes us toward the ML approach and the need to collect sufficient data, though doing so via car accidents themselves would seem dubious. Another approach would be the use of simulations and have humans that gauge and make choices in the car driving simulations, out of which the ML might 'learn' the approach being used by humans (even if we don't know what's actually happening in their minds).

Another approach would involve using an actuary's kind of analytics method. As emotionally difficult as it might seem, there might well be a need to identify and agree to factors that should come to play in these decision moments. The result would be developed as part of the AI for use in the on-board system of the self-driving car. The same kind of gut-wrenching aspects are involved in trying to decide actuarial matters and thus it seems potentially fitting to use the same kind of methods for these purposes.

Rather than leaving this task to the automakers or tech firms alone, some have proposed that an Ethics Review Board mechanism should be utilised. These would presumably be special committees or boards that would meet to aid in determining the parameters and thresholds for use in the ethics aspect components of the AI self-driving car systems. It might be something crafted by industry or it might be something created via potential regulations and regulatory bodies.

These Ethics Review Boards might be established at a federal level and/or a state level. They would be tasked with the rather daunting and solemn task of trying to guide how the AI should be established for these tough decision-making moments (providing the policies and procedures, rather than somehow 'coding' such aspects). They might also be involved in assessing incidents involving AI self-driving cars that appear to go outside the scope of what was already established, and thus be an ongoing aid in the re-adjustment and calibration of the implemented approach.

Some have suggested that if there was an AI component for these ethical decision-making moments, and if there is a desire to standardise it across self-driving cars, perhaps the component should be housed in the cloud. Similar to how self-driving cars will be using OTA (Over The Air) electronic connections to update the AI systems, perhaps the AI component would not be embedded into the on-board system of the self-driving car and instead be accessed remotely.

Of course, the remote access aspects might get in the way of the decision making itself. It is more than likely that the ethics component would need to be accessed in real-time with split seconds to render a choice. Doing so via electronic connection seems dicey and prone to being inaccessible at the moment that the aspect is urgently needed…

What would seem prudent would be to have an on-board capability that could be updated via a cloud or centrally based standard. The on-board component would then be honed to presumably be able to render a choice in whatever sufficient time is available in a given circumstance. If insufficient time existed in any particular instance, there would need to be some shortcut choice capability, which I mention since once again the thought is to avoid an arbitrary choice and one-way-or-another have a 'reasoned' choice that can be understood and explained.

One question that some have posed is whether this ethics decision making component could be truly able to handle all of the many variants of the Trolley problem. For example, I've outlined the case of the driver that is not sure if a pedestrian is in the road, and there is a car immediately behind, and there is a car to their left. Surely there are thousands of such potential instances, all of which would have variants. How could a system possibly contend with so many variations?

I'll bring us back to the aspect that humans seem to be able to contend with these multitude of variants. I'd guess that the driver faced with the situation I've outlined has not experienced that exact situation before. Instead, they have an overall experience base and need to use whatever they can to try and apply it to the moment and the situation at hand. Presumably, the AI component would need to do the same. Plus, the AI component would be sought to be adjusted and enhanced over time (via the use of the Ethics Review Boards).

There is unquestionably bound to be controversy about the notion of the Ethics Review Boards. Some suggest that they should be called Safety Review Boards or perhaps Institutional Review Boards, providing a naming that might be more palatable. There are some that have pointed out that there is the possibility of having them become labelled as 'death panels' as per the political term that arose during the 2009 debates about aspects of the federal healthcare legislation (this phrasing seemed to strike a chord with the public at large, though there is quite a dispute about the merits of the labelling).

In one sense, it could be argued that the Ethics Review Boards would be shaping how the AI will respond to dicey driving incidents, and as such, those Boards are deciding how life-or-death decisions will be made. It would be no easy matter for the members to serve on such a committee. Careful selection and criteria for participation would need to be figured out.

As unseemly as it might seem to have such Boards, the alternatives are to allow whatever happens to just happen or allow for particular automakers or tech firms to make those a priori choices for us all. It would seem to be the case that society would likely prefer the more open and transparent and collective approach of using the Boards, but this is something yet to be ascertained.

A few final comments that I'd like to cover on this topic encompass various security related aspects.

One concern would be that a hacker might somehow be able to mess with the on-board ethics choice component and alter it so that it would do something untoward. When the ethics component is involved in a dire situation, the hacked version might make a choice that purposely seeks to maximise injury and death, rather than minimise it. Of course, systems security does need to be paramount for the on-board AI, and in fact I'd suggest that if the hacker could hack pretty much any part of the AI of the self-driving car, the odds are they can produce an untoward result in some fashion.

In essence, rather than focusing on solely the ethics component, nearly any other element of the AI system if hacked can likewise produce adverse consequences. As such, all I'm saying is that you cannot argue that there should not be an ethics component due to the potential for it being hackable, since you could make the same argument for nearly all other components of the AI system for a self-driving car. If you then are making the argument that any of those components could be hacked and therefore they are inherently untrustworthy, you might as well then say that there is no such viable thing as an AI self-driving car.

In a somewhat similar manner, let's consider the cloud and the OTA. One might argue that suppose a hacker gets to the cloud version of the centralised ethics component and messes with it. The hacker has made things presumably easier for themselves in that they didn't need to try and access any particular self-driving car, and instead they will let the OTA do so for them. The OTA would presumably blindly and dutifully send the updates to the on-board AI systems and thus allow a viral-like spread of the untoward ethics component aspects.

I'll invoke the same argument as before. Yes, if a hacker could hack the centralised version, it would potentially produce this kind of calamity. I would submit though that if the hacker could alter nearly any aspect of the centralised patches that are going to be pushed down to the AI self-driving cars, you can have an untoward result. As such, the security needs to be quite tight at both the on-board self-driving car and at the OTA cloud-based elements. Either one can allow for something untoward if the security is not sufficiently tight.

I've had some AI developers tell me that their 'solution' to these ethical choice situations involves having as a default that if the self-driving car cannot decide what to do, it will simply slow down and come to a halt. I hope that you can readily see that such an approach is nonsensical. Using my earlier example, would we have wanted the driver to have simply slowed down and come to a halt? This is quite impractical in the given situation and as I say is a nonsensical way of thinking.

Another idea that has been offered would be to ask the humans in the self-driving car as to what the AI should do. Again, a nonsensical answer. First, suppose there aren't any human occupants in the AI self-driving car at the time of such a decision-making moment? We are going to have AI self-driving cars driving around on their own, quite a bit.

Second, even if there is a human on-board, would they be able to out-of-the-blue be able to make such a decision? Let's assume they aren't driving and aren't paying attention to the driving task, which in a Level 5 self-driving car is indeed their prerogative.

Third, suppose the human on-board is drunk? Suppose the human on-board is a child? Suppose there are humans on-board and yet the decision needs to be made within 2 seconds – how could the humans be told the problem and offer an answer in a mere second's worth of time. And so on.

Another point some make is that maybe we should setup remote human operators that would make these decisions. Sorry, it's a nonsensical idea. Suppose the remote operator could not fully grasp the nature of the situation? Suppose they only had two seconds to decide and meanwhile they somehow needed to "review" what the situation is and what options to consider. Suppose there are electronic communication delays or snafus and the remote operator is not able to participate in the time needed? And so on…

I'd say that the automation is what is going to get us into this predicament, and it would seem like the automation is the only means to get out of it (as coupled with the Boards and the approach to devising the solution). Though, when I say get out of it, let's be clear that however this is devised, the odds are that the AI system will be second-guessed about the choices made. This would be true of humans and it will certainly be the same about the AI. The AI might 'perfectly' execute whatever the AI ethics component consists of, and yet still human lives might be lost.

There are unavoidable crashes that no matter what you do, a crash is going to occur. For my earlier example, suppose it really was a pedestrian in the roadway. And, suppose that each of the choices involved either injuring or killing someone, either the pedestrian, or the driver in the car behind you, or the driver in the car to your left, or you as the driver. There is not going to be a magical way to get out of the unavoidable crashes unscathed.

Would we prefer as a society to pretend it won't happen and then wait and see? Or, would we rather step-up to the matter and address it head-on?

Time will tell" (Elliot, 2018).

While the ethics of self-driving cars is imperative, necessary to take note of, and still a much needed debated across the globe, "technology is advancing at a phenomenal pace and it's showing no sign of slowing down" (Centre of Technological Excellence, 2020).
Therefore, "…with the swift pace of change and disruption to business and society, the time to join in is now" (Coleman, n.d.).

This research focuses on technological advancements with self-driving cars and the Internet of Things (IOT).
The ethics of self-driving cars is a separate topic outside the boundaries of this research, dealt with by experts and Ethical Review Boards that conduct in-depth social science research on the moral values and principles involved in the use of disruptive technologies.

CHAPTER 1: INTRODUCTION

1.1 Introduction

The automotive industry is a critical component of the world economy, employing millions of workers across manufacturers, suppliers and dealers; however, mobility is increasingly expensive, inefficient and unsafe. South African road fatalities in 2012 exceeded 14,000 annually, with 150,000 injuries, and a cost to the economy of around 210 billion rand per year (South African Road Safety Audit, 2012:iii-iv). By 2019, there had been a slight improvement but the figure was still too high, according to the Automobile Association (RTMC, 2020). Running costs of owning a vehicle, together with roadway construction and maintenance and with the fact that the average person spends 260 hours of lost time annually in the driver's seat because of congestion, are astronomically high. In urban areas, 40% of fuel is used in looking for parking, and in some US cities, parking lots cover over a third of the land area (Kimmelman, 2012). Furthermore, there is a global "go green" drive to pursue practices that are environmentally friendly in an endeavour to reduce global warming, with an emphasis on reducing resources contributing to greenhouse gas emissions such as fossil fuels (Chou & Chou, 2012:447-448).

Google and Tesla, among several other motor manufacturers, have conducted a considerable number of studies on autonomous vehicles (Greenemeier, 2016). Google's concept of a driverless taxi incorporates a mix of global positioning systems (GPSs), lasers, altimeters and gyroscope technology, which, in a combined effort, navigate a vehicle to predetermined locations without human intervention (Joy, 2014; Standage, 2018). Besides the cost of putting such a vehicle on the road, overall vehicle integrity raises concerns over safety aspects, with emphasis on the safety and security of communication systems.

According to Mervis (2017) and Litman (2020), AVs will occupy roads alongside present-day vehicles for some time to come dictated by technology, expenditure and road infrastructure. The crucial element of human safety necessitates a system that responds instantaneously, such as fibre optics or electromagnetic radiation (radio waves) moving at the speed of light, and in use worldwide as the chief source of communication, transporting significant volumes of data over a considerable distance (Howard, 2011). Until recent advances in Assist Global Positioning System (AGPS) latency in GPS, satellite transmission has been a major limiting factor regarding collision avoidance, and a setback to the implementation of Level 3 vehicles on our roads (Yass, 2017). This research proposed an alternative solution using IoT to accommodate vehicles on our roads.

This research focused on exploring the workings of a virtual system and applying the same principles to *route* autonomous vehicles; this means the use of data networks using Internet

Protocol (IP) addresses instead of GPS, with its stream mapping data. Cisco routers, their OS protocols and embedded programming were investigated in the process, as well as a variety of contemporary detection and communication techniques such as machine learning (ML) and their underlying properties in order to develop an Internet of Things (IoT) solution. This research is multidisciplinary in nature, with the proposed IoT solution falling within the information technology discipline.

It is feasible to assume that with an ever-increasing number of motorists, along with an increase in defiance and human error factors, the ultimate goal of research lies in designing a completely automated guideway system; however, implementation is a process and this research is a progressive step towards achieving this goal. Viability depends on return on investment (RoI), and the research further suggested implementation within a South African perspective (see section 1.3 for detail on the South African perspective).

1.2 Keywords

The following keywords and key concepts delineated the research:

- Internet of Things (IoT)
- Computer simulation
- Multidisciplinary
- Machine learning (ML) > artificial intelligence (AI)
- Routers > external networks
- Switches > internal networks
- Radio navigation beacons > transmitting sources
- Radar and light detection and ranging (LiDAR)
- Packet switching

1.3 Background to the research

An increase in population and in middleclass earners who are attracted to employment prospects in urban areas contributes to traffic jamming (Padayachi & Thambiran, 2012:1). In South Africa, freight by rail has reduced, resulting in a switch to the use of extra heavy commercial vehicles (EHCVs) for freight transport, even though Transnet Freight Rail has invested R201bn in infrastructure over time. This has led to increasing numbers of heavy vehicles on already congested roads (UD Trucks, 2013). Moreover, according to the 2016 carbon budget annually updated by the Global Carbon Project (2016) (Le Quéré et al., 2016), South Africa has the highest carbon dioxide (CO^2) emissions from vehicles in Africa, and twice the global average, which contributes vastly to the annual increase in the greenhouse effect (CSIR, 2016).

Despite efforts to mitigate the situation, fatalities continued to increase and during 2010/2011, a person died every 38 minutes on South African roads. By 2015, the country was rated as having one of the worst fatality figures globally (South African Road Safety Audit, 2012; Taylor, 2016; WHO, 2015). Stringent road and safety laws since implemented have had little impact on improving the situation, as they do not eradicate non-compliance and human error. Furthermore, an inadequate response to road maintenance issues and a lack of enforcement of the rule of law exacerbates the situation, while road fatalities and injuries steadily increase (Taylor, 2016).

Human error or bad choices were responsible for 94% of accidents. According to Peissener, Doebler and Metze (2011) and Pyper (2014), the risk potential for an accident to occur increases as drivers practice secondary activities that divert attention from the primary task of driving, such as talking on cell phones, which in turn influences cognitive responses. Any activity that takes the driver's attention away from the primary driving task is considered a distraction (Horberry et al., 2006; NHTSA, 2014). In the United States (US), nearly 80% of accidents and 65% of near-accidents involve some form of driver inattention within three seconds before the event happens (NHTSA, 2014). In his 2011 TED talk, Professor Sebastian Thrun (2011), Director of Stanford University's Artificial Intelligence Laboratory, stated that people would wonder why they ever thought to drive cars themselves. By cutting down the human intervention element of driving to a considerable degree, roadways will become safer areas of passage and autonomous vehicles will inexorably help to alleviate congestion and the accident rate.

In this research, the researcher demonstrated by means of an IoT model how the concept of packet switching can be applied effectively to route real-life autonomous vehicles on our existing road infrastructure. A packet is a basic unit of communication transmitted across one or more digital networks. In order for the data to be transmitted, it is broken up into similar tiny edifices that, on reaching their destination, reconstruct themselves into their original form (Sanders, 2007).

What is the problem?

Literature on information systems (IS) and empirical research on the packet switching concept as a routing mechanism for autonomous vehicles is limited. Only meagre related literature exists, such as research conducted in the USA (Fiske, 2002) to route magnetic levitating vehicles travelling along a track, and which has been conceptualised in an MTech dissertation by the researcher (Wahl, 2013). Communication within the current context refers to the conveying of messages between a sending unit and a receiving unit over a medium. Such transmission of data over a digital network via packet switching has been utilised extensively as a communication medium in computer networks (Sanders, 2007).

Other developments include technology that constructs three-dimensional maps of the surrounding area and processing of all data at a rate that allows operation from inside the vehicle. Delays in communication experienced in transmission when using the GPS have largely been rectified with the latest development of AGPSs that use cell phone towers; however, the potential for tower outages still exists. In addition, autonomous vehicles are prohibited from operating in bad weather conditions.

GPS is a global radio navigation system controlled by the USA Department of Defence, involving a constellation of orbiting satellites and their ground stations (USA Department of Defence, 2008; Garmin, n.d.). Microwave signals are transmitted to GPS devices to provide information on location, time, direction and vehicle speed, which are processed and computed by a receiver (Bertnaga, 2010). Furthermore, the possibility of intercepting transmissions poses a security concern, even though modulation and encryption methods would have to be decrypted (Muoio, 2016). Google and Tesla have shown that autonomous vehicles are a feasible futuristic development taking all advances into account in the IoT, including developments such as computer vision affiliated to ML, which can distinguish objects and include a feature used to discern behaviour of other road users, vehicles or pedestrians.

1.3.1 Internet of Things

The term ‘Internet of Things’ (IoT) was coined by Kevin Ashton of Proctor & Gamble and later co-founder of the Massachusetts Institute of Technology’s Auto-ID Centre in 1999 (Ashton, 2009). IoT refers to any network of physical objects combined with electronic devices, including vehicles embedded with sensors, electronic actuators, or software that are connected via a network, and which allows them to communicate and exchange information or data with each other (Brown, 2016).

Raji (1994) had already envisioned the IoT concept, describing it in the IEEE Spectrum magazine as small packets of data that are conveyed to a large location of nodes, allowing everything to be combined and automated, from home appliances to entire factories. Objects can be controlled by means of the IoT as they inter-operate within the internet infrastructure, individually identified through their embedded computing systems. A more direct integration of the physical world into computer-based systems reduces human intervention and ensures efficiency and accuracy, while being more economically viable (Santucci, 2011; Lindner, 2015).

Figure 1.1, compiled by the researcher from the description by Zhuge (2004), describes the basic concept of a model proposed in 2004 of a futuristic interconnected environment, laying the foundations of IoT. The ternary universe encompasses physical, virtual and mental worlds consisting of primary levels that begin with the bottom physical level, comprising nature and devices; followed by the virtual world; with the human stimulus at the top level, to promote active and cooperative assistance in accomplishing tasks and solving problems by utilising the network (Zhuge, 2004, 2005). The increasing number of devices connected and controlled by the Internet means the IoT is not excluded from the need for stringent security standards. Devices need to be created by companies with this in mind (Porup, 2016), as it was accurately assumed that by 2019, the Enterprise Internet of Things (EIoT), a term referring to devices used in business and corporate settings, would account for almost 9 billion devices (Greenough, 2015). The Industrial Internet of Things (IIoT) encountered in the manufacturing industry is estimated to generate $12 trillion of global GDP by 2030 (Daugherty et al., 2015). Interfacing between the human world and the cyberworld, particularly in large industrial corporations, is planned by means of an architectural 5C system (connection, conversion, cyber cognition and configuration), converting data into information that spurs the physical assets on to produce the required results.

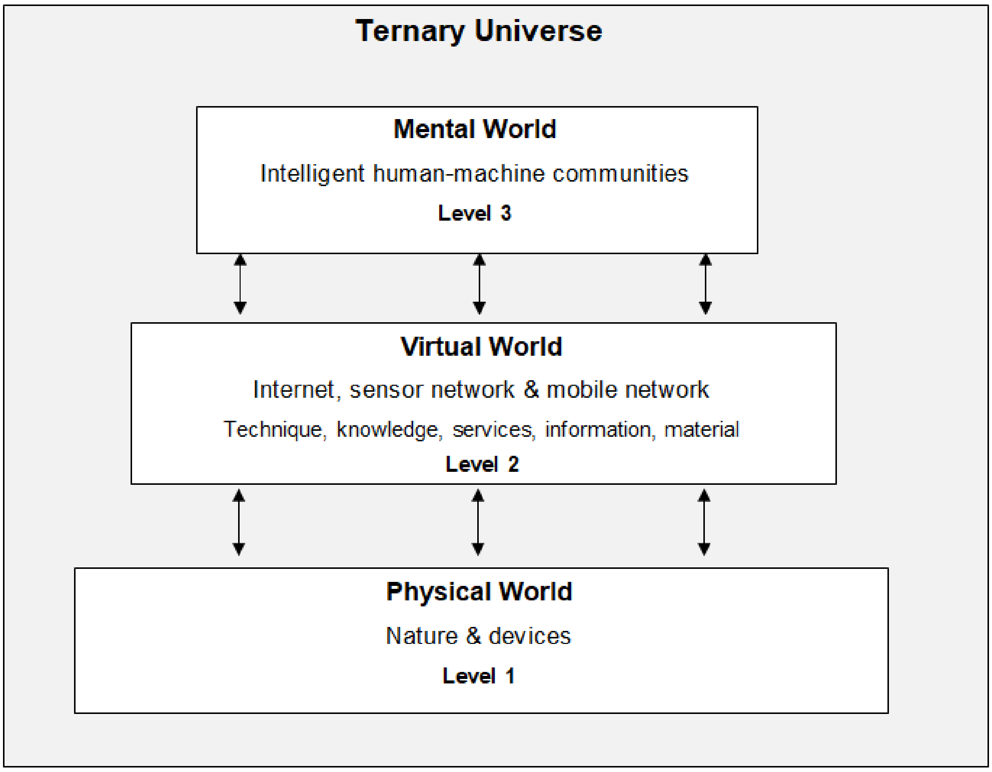


Figure 1.1: Model-prompting trend for the IoT

IoT can monitor and manage infrastructure such as structural conditions of bridges or roads that could compromise the safety of autonomous vehicle occupants. IoT can furthermore be extended to all facets of the transport system to enable vehicle-to-vehicle communication, smart parking and vehicle control. Research to integrate IoT and autonomous control is shifting towards an open network of intelligent entities, such as Web services and service-oriented architecture (SOA) components, as well as virtual objects (avatars) that are able to operate by deciding independently whether they choose to share objectives with other virtual objects in finding a solution or whether to follow their own initiative to solve a problem. This is accomplished by the ability of objects to sift and reason through collected data, detect environmental changes or defects affecting sensors, and, consequently, apply appropriate solutions (Alippi, 2014).

Enabling technologies for IoT means that objects are allocated a unique identification IP address or a uniform resource identifier (URI) that identifies the name and location of a file; however, due to limited address space of IP version 4 (IPv4), IoT enabling technologies are now being supported by IPv6, which is the follow-up generation of the Internet Protocol System. The network to communicate between devices is provided by several wireless (short- to long-range) or wired (Ethernet or power line communication) technologies (Deering & Hinden, 1998).

This research focused on the data transmission aspects of IoT, with particular emphasis on Ethernet network infrastructures as a means of communication between IoT devices and receiving nodes. The communication principles are analogous to those in a traditional local area network (LAN) configuration where the Open Systems Interconnection (OSI) framework is employed. Medium to enterprise LAN configurations customarily include network routers and switches in their architecture in order to relay data to a predetermined destination, either internally or outside of the LAN. This routing mechanism is also referred to as packet switching, which is the focal point of the research, i.e., incorporating the mechanics behind packet switching technology into a routing IoT solution for autonomous vehicles.

An example of IoT devices configured in an Ethernet network is the ‘smart home’, which comprises a cluster of IoT devices all wired to a central point of communication or IoT access through network cables, but can also be configured wirelessly. This access point is usually a router issued by a service provider. Smart home devices typically consist of an array of sensors that monitor water and temperature as well as light contrast, among many other applications (Chambers, 2011). Another example of IoT devices are those used in industry with the main purpose of recording and monitoring functionality of various factory-grade machinery, and which can also control components of such machines on demand and from a remote location. Aspects of disparate manufacturing plants can now be synthesised and controlled from a central point of access with the aid of IoT (Porup, 2016).

To demonstrate controlling AVs through the concept of packet switching, the researcher contributed to the existing body of knowledge by developing a prototype (computer simulation) reliable routing system that reduces dependence on vehicle-to-vehicle (V2V) communication, GPS and road topography.

1.3.2 Rationale and motivation

Autonomous vehicles on our roads could decrease accidents and fatalities, resulting in a reduction in i) health-care costs, ii) lost work time, iii) litigation costs and iv) other adverse consequences (Lissy et al., 2000). Significant continual technological advances in autonomous vehicle technology have made the notion of AVs an imminent and acceptable reality, with sustainability envisaged through the improvement and management of the wellbeing of road users (Giarratana, 2016; Tsiu, 2020). Furthermore, economic benefits and quality of life can increase significantly, particularly when elderly and disabled persons are able to achieve a considerable level of independence (Giarratana, 2016; Jansen, Li & Lorenz, 1995). At the same time, besides increased work performance and optimised work hours (Triplett et al., 2014), the AV concept offers possibilities for entrepreneurship on a macroeconomic level.

According to the USA Department of Transportation, besides the potential to avert a high percentage of vehicle accidents, V2V on-board communication and vehicle-to-infrastructure (V2I) communication technology has been shown to increase the overall efficiency of vehicles and reduce emissions (Pyper, 2014). South Africa is a member country of the Kyoto Protocol, drafted by the United Nations Framework Convention on Climate Change (UNFCCC). The country is therefore committed to decreasing greenhouse gas emissions through the development of technologies and processes designed to encourage local governments to be more amiable in granting clearance for the introduction of numerous new autonomous vehicles on the road (UNFCCC, 2010; South African Department of Environmental Affairs, 2013).

Conventional autonomous cars rely extensively on GPS, whereas in this research, an uncomplicated packet switching routing mechanism that reduces satellite dependency on GPS has been proposed and demonstrated by means of an IoT solution. Latency, congestion on the network, and any other problems experienced within the packet switching process is automatically rectified by the system itself.

1.4 Problem statement

The growing number of vehicles on our roads and the consequent accident death rate is detrimental to South Africa’s economy (Nel, 2019). Autonomous vehicles, such as those proposed by Google and Tesla to run concurrently with conventional vehicles on existing roads, are exorbitantly priced (Ackerman, 2016). Furthermore, as mentioned in section 1.3, autonomous vehicles do not function effectively in inclement weather conditions and are completely reliant on GPS. This study endeavoured to show a possible way that autonomous vehicles could function effectively using the fundamental concept of packet switching as a routing mechanism that encompasses IoT technologies.

1.5 Research aim

This study aimed to explore the probability of developing a safe, reliable, cost-effective and faster method of routing autonomous vehicles by adopting packet switching as a router mechanism. The research further aimed to propose an IoT model encompassing vehicular wireless technologies, and to explain and substantiate the operation of such a system based on empirical simulation.

1.6 Research questions

A primary research question and secondary research questions guided the study.

1.6.1 Primary research question (PRQ)

PRQ: What IoT solution can be proposed to incorporate the virtual packet switching concept of traversing data as a router mechanism for controlling real-life autonomous vehicles?

1.6.2 Secondary research questions (SRQs)

SRQ1: What technologies support autonomous vehicles?

SRQ2: What technologies embrace packet switching?

SRQ3: What cybersecurity technologies are available to protect against system security threats and attacks?

SRQ4: What is the magnitude of environmental and societal benefits to developing such a system?

SRQ5: What are the challenges in applying the virtual packet switching concept to serve as a routing mechanism for real-life autonomous vehicles?

SRQ6: What model can be proposed to validate the operation of packet switching as a routing mechanism for autonomous vehicles in the South African context?

1.7 Research objectives

The primary and secondary objectives that qualified the research questions are stated below.

1.7.1 Primary research objective (PRO)

PRO: To determine, by means of a proposed IoT solution, the viability of applying the virtual packet switching concept of traversing data as a routing mechanism to real-life autonomous vehicles

1.7.2 Secondary research objectives (SROs)

SRO1: To determine the state-of-the-art technologies supporting autonomous automobiles

SRO2: To determine the technologies involving packet switching

SRO3: To identify the availability of security technologies to safeguard system security against attack

SRO4: To describe the environmental and societal benefits of developing the system

SRO5: To determine the challenges surrounding the technical implementation of the system

SRO6: To propose an IoT model to validate the operation of packet switching as a routing mechanism for autonomous vehicles in the South African context

1.8 Research design and methodology

A brief overview of the methodological deliberations that were followed in the research study is provided in this section. In Chapter 3, the researcher provides a comprehensive description of the research design and methodology used to address the research questions. Research is defined as a search for knowledge (Kothari, 2004), while research design is a plan or procedure for research, ranging from broad assumptions to detailed methods of data collection and analysis. Research design furthermore deals with the philosophical assumption(s), strategy of enquiry, and specific methodology used in this study (Creswell, 2009). Methodology delineates data and data analysis procedures, and according to Peffers et al. (2007:5), methodology is “a system of principles, practices and procedures applied to a specific branch of knowledge”. This means that, in order to answer the research questions, the design explains the data required, the methods followed in the data collection process, and the method used to analyse the data.

1.8.1 Strategy of enquiry and philosophy

Computing often provides an analogy of the real world, and an awareness of one’s environment can encourage creativity to help find the metaphor that resembles a concept (Olivier, 2011). This researcher adopted a quantitative research approach within a design science paradigm; however, the researcher acknowledges Trochim and Donnely’s (2008) findings that qualitative and quantitative data are unambiguously linked, as all quantitative data are based on qualitative judgments, and all qualitative data can be described and deployed numerically. Similarly, IS research is characterised by design science and behavioural science. Design science is rooted in engineering and the sciences of the artefact, while behavioural science has origins in natural science research methods (Hevner et al., 2004). Design science seeks to solve problems through creating innovations that can be designed and implemented using IS to accomplish this.

Quantitative research is a systematic and empirical investigation of observable phenomena through mathematical, statistical, or computational techniques, and is a linear, pre-planned research design (Given, 2008). Quantitative research adopts the positivist epistemological stance and an objectivist attitude rooted in Auguste Comte's (1830-1842) positivism framework, which combines a deductive approach with precise measurement (Kasim, Alexander & Hudson, 2010).

Archival research was conducted by collecting data on aspects of packet switching from literature such as books, journals and the Internet, utilising databases such as Science Direct, SpringerLink and the Institute of Electrical and Electronics Engineers (IEEE, 2017). The positivist stance together with the accompanying deterministic philosophy adopted by the researcher holds that causes determine outcomes, and this needs to be taken into consideration with data collection (Creswell, 2009). The data collected through archival research were analysed and interpreted as a depiction of the underlying algorithms working in symmetry with all key components of the simulator (Figure 1.3) in relation to the constructed environment (Creswell, 2009). Potential threats to the security of the simulated environment were also examined. The philosophy behind IT research considers computers as human creations and suggests that most IT research attempts to realise theories acceptable to researchers in order to guide the construction of automated systems (Kuhn, 1970). The range of theories extends from engineering aspects of the methods used in constructing a system to the impact that human nature exerts on the development of automated systems (Olivier, 2011; Soare, 2009; Turing, 1939).

1.8.2 Methodology

Given the pioneering nature of this research, the researcher examined several characteristics of quantitative research, with a specific interest in descriptive design and experimental research. Explanatory research entails establishing any causal relationship between variables in order to explain factors concerning the research problem (Saunders, Thornhill & Lewis, 2009) and, to an extent, was found relevant in this research. Descriptive design measures an association between two variables (independent and dependent) and, to a certain degree, would also have fulfilled the purpose of this research (Grove, Burns & Gray, 2013). Alternatively, experimental research is a scientific method used to establish a cause-effect relationship between variables, where one variable (independent variable) is deliberately manipulated to determine the effect of this manipulation on the other variable (dependent variable). An experiment can also be designed to determine 'what-if' without a clear expectation of what might be revealed by experimentation (Bacon, 1620).

After deliberation, it was decided that **experimental research in the form of a computer simulation** was the most appropriate to adopt as approach, since the research used a computer program that simulates a theoretical model of a particular system, and this approach best describes the intentions of the research. Computer simulation, referred to as "an active model" or a "computational experiment" (Olivier, 2011:11), can be applied to model actual experiments and observations, such as the one described in this research. Models are used in design science for several reasons, some of which include:

i) Simplicity – to be able to understand the principle of a concept.
ii) Comprehensiveness – models can methodically address all or most features of a problem.
iii) Clarity – all elements of a procedure are evident.

For simulation, a real process or action has to be imitated to prove a hypothesis. Simulation is suitable for observing correlations between variables, and the method allows for estimation and prediction (Bhattacherjee, 2012; Myers, 2009). The goal of simulation is to examine the effect of a new algorithm, as was intended in this research. The computer model is the actual prototype containing the algorithms and equations used to imitate (simulate) the behaviour of the situation under demonstration.

A model was proposed as the major thrust and outcome of this research (Figure 5.1 to Figure 5.4). The research was initiated by proposing the development of a prototype to represent realistic aspects of the real-life environment. By means of IoT vehicular technology, the researcher then followed the research process and demonstrated through algorithms (Merriam-Webster, 2020) that the concept can work in practice. According to Olivier (2011:9), the construction of a prototype demonstrates insight into a new model. For this research, the proposed model is therefore a main outcome of the research. The **model** comprises the developed **prototype** containing the **algorithms** that **simulated** the theoretical model of IoT vehicular technology (discussed in Chapter 5). The merits of the constructed prototype were analysed and discussed, and the results obtained from the implementation exercise were shared. Figure 1.2 depicts the building process of the proposed computer model, showing how the interaction of simulation and theory was applied in this study.

In other words, insights can be gained into new technology and into analytical solutions applied to complex performance systems (Strogatz, 2007), all of which are aligned to the aims of this research. Independent pairs of attributes classify computer models, and aspects of this research seemingly required delving into both continuous simulation models and event-based models, such as discrete event simulation (DES) that manages events in time. The majority of computer simulations, including DES, are logic-test and fault-tree types. A simulator in this type of simulation manages a queue of events that are sorted according to the simulated time when it should take place. The simulator reads the queue and initiates a new event as the preceding event is processed. The data produced by the simulation can be accessed to discover logic defects in the design or in the sequence of events (Nutaro, 2010).

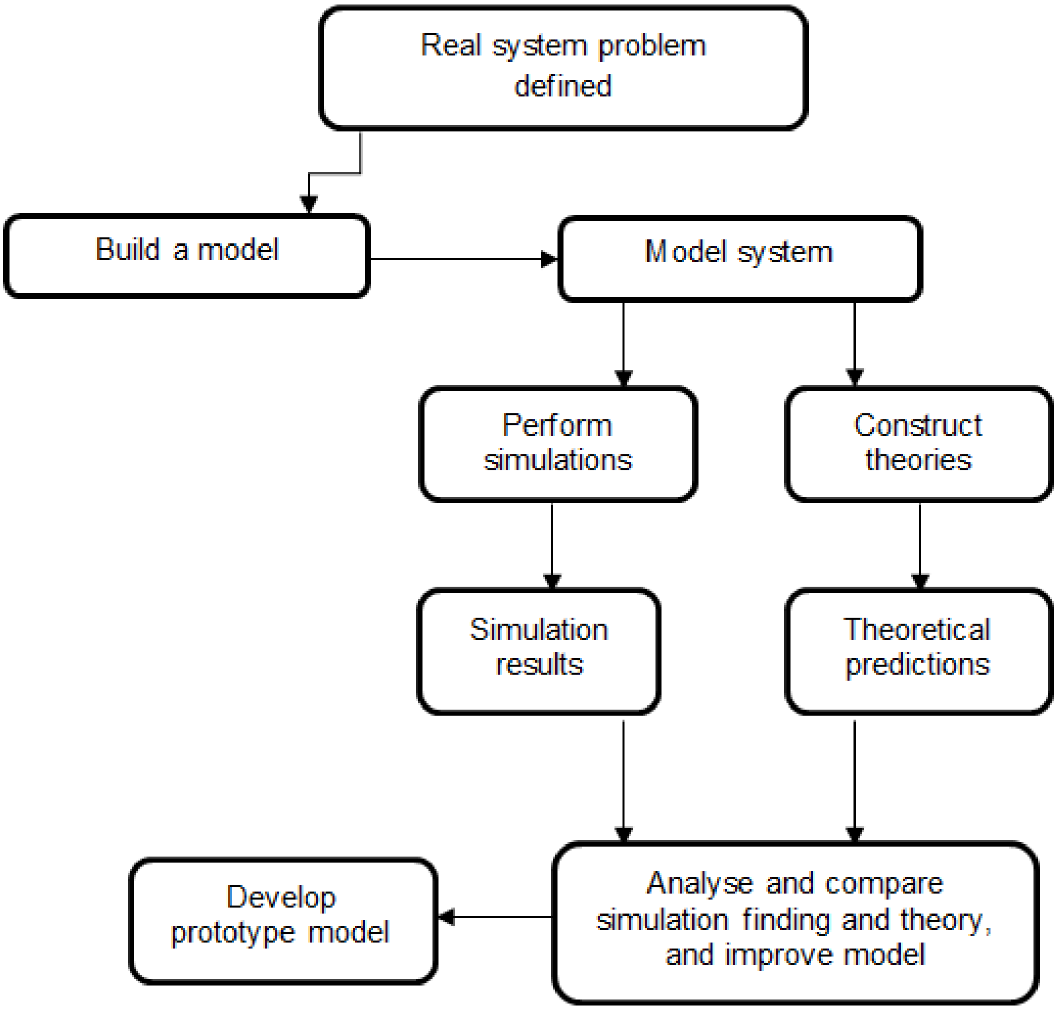


Figure 1.2: Process of constructing the proposed computer model (Source: Researcher)

The researcher hypothesised that the characteristics of models such as DES could be utilised to simulate the routing of autonomous vehicles by means of the packet switching process. In order to demonstrate the routing of AVs by means of packet switching through experimentation, existing computer simulation software was examined to determine the possibility of adapting the software to this research. Open-source discrete-event simulation software such as Facsimile Simulation Library and GNU Octave, as well as NetLogo, an open-source multi-agent simulation, were considered and the most suitable to this research was determined to be Psimulator2 (GitHub, 2008), developed by European researchers.

1.8.2.1 Simulation

To create a model, the researcher made use of an open-source network simulator, **Psimulator2** (Figure 1.2), developed in JAVA by the Faculty of Information Technology at Czech Technical University in Prague to teach basic networking topics to their students (GitHub Inc, 2008).

Figure 1.3 depicts the necessary tools – developed by the researcher using GitHub Inc's (2008) background information – contained in the simulator and used to design, configure and implement a basic network topology that utilises the quintessential networking building blocks (router, links and switch, among others) that only support a small subset of commands that typically exists in Linux and Cisco routers (Linkletter, 2016). The researcher postulated that this would suffice for the purpose of the research as, besides being open-source, the simulator allowed for the provision of a platform with the necessary scaffolding that could be utilised and customised by the researcher to simulate a routing mechanism for autonomous vehicles.

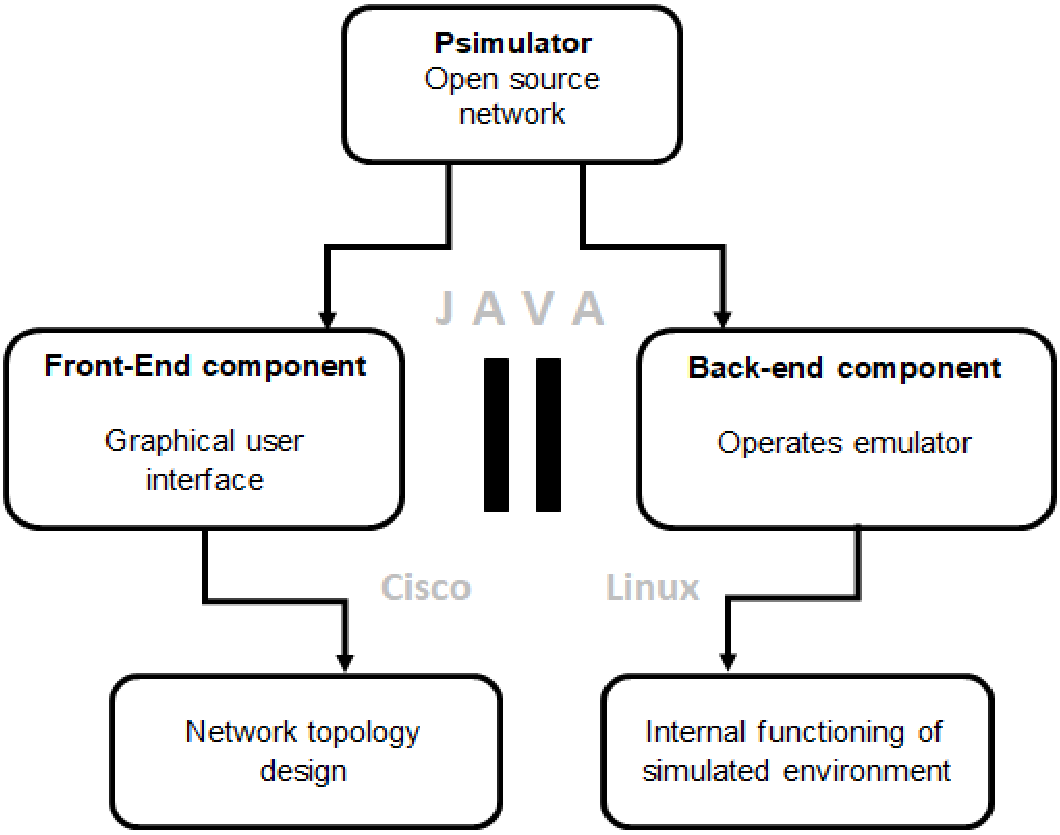


Figure 1.3: Basic components and function of Psimulator[1]
(Source: Researcher)

[1]Psimulator, Psimulator2, and the Psimulator franchise are all the same concept. The researcher designed a modified Psimulator (prototype).

The simulator comprises two components – a front-end and back-end. The front-end component is a graphical user interface (GUI) that allows users to design network topologies and view the transmission status of packets using simple animation. The back-end is a network that runs the emulator and is responsible for the 'inner workings' of the simulated environment, and which runs in parallel with the front-end. The user first needs to finalise a design that enables communication to be established with other nodes or components in the design via the back-end emulator. An emulator replicates the simulated real-life situation under demonstration (Churchman, 2017; Law & Kelton, 1991).

The researcher downloaded the source code for the simulator, which is available online, and modified (or customised) this to represent a vehicle routing system using similar routing mechanisms and principles to that of Cisco and Linux routers. The bulk of the code alterations were made on the front-end component of the application where the graphical representations of existing network components were replaced by components resembling a road system.

To create an efficient model, the researcher took into account that problems exist which inhibit the manufacture of detection equipment that possesses the sophisticated capabilities of detecting non-motorised bodies not fitted with transmitting devices, such as pedestrians and cyclists. Aspects of LiDAR, in conjunction with detection sensors previously proposed by researchers to achieve this, were revisited (Schippers & Socash, 2017) to determine whether these needed to be integrated into the model. To facilitate an effective routing system on conventional roadways, a model would have to be equipped with additional safety apparatus such as the use of multiple radio beacons to update on-board systems by loading accurate data at strategic positions; however, for the purpose of this research, such requirements were to be retained at a minimum. Moreover, the application of how artificial intelligence (AI) featured within the system also needed to be demonstrated.

The researcher endeavoured to address any criticisms and security intrusions probable in the wireless transmission which also needed to be accommodated and engaged with in the model. A Programme Evaluation Review Technique (PERT) analysis, i.e., a graphical illustration of the project's schedule, aided in determining tasks to complete the model within a certain timeframe.

The steps depicted in Figure 1.4 were pursued in the process of designing the conceptual model of the simulator.

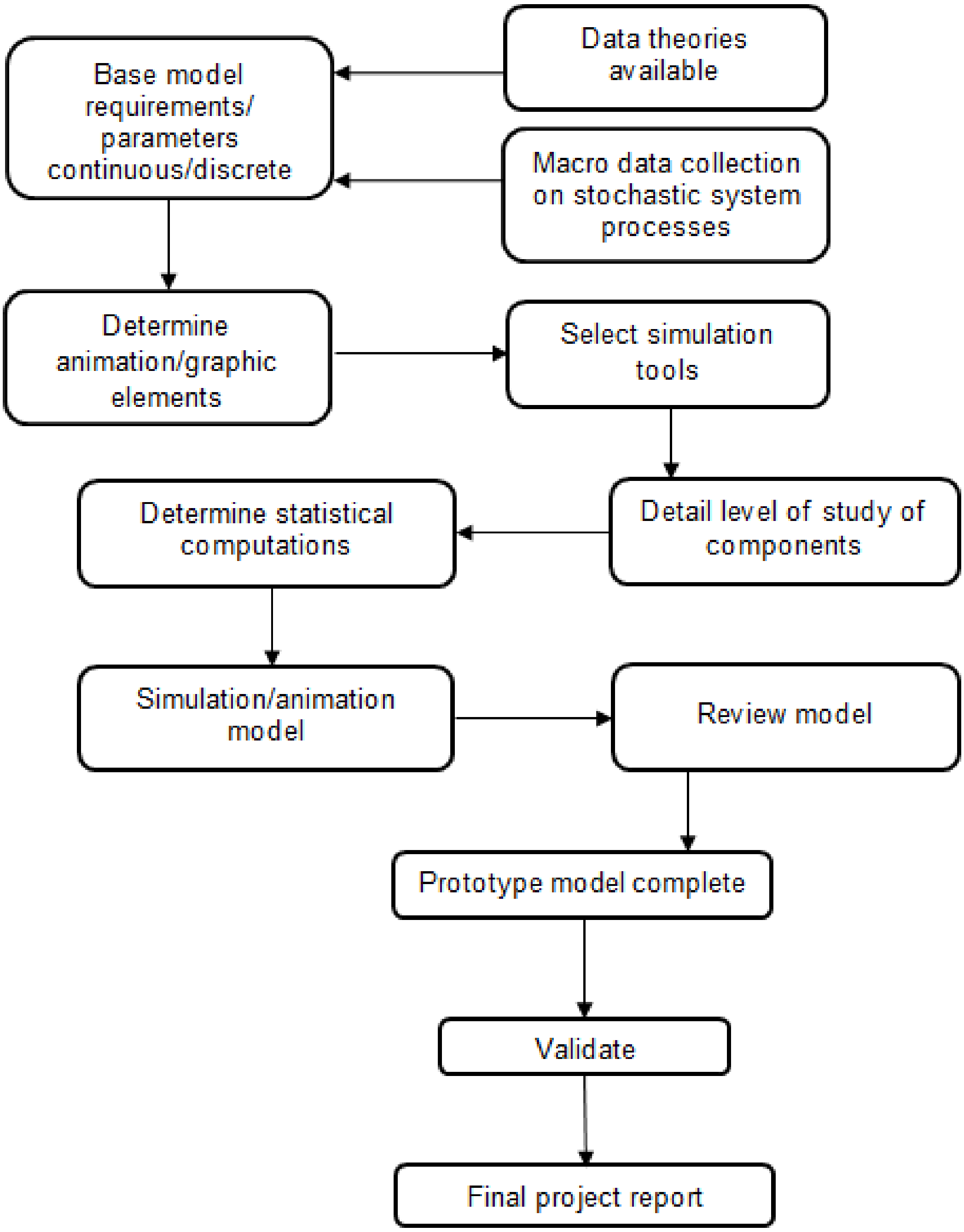


Figure 1.4: Drill down on proposed simulation process

1.9 Validity and reliability of data

According to Botswick and Kyte (1981:120-121), high reliability does not ascribe validity to results, but no results can be valid without reliability.

1.9.1 Validity

Validation and verification are crucial in developing a computer simulation model, particularly when it involves the safety of humans. Results obtained needed to remain constant for each execution.

A base model was developed and calibrated to show the area of study. The calibrated model was verified to confirm that the simulated output operated as close to the expected real system as possible once inputs were entered.

Validating a model as a final step compares results with historical data from the literature. In this instance, the prototypical research had limited access to historical data with which to compare results. Results, however, were to comply with what was expected to happen, based on what was known from the historical data by means of a protocol. This meant that validation of the model to establish credibility was vested in the model's ability to replicate precisely what the researcher proposed.

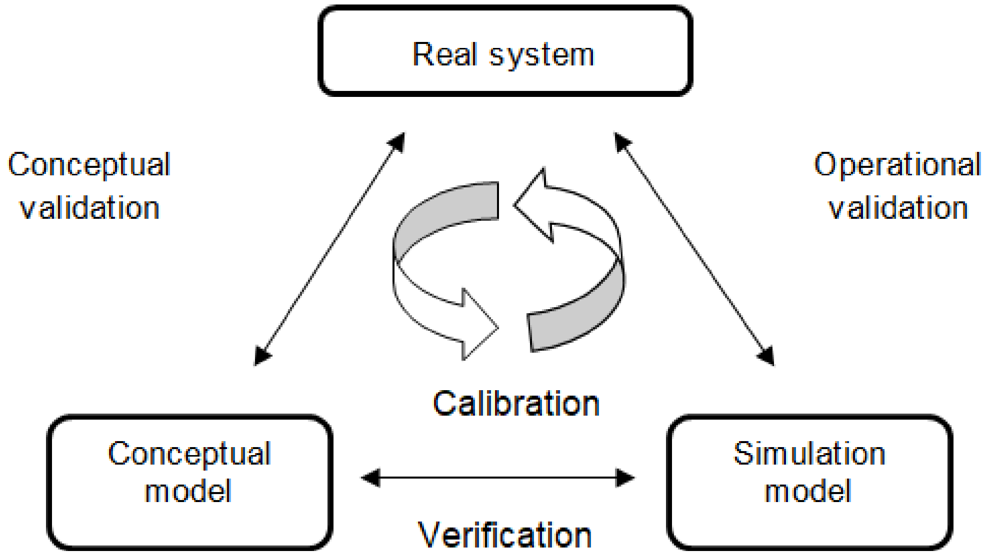


Figure 1.5: Conceptual and operational validation and verification in simulation (Source: Ulgen et al., 2014:11)

Figure 1.5 shows the relationship between the conceptual validation, operational validation, and verification in simulation. During this phase, the researcher applied building software techniques. A suitable approach to validation in this research was through construct validity, which uses data from a variety of sources and applies it to what is being measured and to "how and why it operates the way it does" (Botswick & Kyte, 1981:112). This infers that insight into the construct is important and that relationships between this and other constructs suggested by a theory or model needed to be identified (Delport, 2007).

1.9.2 Reliability of data

In quantitative research, reliability refers to whether a measurement procedure is stable or consistent. This means that a measuring instrument would produce consistent statistical results when applied, unless there is alteration in the variable being measured (Monette, Sullivan & De Jong, 2002). The reliability and trust people put in computer simulations depends on the validity of the simulation model. This is an indication that a distinct theoretical definition for the construct needed to be developed.

Simulation models can be used as a tool to verify engineering theories, but they are only valid if calibrated properly. Furthermore, there are inherent reliability qualities of packet switching, such as guaranteed data integrity via the 3-way handshake. In addition, when a unit fails, other units remain unaffected and the system adjusts itself to continue unabated (Cisco, 2011). The model was to be checked to ensure that it functioned adequately, as intended.

1.10 Delimitations

The study focused specifically on the routing aspect of autonomous vehicles, and factors that could enhance this facet were included in the spectrum of IoT. The researcher recognised the importance of other constituents of autonomous vehicle engineering in the form of radar technology gyroscopes, which are employed in AVs for detection and identification purposes, and power plant, but did not delve in-depth into these features. The following were excluded, as they were not required for the specific area under study:

- Physical construction of components
- Resources of energy to power the real-life system
- Finance and procurement processes

1.11 Ethics

Ethical considerations in this research engaged both the topic under study as well as the conduct of the researcher. Legislation governing the use of our roads does not include autonomous algorithm-encoded vehicles where the legal implications could well confront ethical implications in the event of an accident (Mathews, 2020). How would the legal implications correspond with ethical approval in the autonomous decision-making characteristics of autonomous vehicles? Vehicles may need to be programmed to obey traffic law and to practice independent lawful decisions in unforeseen situations, without compromising safety. According to a 2016 survey undertaken by the American Automobile Association, only 20% of Americans would trust an autonomous vehicle to transport them (Hsu, 2016).

In promoting well-being and minimising harm, car manufacturers are faced with a moral dilemma in certain situations. Should an AV end up killing a single adult passenger to swerve from hitting a group of children? Creating moral machines with universal ethical principles poses problems when the AV is confronted with a life-threatening situation and is programmed to respond to a set of principles. Consensus may be important for convincing people to accept AV driven vehicles together with specific programmed ethical principles, even though ethical preferences vary amongst cultures. Awad et al. (2018) found in their research strong preferences for sparing humans over animals, sparing more lives than that of perhaps a single person in a certain scenario, and sparing young lives. The authors opine that the three preferences make it at least possible to consider creating an important foundation for machine ethics by policymakers (Awad et al., 2018).

Permission to conduct this research was obtained from the Ethics Committee of the Faculty of Applied and Computer Sciences at the Vaal University of Technology. Research institutions and corporate bodies considered during the course of this research were not expected to be problematical in imparting information or to call for anonymity as they were overtly available to be accessed online or via the University's jurisdiction to scholastic and other materials. A situation did not arise at any time that necessitated measures of anonymity during the course of the research and the researcher had no need to obtain authorisation to gain access to relevant resources. Authorisation is the rights given to an individual to access a user or a program to conduct certain actions on a computer or system (Yu & Tsai, 2011; Regan, 2013; Zacker, 2014). Accuracy of reporting data remained foremost, and the researcher has recognised contributions of others to this research at all times through detailed referencing (Neuman, 2000).

1.12 Contribution of the research

Numerous research efforts have been conducted on a similar topic to the topic of this study, most of which specifically outlined dependency on various satellite navigation techniques with an emphasis on their significance in the all-round functionality of vehicles (Huang, Su & Yan, 2017; Eckelmann et al., 2017). These efforts point to a knowledge gap in a communication facet of the system that involves real-time and ubiquitous interconnectivity among vehicles by omitting the dependency on satellite navigational aids as a primary component for ascertaining the routing data of each unit.

Research studies contributed significantly to 'theory and practice' by providing avenues to understand methodology and innovation (Hofstee, 2009; Jansen, 2012), and were accessed by the researcher in the process of data collection. Presented with the limited empirical research investigating the concept of packet switching as a routing mechanism for autonomous vehicles, this study proposed the consideration of an **alternative** routing mechanism **to reduce** the **number of sensors** needed and **total dependency on the GPS satellite navigational aid** as a primary component of interconnectivity between vehicles. The researcher, therefore, aimed to contribute to the prevailing knowledge, and to pave the way for further research by suggesting a novel, alternative method to resolve the real-world problem of a progressively challenging traffic congestion issue, thus benefitting the automotive industry, organisations, academia and society.

1.13 Chapter outline

The research was sanctioned by the Faculty of Applied and Computer Sciences at the Vaal University of Technology to commence and the researcher proceeded to develop the model to ensure completion within the period set aside for this study. The content of each chapter is summarised in Table 1.1.

Table 1.1: Outline of chapters

Chapter 1: **Introduction**	In Chapter 1, the researcher explains the rationale and motive for the research. A brief introduction to the background to autonomous vehicles and the packet switching concept is provided. The problem statement, research questions, research objectives, research design, delineations, overview of the methodology and ethical considerations are dealt with.
Chapter 2: **Literature Review**	In the literature review, the researcher presents an overview of the workings of packet switching, describing challenges and various security technologies that would affect incorporating packet switching as a routing instrument into the design of autonomous vehicles.
Chapter 3: **Research Methodology**	In the chapter dealing with the research methodology, the researcher describes artefacts required for the successful design of the model, with a conceptual bearing on real-life implementation of the concept as a routing mechanism. Challenges such as security are also addressed.
Chapter 4: **Simulation and Data Analysis**	In the simulation and data analysis chapter, the researcher demonstrates model simulation and presents an analytical and systematic discussion of the various components, as well as and how software was adapted to develop the new algorithm.
Chapter 5: **Model Interpretation**	An overview of the mechanics of the model is presented by the researcher in this chapter, clarifying the virtual system of routing AVs (virtual aspect) with the real-life implementation.
Chapter 6: **Findings, Recommendations and Conclusion**	In this concluding chapter, the researcher addresses the research objectives and subsequent findings, discusses the limitations of the research and the merits of a newly constructed prototype, suggesting future research possibilities.

1.14 Summary

Background and justification for this research were provided in this chapter, together with a summary of the design and methodology employed. Delimitations, ethical concerns and contributions of the research to the existing body of knowledge are also deliberated.

In the next chapter (Chapter 2), the researcher will offer insights into the literature available dealing with packet switching and autonomous vehicle research, highlighting advancements in routing mechanisms and vehicular networks that converge within the omnipresent influence of IoT.

CHAPTER 2: LITERATURE REVIEW

2.1 Background

The notion of autonomous vehicles stems from as early as 1925 in New York where a vehicle was equipped with a transmitting antenna and operated by a second vehicle (Google News Archive, 1926). Later, in 1937, General Motors added electromagnetic fields embedded in the roadway to a similar vehicle (O'Toole, 2009), yet efforts to develop and implement a fully automated, functioning system of automated vehicles has been slow. This is primarily due to a number of complexities and variables that must be considered when coalescing such a system into contemporary roads and guideways. The first major breakthrough in the advent of a commercialised, legitimate vehicle that offered a certain level of automation, thereby enabling occasional driver intervention while in transit for on-road vehicles, only took form in 2013. This prompted four levels of driving automation determined by the Germany Federal Highway Research Institute and adopted by the US Department of Transportation's National Highway Traffic Safety Administration (NHTSA, 2013; Marinik et al., 2014). In 2014, the Society of Automotive Engineers International (SAEI, 2018) released six levels of autonomous vehicles, which was updated in 2016 and again adopted by the US Department of Transportation. A Level 5 (L5) autonomous vehicle encompasses full self-driving automation that performs all safety-critical driving functions and monitors all roadway conditions at any speed and in all weather conditions. The driver provides destination input and need not occupy the vehicle as control rests solely with the automated vehicle system. L6 is an automated driving system that monitors the entire driving environment.

The automotive industry is a critical component of the world economy, employing millions of workers across manufacturers, suppliers and dealers, with mobility becoming increasingly expensive and inefficient. The running costs of owning a vehicle, the costs of roadway construction and maintenance, and the cost of the average person spending 260 hours lost time in the driver's seat, all amount to astronomical costs. In urban areas, 40% of fuel is used in looking for parking, and in some US cities parking lots cover over a third of the land area (AAAFTS, 2018). Reports compiled from data sourced from 180 countries by the Global Status Report on Road Safety (WHO, 2018) revealed that the global traffic accident death rate averaged 1.35 million people annually, while the main cause of death for the age group 5-29 years was as a result of traffic injuries. The National Highway Traffic Safety Administration (NHTSA) administrator, Mark Rosakind, reported that human error and negative human behaviour, such as drunken driving and drowsiness, were responsible for 94% of accidents. The Road Traffic Management Corporation (RTMC) reported in 2016 that South Africa recorded the highest road fatalities in Africa, with figures depicted in Figure 2.1, which, according to 2019 statistics, was a record still upheld (Ngquakamba, 2019). While other fatality and accident data on the Internet may be more recent, the researcher deliberately chose the illustrated official figures. The motive for this took into account that later figures released do not reflect the historic normal scenario given the disruption of travel restrictions in an effort to prevent the spread of the Covid-19 pandemic enforced from March 2020 (Wiysonge, 2020). With relevance to this research, the automation of vehicles is intended to reduce traffic congestion, free up the hours spent by motorists on the road and increase work performance, with consequent economic benefits and a better quality of life.

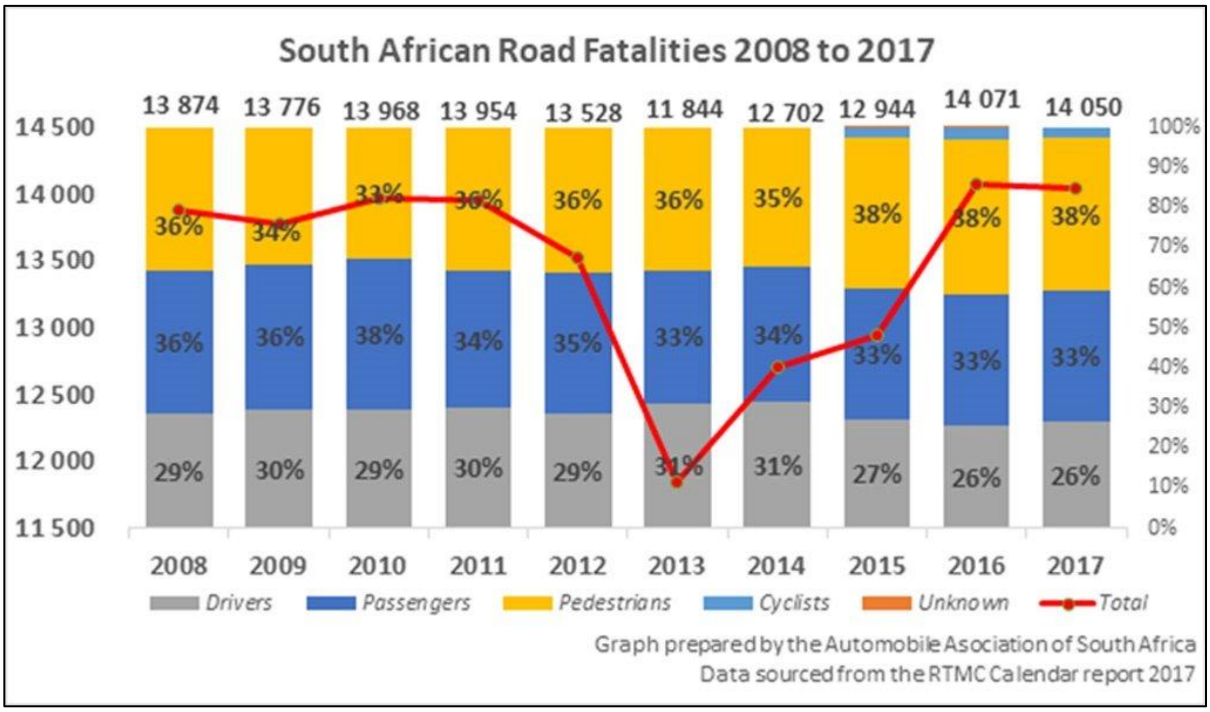


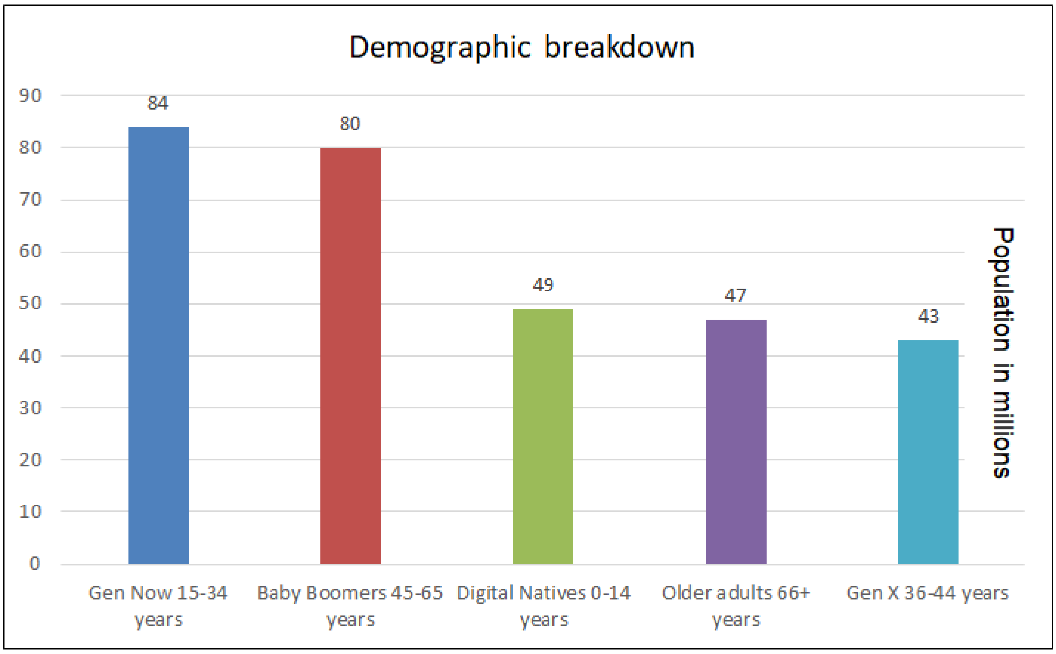
Figure 2.1: Graphical representation of South African road fatalities 2008 to 2017 (Source: Automobile Association, 2018)

A considerable number of studies have been conducted on aspects of vehicle automation, but most focus on the automation of components used by traditional vehicles and carry with them expensive and cumbersome equipment, which usually require a direct line of sight to nearby satellites. Developments have however emerged in the wake of full automation, the most prominent of which are designs by Tesla's Motors and Google's self-driving car (Simonite, 2016). These prototypes each host a number of interconnecting components, ensuring reliable operability of their respective vehicles. The vehicles operate using similar principles and both are currently being tested on public roads, presenting numerous challenges, the foremost challenge being the presence of humans and their behaviour, which impetrates the need for sound predictability within each vehicle's immediate area. To solve this problem, a computer programme was conceived using a technique known as 'machine learning'. Machine learning (ML) is the process of recording and analysing real-time data by way of digital imagery. The computer recollects this data, based on certain actions and scenarios that are encountered, and acts accordingly, identifying certain trends by applying complex algorithms that can be used to better interpret a vehicle’s surroundings and improve its capacity for predictability (Hurwitz & Kirsch, 2018).

The above prototypes are considered reliable to a large degree; however, they still present several drawbacks. Drawbacks include an immense expense due to the calibre of engineering required, as well as the computer requirements and the on-board navigational equipment, particularly the LiDAR system in use (Schippers & Socash, 2017). Of further concern is the security of the system from attack by hackers. Such attacks could involve a driver’s personal information being intercepted by hackers, or infiltration of a more serious nature, with devastating consequences. In addition, on-board systems have a strong reliance on satellite navigation, which can be problematic when operating in various weather conditions. The inability of sensors to navigate in extreme weather is one of the biggest obstacles to the development of self-driving technology at present, as sensors have been known to become completely inactive in snow blizzards, in heavy rainfall conditions and in dust storms (Berg, 2018). An additional reliability factor also evident with GPSs is that they are subject to occasional error and may for example direct vehicles through a road block (Hanley, 2016).

Departments of Transportation (DoTs) in the individual states of the USA and elsewhere have shown little interest to invest in any form of roadway infrastructure adaptation to modify road facilities in order to support automated vehicles by providing the road environment with, for example, sensors along segregated lanes (Marinik et al., 2014; Wahl, 2013). This was evident even though the NHTSA’s Associate Administrator for Vehicle Safety in the United States determined in a 2012 report that there was an urgent need to develop autonomous vehicles. A survey by the NHTSA administration determined that driving demographics had changed over the years as attitudes towards driving had changed, as demonstrated in Figure 2.2.

Younger generations are more console and smartphone conscious than the baby boomer generation born between 1946 and 1965, whose greatest goal was obtaining a drivers licence on their eighteenth (or sixteenth in some countries) birthday. This was also evident in statistics involving fatalities due to mobile phone distractions, including texting behind the wheel (CAR, 2014). Taking the aforementioned into account, autonomous vehicles are destined to become a reality to increase safety and enhance mobility to those who are unable to drive (Google, 2014). Despite this, among the issues that Google has had to consider is how to counter negative public perception of autonomous vehicles. The transition of transferring full autonomy from the driver to the vehicle is expected to impact dramatically on people’s lives (CAR, 2014).



- Population untapped due to below driving age
- A percentage of older adults are driving impaired. Autonomous vehicles will provide added mobility
- A percentage of baby boomers will enter older adults' category when autonomous vehicles enter market

Figure 2.2: Demographic breakdown by the Centre for Automotive Research (Source: CAR, 2014)

The reality exists that certain human behaviours cannot be changed and security loopholes present in any technical system will be found, with the possibility that autonomous cars can be hacked, which is another problem that Google must confront. With one of the best cyber-security teams in the software industry, Google has already tapped into their own automated systems, having accessed through the entertainment systems (Greenemeier, 2016). According to Greenemeier (2016), it may be difficult to intercept communication signals from both the downlink and uplink of a satellite, as the frequency, modulation and method of encryption would have to be known; yet, it is still possible to achieve this.

Traffic congestion and routing in the virtual environment of packet switching has received consideration by some researchers including the author (Wahl, 2013), but little research has been conducted on the routing of autonomous vehicles within a real-life transport network. For the literature review, the researcher investigated how research has developed to overcome some of the above concerns with the aim of improving reliability of autonomous vehicles by means of various routing mechanisms. Careful consideration was given to literature that focuses on computer network transmission routing protocols in the form of packet switching as a means of routing autonomous vehicles within a decentralised system that is economically sustainable. A routing protocol is a path selection procedure that suggests how networks communicate amongst each other by disseminating information that enables them to choose routes along which traffic can be sent, via routing algorithms between two nodes on a network (CCNA, 2018).

Mechanisms that enhance the execution of the above, such as Artificial Intelligence (AI), machine learning (ML), and the communication technologies within Intelligent Transportation Systems (ITS) aligned to the Internet of Things (IoT) are also discussed in this chapter to give the reader clarity on each of these aspects.

2.2 Technologies encompassing packet switching

Packet switching is a network communications system invented by Paul Baran in the 1960s. It is a mode of data transference that consists of networking equipment such as a medium, routers, switches and nodes (end users), which collectively distribute data to predetermined receiving nodes (Curtis & Taylor, 2004). A packet is considered a simple communication unit that can be transmitted across one or several digital networks simultaneously. To enable the virtual packets of data to be conveyed through a multi-tasking access medium (cable), they are broken up into tiny similar structures that queue up to be distributed and then reconstruct themselves on reaching their target node destinations (Sanders, 2007). A binary algorithm converted into bits using routers and switches ensures reliable delivery of individual packets (Cioara, Minutella & Stevenson, 2008).

A packet-switched network is therefore an interconnected system transmitting multiple data requests over a medium and connected by routers. A router is a switch that separates broadcast domains and can transmit packets across different types of data link layer networks such as the Internet. IP routing is the process of sending packets from one host on a network to a host on another network. The router analyses each incoming data packet's nominated network address, compares it to a selection of network addresses stored in the routing table or routing information base (RIB) and forwarding information base (FIB) to identify the outbound port for delivery. The procedure ensures that optimal ports are selected for a packet on its path to the destination, with the most common strategy being to use the shortest path to the metric, depending on the configured protocol. Networks have inherent sizes, so routers receive fragmented packets in the form of bits to fit within data frames, which reassemble again once they arrive at their respective destinations (Cioara, Minutella & Stevenson, 2008).

Data frames are allotted a source and destination media access control (MAC) address, which is a unique address hard-coded into every network interface card (NIC), which uses an address resolution protocol (ARP) to find the MAC addresses of other devices (Sanders, 2007). Switches, by definition, are networking devices that will route data frames (not packets) to their relevant destination MAC addresses, and which will issue ARP requests to obtain MAC addresses from connected devices and, subsequently, build a routing table comprising MAC address – port number mapping. Metadata informs each router/switch en route to the destination and a router momentarily stores a packet, prior to forwarding, in order to read its information. To avoid this holdup, performance is enhanced using multilayer switching techniques, with various tasks allocated in each layer, as shown by two network models in Table 2.1. The OSI protocol consists of seven layers, while the more refined transmission control protocol (TCP), designed for Internet use, utilises only four layers. Routers drop or eliminate packets in an overflow situation or if a packet's network address does not correspond with the one in the routing table; but end systems run by TCP, which is connection-orientated, will determine that packets have been lost (Cioara, Minutella & Stevenson, 2008).

Table 2.1: Comparative hierarchical functions of seven-layer Open Systems Interconnection (OSI) model and four-layer Transmission Control Protocol model (Cioara, Minutella & Stevenson, 2008)

OSI MODEL	TCP MODEL
Application Layer 7 Means to access network resources	**Application Layer 4** Where all higher protocols, such as SMTP, HTTP, SSH, TFTP operate Processes are addressed via ports that represent services
Presentation Layer 6 Encodes and decodes data into readable material according to protocol	
Session Layer 5 Establishes, manages and terminates connection among communicating devices	
Transport Layer 4 Provides reliable transport services to lower layers	**Transport Layer 3** Establishes connection and reliable data transmission; main protocols UDP & TCP
Network Layer 3 Routes data via routers between physical networks; handles packet segmentation	**Internet Layer 2** Routes IP datagrams containing source and destination address
Datalink Layer 2 Means of transporting data across physical network; identifies physical devices such as bridges and switches that operate at this level	**Network Access Layer 1** Medium through which data are sent through network including means of how bits are transmitted, such as optical fibre or copper wire
Physical Layer 1 Medium through which data are transferred	

A connection-orientated layer is a virtual network circuit or switching function between sender and receiver that allows packets to flow, with routers forwarding packets along the best path to the destination. This means the transport layer can only request a transmission once the network has been informed to set up a connection. Packets can be forwarded independently of other packets in a connectionless protocol model, as shown in Figure 2.3. An example of this is IP or user datagram protocol (UDP) that does not require the establishment of a network to begin transmission, and which is less reliable than a connection-oriented protocol. Packets arrive randomly without following the same route so connection can be faster although traffic congestion is high (Dias, 2015; Sheldon, 2001).

TCP is a connection-oriented protocol, depicted in Figure 2.4, where the recipient confirms receipt of the transmission and the sender keeps track. The connection is maintained and, therefore, this protocol is suitable for long and steady communication. Packets travel in sequence and follow the same route. There is an initial delay in transfer of information until connection is established, allowing for faster delivery. Reliable transfer of data are ensured using a 'handshake' process where a connection agreement packet is sent from the sending node to the receiving node, which accepts the request to connect and relays a packet back to the sending node to determine the rules of the agreement. A final acknowledging packet is sent to the receiving node and a connection is opened to allow tagged packets to be transferred in sequence for analysis by the receiving end-device, which, in turn, issues a response packet requesting another data packet to be sent in sequence (Cisco, 2011).

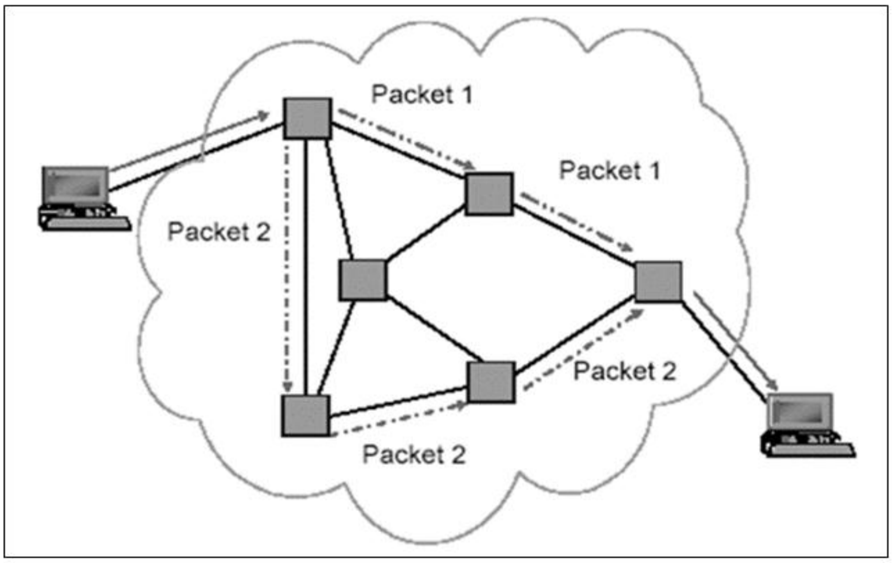


Figure 2.3: Datagram connectionless packet switching (Source: Dias, 2015:17)

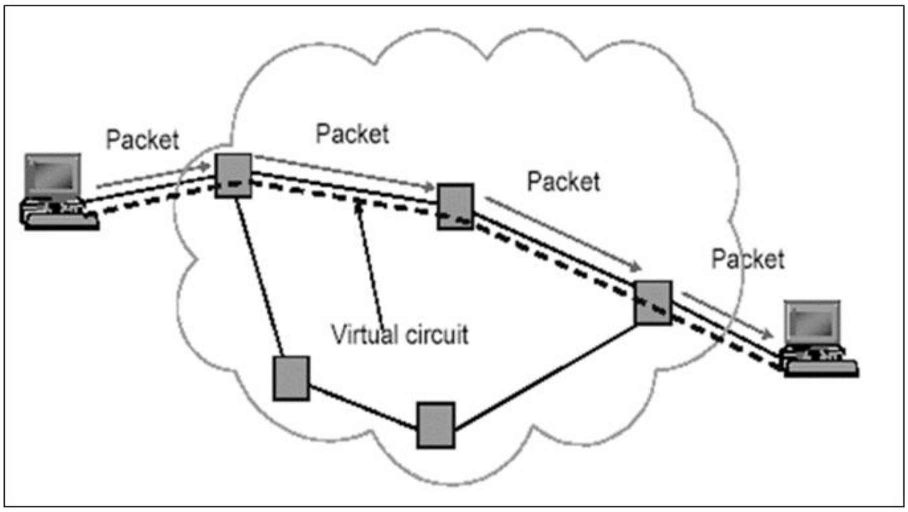


Figure 2.4: Virtual circuit connection-oriented packet switching (Source: Dias, 2015:21)

Packet collisions do occasionally occur in packet switched networks, which means that packets must be resent. This disrupts network efficiency, necessitating modifications in order to safeguard vehicles from collision. Collision and broadcast domains are separated by routers with switching and routing abilities that can restrict access to specific incoming and outgoing traffic, allowing port access to specific packets, as shown in Figure 2.5. These routers and switches function in the network layer (Layer 3) of the OSI model and Internet layer (Layer 2) of the TCP model respectively, and can distribute data to remote networks. They are also suitable for wide area network (WAN) connectivity used in vast communications networks (Cioara, Minutella & Stevenson, 2008:748).

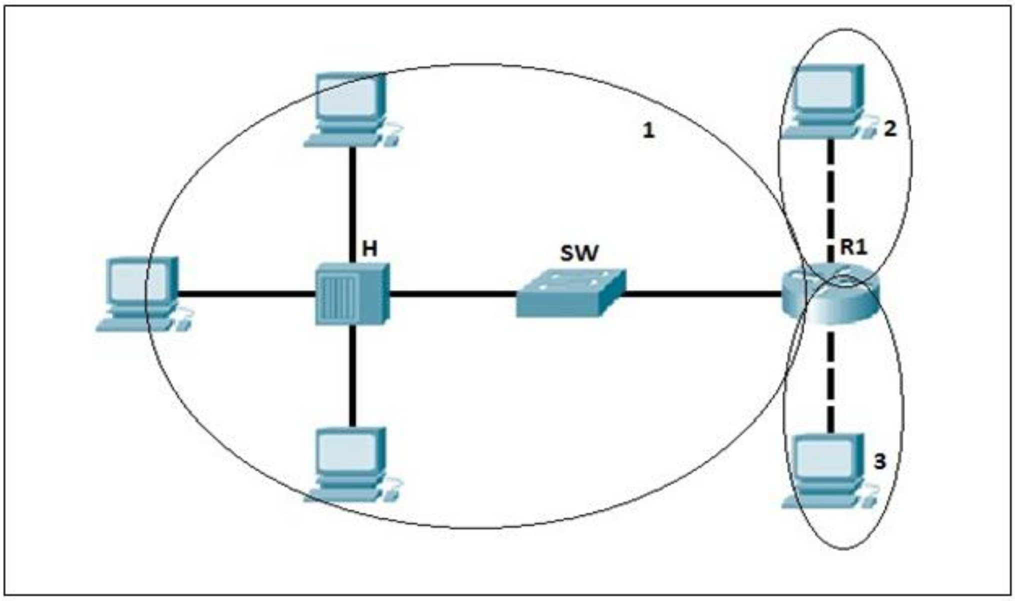


Figure 2.5: Ports on hub or switch in the same broadcast domain and ports on router (Source: Cioara, Minutella & Stevenson, 2008:748)

Routers do not retain information to enable tracking of packet delivery or detect errors; they operate routing protocols and, using processors, they collect, store and assemble information about routes in the network, calculate routes and construct routing tables. This information is all stored in random access memory (RAM) to allow the protocol that runs in the router to perform route selection when forwarding packets. RAM is the main memory in a computer used to temporary store information to allow for fast access. Examples of routing procedures or protocols are routing information protocol (RIP), open shortest path first (OSPF), and enhanced interior gateway routing protocol (EIGRP). Interior gateway protocols (IGP) such as OSPF or intermediate system-intermediate system (IS-IS) are used to exchange routing information between routers within an autonomous system (AS), whereas exterior gateway protocols (EGP) such as border gateway protocol (BGP) are used to communicate routing information between autonomous systems (Cisco, 2011).

The routing protocol allows routers to screen packet information and discard any packet that does not match a series of network addresses. A datagram network's routing table contains a destination address + next hop information, whereas a virtual circuit (VC) contains incoming VCI + outgoing VCI + out-going port #. A router identifies the path a packet should follow by assessing the content on a routing table as follows:

- *Prefix length* is represented by the number of binary bits in the *on* position
- *Administrative distance* is the preferred shortest path to a destination chosen as each routing protocol receives routing information and updates
- *Metric*s is the unit of measure employed by the protocol to calculate the best path (Edgeworth, Foss & Rios, 2015)

Vital components of routers are buffers, which are memory regions that temporarily retain packets until they are processed. A buffer is needed when a router's capacity to process incoming packets is reached and the number of incoming packets exceeds the number of outgoing packets (section 4.4). Each crossing point, or interface, consists of input and output buffers. Buffers need to be large enough to hold network traffic so that it does not flood into the router faster than the traffic can be processed by the router. Limited numbers of buffers also mean that they are soon filled and incoming packets will be dropped. Dropped packets need to be retransmitted, affecting network performance as congestion worsens. Alternatively, too many buffers can cause serious delays, so buffers and queues should be effectively manipulated to ensure smooth and fast delivery. Current multiport routers use application-specific integrated circuits (ASICs), a Layer 3 switch that can forward up to 50 million packets per second. In order to build scalable networks, a router-based interconnecting network is required. Figure 2.6 illustrates the route determined in a virtual circuit connection, where switches contain tables that operate forwarding to various numbered virtual circuit identifiers (VCIs). Figure 2.7 shows the associated routing table for Figure 2.6. The diagram shows three connections:

1. Solid line: A 1 3 6 B with local VCIs 1, 2, 7, 8
2. Dotted line: A…1…3…4…5…D with local VCIs 5, 3, 4, 5, 2
3. Dashed line: C --- 2--- 4--- 3--- 6 B with local VCIs 6, 3, 2, 1, 5

Local VCIs in favour of global VCIs are used, as they are more readily available and have more connections.

Link 1—3 is shared by connection 1 and 2

Link 3—4 is shared by connection 2 and 3

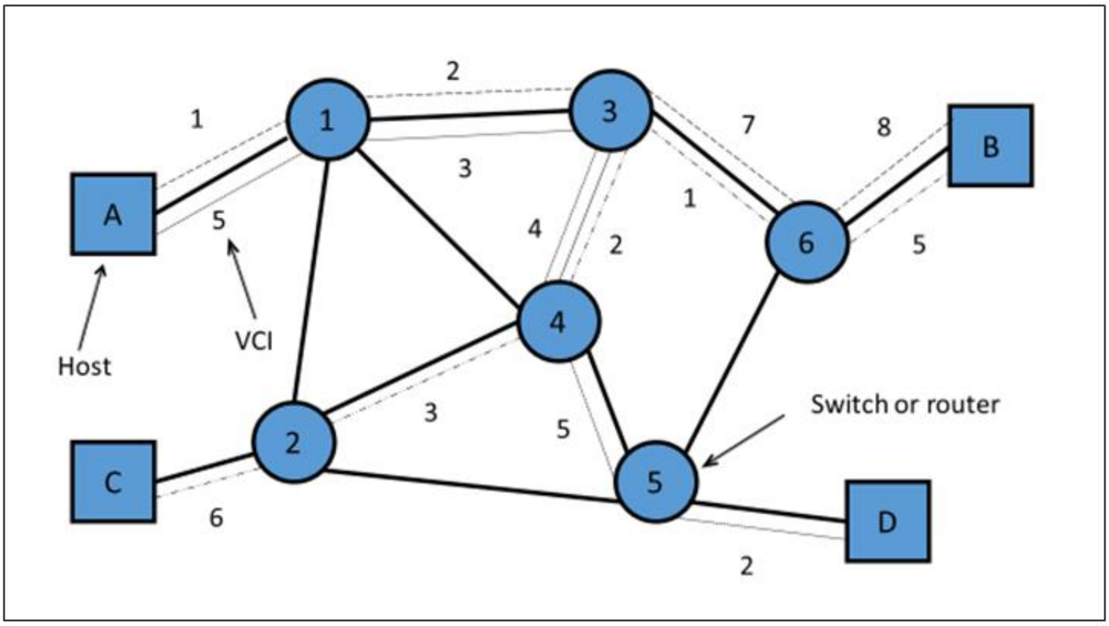


Figure 2.6: Virtual circuit packet network
(Source: Dias, 2015:56)

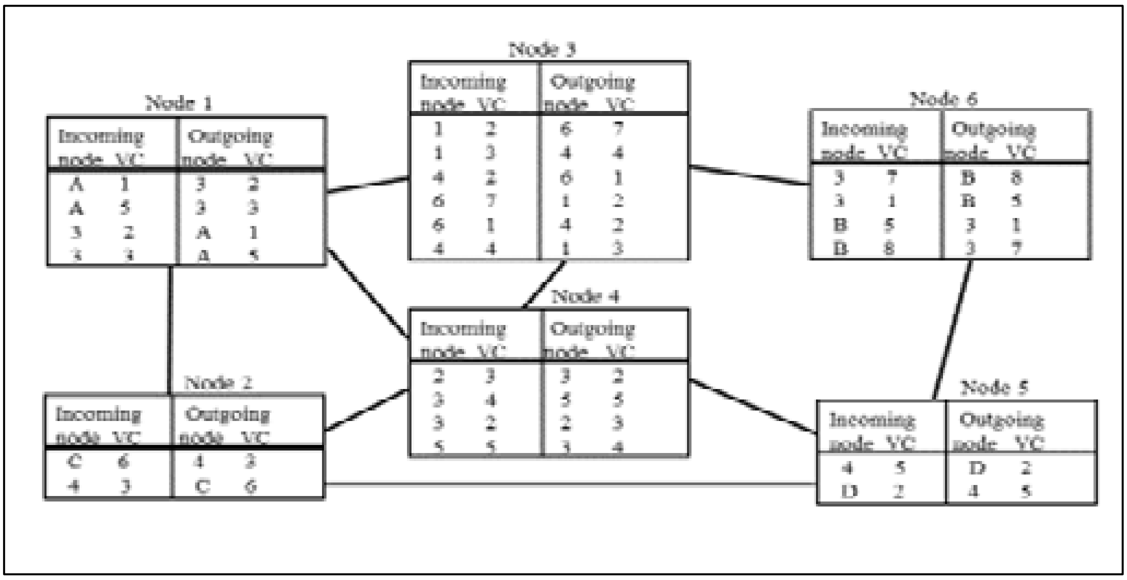


Figure 2.7: Routing table for virtual-circuit packet switching network
(Source: Dias, 2015:57)

2.2.1 Packet switching mechanisms used by routers

While routing is the procedure of choosing a path to navigate traffic, such as network packets in a network or across multiple networks, packet switching mechanisms are techniques used to forward packets and include the following:

2.2.1.1 Process switching/software switching

Process switching/software switching is per packet-based and can route traffic across parallel paths to a destination. This approach to switching is the slowest forwarding mechanism and packets do not arrive in sequence due to latency in the routing process, and therefore, system performance is reduced.

Figure 2.8 shows how the IP-input process obtains the next-hop router's IP address, outgoing interface and MAC address from the ARP table and the RIB. The next destination MAC address of the packet is overwritten with the next-hop router's MAC address. The source MAC address is also overwritten with the MAC address of the outgoing Layer 3 interface. The IP time-to-live (TTL) field is decremented, the IP header is recalculated, and the packet is conveyed to the next-hop router. The ARP table contains information obtained from the ARP protocol used by IP hosts to learn the MAC addresses of other IP hosts on the subnet dynamically. RIB is created from data obtained from the dynamic routing protocol, static routes and directly connected routes (Edgeworth, Foss & Rios, 2015).

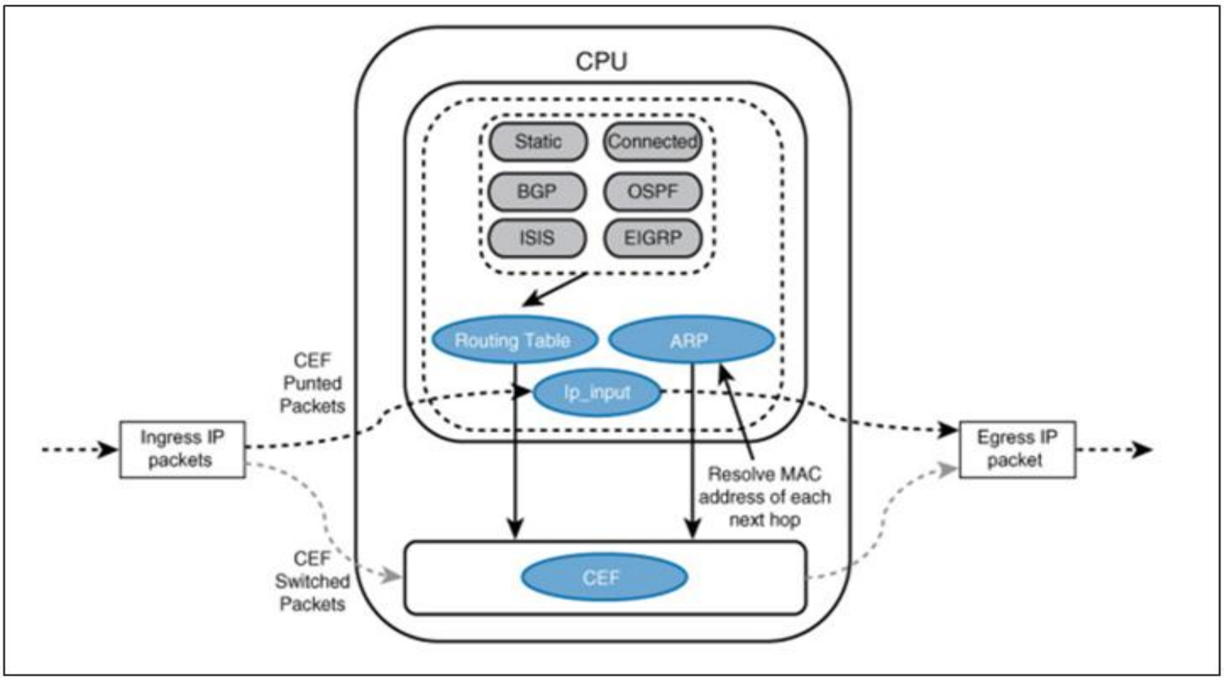


Figure 2.8: Process switching flow
(Source: Edgeworth, Foss & Rios, 2015: Chapter 3)

2.2.1.2 Interrupt context switching

Interrupt context switching maintains a cache, which stores the first packet's information and bypasses the processor to complete the task of forwarding the packets that match that packet to their destination. This saves on unnecessary processing time and is, therefore, faster than process switching. Interrupt switching describes various switching methods, namely: **i) fast switching; ii) optimum switching;** and **iii) Cisco Express Forwarding (CEF).** Figure 2.9 illustrates this concept as follows:

- The interface processor detects a packet and forwards it to the input/output memory
- A receive interrupt is generated by the interface processor, while the central processor determines the packet type (IP) and starts to switch the packet
- The processor searches the route cache for the destination, output interface, next hop and conversion of MAC address, before applying the information

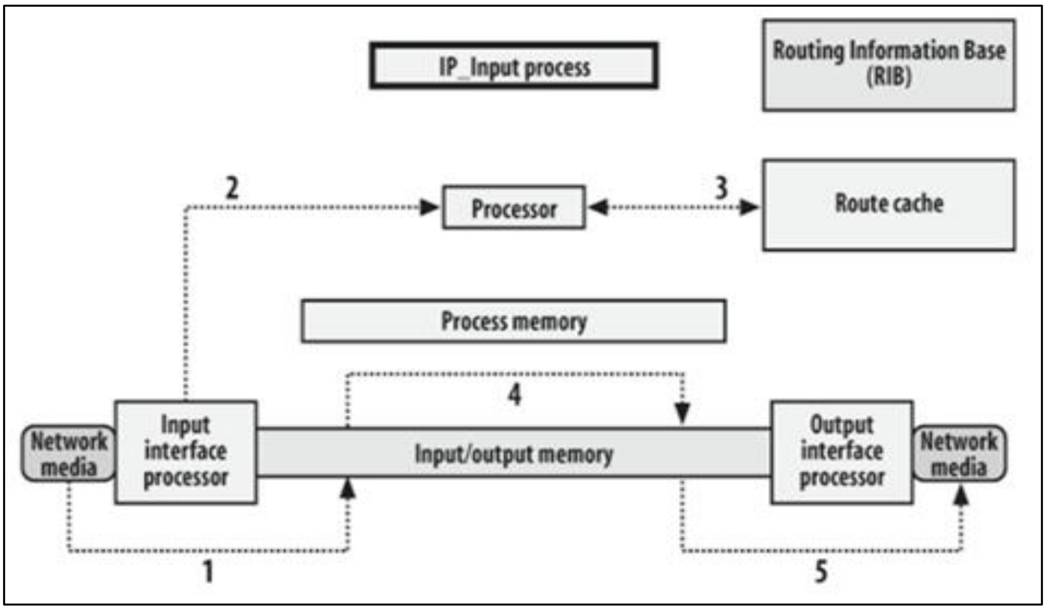


Figure 2.9: Interrupt context switching
(Source: Edgeworth, Foss & Rios, 2015: Chapter 4)

i) **Fast switching** stores information on how to reach a destination in a fast-switching cache once a packet has been forwarded. This allows subsequent packets heading in the same direction to re-use the next stop information from the cache. The MAC address is written in a binary tree format to record and retrieve information. Figure 2.10 shows a binary tree structure starting at root 0 with the next lower branch containing 0 and a 1 above. Each branch affixes 0 or 1, depending on the previous level's value. Fast switching has depreciated in newer internetwork operating system (IOS) releases.

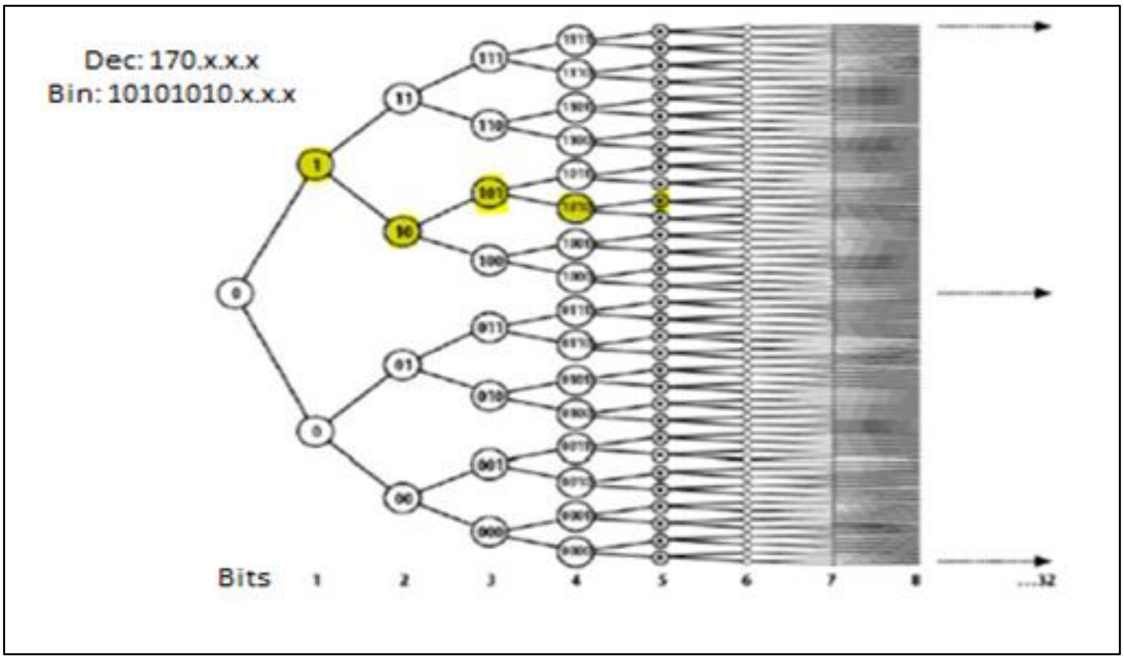


Figure 2.10: Structure of a binary tree format
(Source: Stringfield, White & McKee, 2007:71)

ii) Optimum switching – similar to fast switching, optimum switching differs from the tree structure by using a multiway format, as depicted in Figure 2.11, to record and retrieve information in the route cache, and its prospective size is limitless.

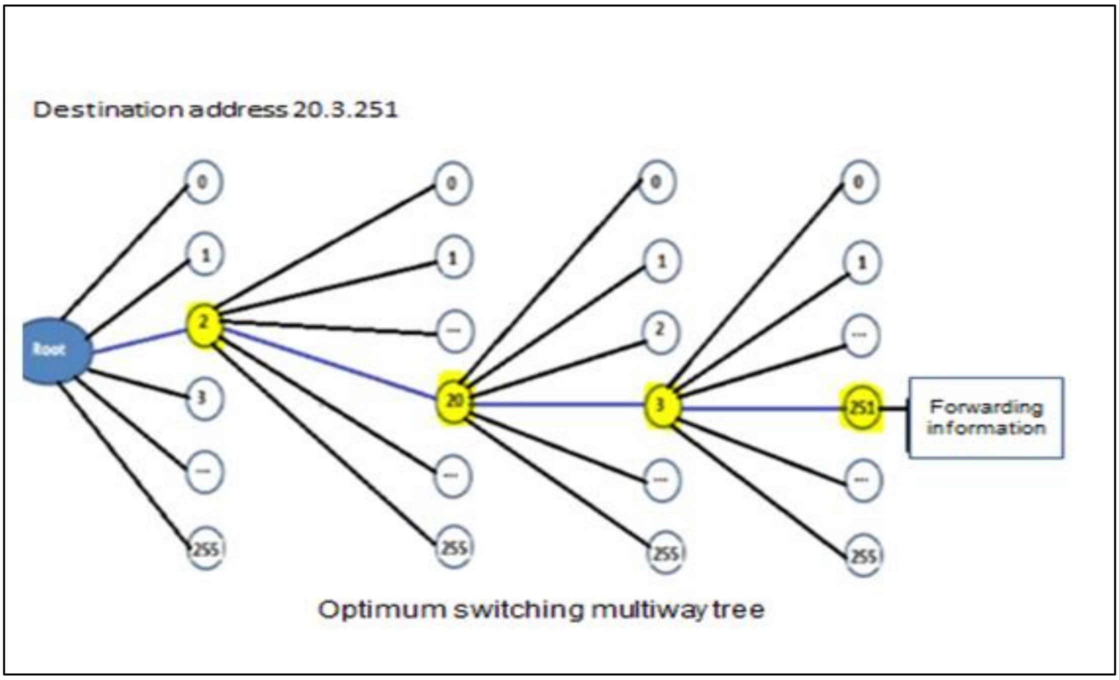


Figure 2.11: Optimum switching multiway tree
(Stringfield, White & McKee, 2007:72)

iii) Cisco Express Forwarding (CEF) – optimises the router to enable faster forwarding of more packets via the Forwarding Information Base (FIB) and Adjacency Information Base (AIB). The FIB consists of pre-calculated reverse lookups, next hop information for routes, the interface, and Layer 2 (L2) information. They are used by both *software-based* and *hardware-based* routers and perform similar functions (Stringfield, White & McKee, 2007:71).

a) *The Software CEF/Forwarding Information Base,* constructed directly from the routing table, contains next-hop IP addresses for each destination. Topology or routing changes are reflected in the FIB and used by the CEF to make switching decisions. The AIB contains MAC addresses and outgoing interfaces of all directly connected next hops. It also contains data from the ARP table and L2 protocol tables (Edgeworth, Foss & Rios, 2015). Figure 2.12 illustrates how a packet is switched by the CEF through a router once the CEF table has been built from the routing table and ARP table. An IP packet with a valid FIB and adjacency entry will have the router overwrite the destination MAC address with the next-hop router's MAC address. The source MAC address will be overwritten with the MAC address of the outgoing Layer 3 interface, the IP TTL field is decremented, the IP header is recalculated and the packet is delivered to the next-hop router.

b) *The Hardware CEF* may allow for a higher packet rate but is an expensive process to construct and can be limiting, as they are set to perform specific tasks. Software CEF in hardware-based platforms is used to programme the hardware CEF and not for packet switching as in software-based platforms.

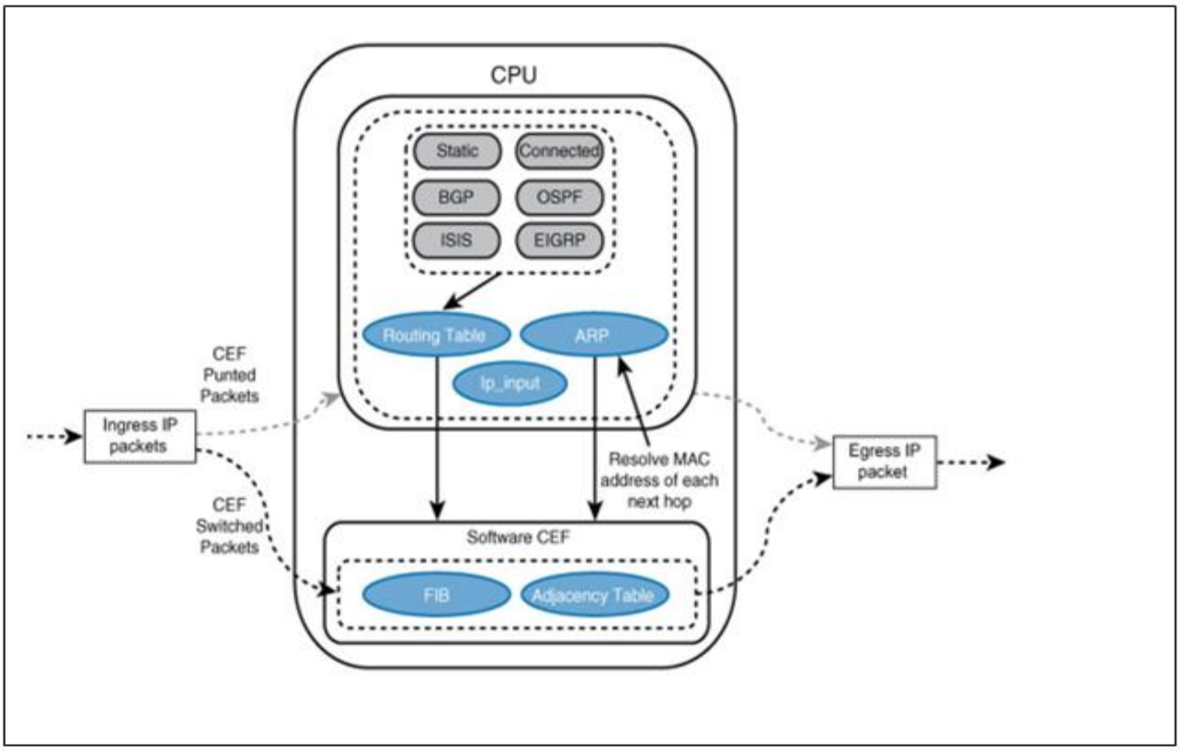


Figure 2.12: CEF switching flow
(Source: Edgeworth, Foss & Rios, 2015: Chapter 3)

2.2.1.3 Network load balancing

Load balancing is a technique used to dispense the workload over multiple paths or multiple processors in order to reduce congestion in the network. Load balancing algorithms described are weighted balance, priority, overflow, persistence and least used lowest latency.

Primary and alternative paths are present in a network and when it becomes congested, the secondary path is used. Figure 2.13 illustrates the basic load-balancing algorithm showing how the network balances the load on the primary and secondary paths.

P1 and P2 represent Path 1 and Path 2, respectively. S is the source node and D is the destination node. The total number of routers between source and destination is R. The number of routers in path1 and path2 are assumed to be R1 and R2 respectively. N is the total number of packets at the source at time T0.

$$R = (\Sigma R1) + (\Sigma R2)$$

Z = N/R

If P1 hops (S, D) > P2 hops (S, D)

Then P1 sends (Z* Σ R2) no. of packets

Alternatively:

If P2 hops (S, D) > P1 hops (S, D)

Then P2 sends (Z* Σ R1) no. of packets

Otherwise:

Both P1 and P2 send N/2 no. of packets

Suppose source S has 90 packets at time T0, and then it will send 60 packets from path 2 and 30 packets from path1.

Then according to the above formula: If p1 hops > p2 hops to S, D

Then P1 sends (Z* Σ R2) number of packets

So 90/6*4 = 60 through p1 and therefore 30 will go through path 2

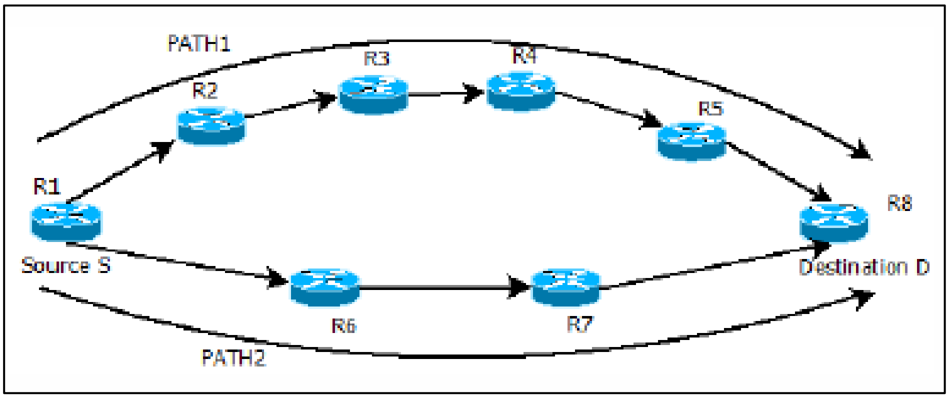


Figure 2.13: Algorithm and relative figure for load balancing approach (Source: Devikar, Patil & Chandraprakash, 2016:425)

Dijikstra algorithms used for source to destination shortest path selection may be inefficient in the presence of long delays and may be unable to solve the end-to-end delay-partitioning problem. This might call for an alternative routing algorithm to be considered.

2.2.2 Router operating stages

In order to comprehend the concept of routing in its entirety, a brief description of the operating stages of routing are described below:

i) *Control plane* consists of the dynamic routing protocols, the RIB and updates. It is the brain of the router responsible for maintaining and exchanging protocol information.

ii) *Data plane* is responsible for the switching process of packets through the router.

iii) *Management plane* is utilised to manage a device through its connection to the network (Edgeworth, Foss & Rios, 2015).

The layered architecture exists in IOS XR, which is an internetwork operating system train of the various Cisco systems employed, considered more reliable than the massive IOSs, where failure in one area can cause the entire system to become inoperative. Figure 2.14 depicts the three operating planes. The processes remain independent from each other in IOS XR, where failure of one does not interfere with other processes and, therefore, creates a more reliable model.

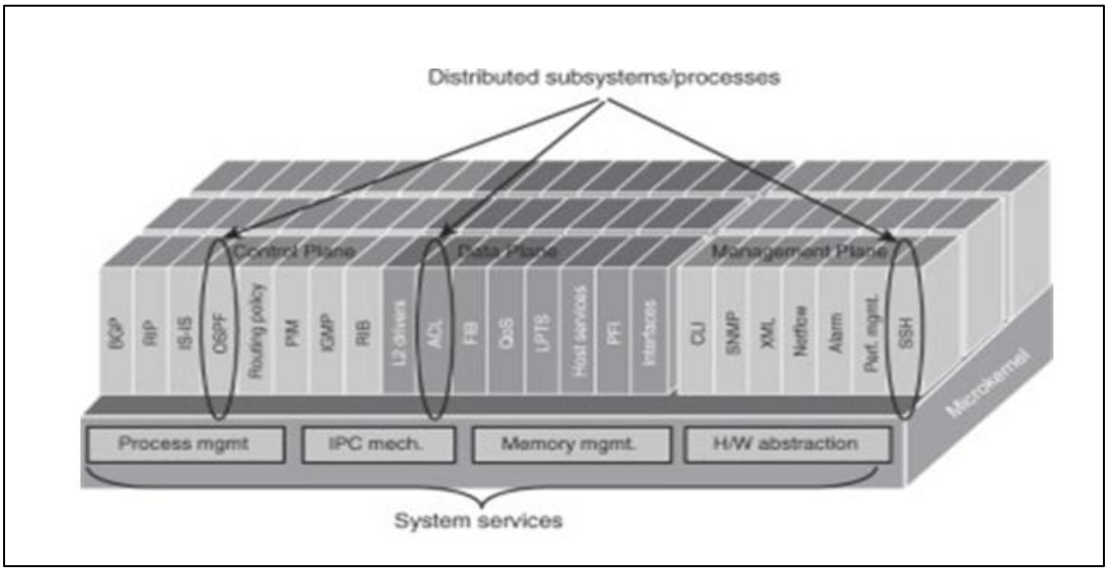


Figure 2.14: System services
(Source: Edgeworth, Foss & Rios, 2015: Chapter 4)

2.2.3 Routing mechanisms

For reliable and effective communication to take place between nodes within a network, a routing protocol is required. In order for the router to populate its routing table, a combination of the following basic routing methods is used:

- *Directly connected interface* from the router interface to other networks or subnets, configured to be recognised by the routers so that traffic is forwarded without assistance from routing protocols (Osterloh, 2001)
- *Static routing* – network routes that have been manually entered into the router's route table, defining the IP address of the next hop router when forwarding traffic to the destination. This route does not adjust to changes in the network because of its static nature and, should the interface become unavailable or fail, so does the route to the destination become unavailable or fails (Osterloh, 2001)
- *Default routing* – static routes that are used to allocate a route to an unknown destination and do not have to be configured on a router because the router will be familiar with the route to the destination by consulting its route table. Default routing, however, provides end hosts with a means to exit their local subnet and routers as a last resort if no other route exists in the route table

- *Dynamic routing* discovers routes automatically on a network and adjusts to network topologies. If a router attachment fails or becomes congested, the protocol within the router informs other routers of the change and initiates a routing algorithm to recalculate the network routes and to update the routing tables, saving on time and expenditure (Osterloh, 2001). Dynamic routing dominates most systems, including the Internet; and examples of protocols and algorithms include RIP, OSPF and EIGRP

A disadvantage of dynamic routing is that the central power unit (CPU) and memory are increased on a router, as it has to process routing information and calculate its routing table.

2.2.4 Routing protocols

Several types of protocols have been developed, including distance vector routing and link state routing, which is favoured for large internetworks (Geeks for Geeks, 2018).

2.2.4.1 Distance vector routing

Distance vector routing is a protocol based on *distance* in terms of number of hops, which is the route metric to reach the network and a *vector*, which is the direction or interface to reach the network. An example of a distance vector routing protocol is RIP, which uses a hop count, and EIGRP, which utilises bandwidth. Once each router receives information on available routes, the Bellman-Ford algorithm is used to calculate routes utilising the shortest paths from a single source vertex to all other vertices in a weighted digraph. As an extension to vector routing protocols, some grid environments use ad hoc on-demand vector (AODV) and destination-sequencing distance vector (DSDV), which will not be expanded on in this chapter (Cioara, Minutella & Stevenson, 2008).

i) Routing Information Protocol (RIP)

RIP is used in small networks, as it is simple to configure and maintain, unlike the more complicated advanced features of routing protocols, such as OSPF or EIGRP. The protocol consists of version 1 and version 2, and both use hop count as a metric. Version 1 uses only broadcast for updates while RIP version 2 can advertise subnet masks and uses multicast to send routing updates. Multicast refers to the same communication being disseminated to a group of nodes. The entire routing table is sent by RIPv2 every half a minute, using considerable bandwidth (Cioara, Minutella & Stevenson, 2008).

In Figure 2.15, router R1 directly connects to the subnet, its RIP advertises the route and updates are sent by R1 to R2 and to R3. The subnet, subnet mask and metric are then listed for this route and are received by each router, so that the update can be added to their routing tables. The network consists of one hop applying a metric of 1.

It is interesting to note that distance vector protocols are prone to routing loops, prompting the introduction of features to prevent this occurrence.

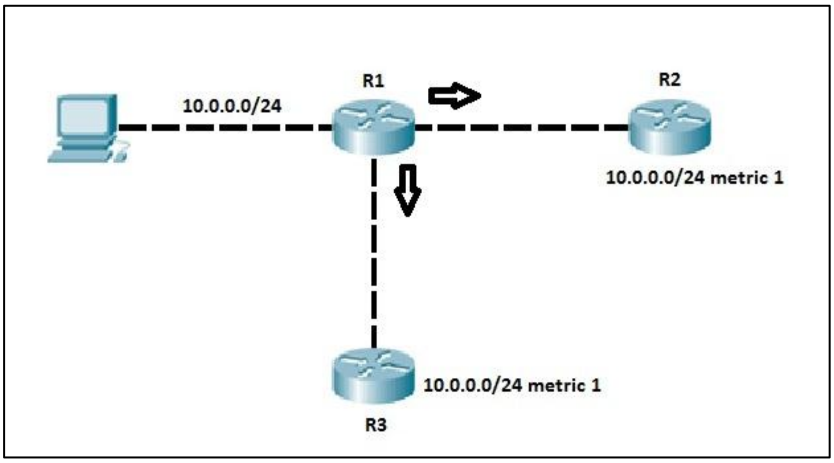


Figure 2.15: Demonstration of RIP functioning
(Source: Cioara, Minutella & Stevenson, 2008:232)

ii) Enhanced Interior Gateway Routing Protocol (EIGRP)

These Cisco routers run the advanced EIGRP distance vector routing protocol and establish relationships with neighbours, before exchanging routing information by sending packets to the multicast address every few seconds.

EIGRP calculates its metric using bandwidth, delay, reliability and load, while messages are sent via reliable routing protocol. Three tables store routing and topology information:

- *Neighbour table* – stores information on EIGRP neighbours. EIGRP neighbours are regarded as an autonomous system, with each router within an autonomous system having the same configured autonomous system number
- *Topology table* – stores route and interface updates, learned from neighbouring routers, successors and feasible successors to a destination, as well as locally connected subnets. A successor is the best route to take to a destination, while a feasible successor is the alternative path should the successor route in the routing table fail
- *Routing table* – stores only the best routes (successors) from the topology table (Cioara, Minutella & Stevenson, 2008)

Advertised distance (AD), also referred to as reported distance (RD) is the metric advertised by a neighbouring router for a specific route. Feasible distance (FD), listed in the routing table, is the metric of the best route to reach a destination. When a route is chosen as a feasible successor, a neighbour's AD for the route must be less than the successor's FD (Cisco, 2011). In Figure 2.16, EIGRP routers R2 and R3, both connected to the subnet, advertise their respective distances to R1, which receives the updates and establishes the best route. R2 is stored as the best metric (20) in the R1's routing table. The advertised distance of the R3 route (15) is found to be less than R1's feasible distance of 20 and router R1 stores that route in the topology table as a feasible successor route. The R3 route can be used instantly if the primary R2 route fails. Some multiple network layer protocols supported by EIGRP are IPv4, IPv6, IPX, and AppleTalk with EIGRP authentication, ensuring prevention from attack.

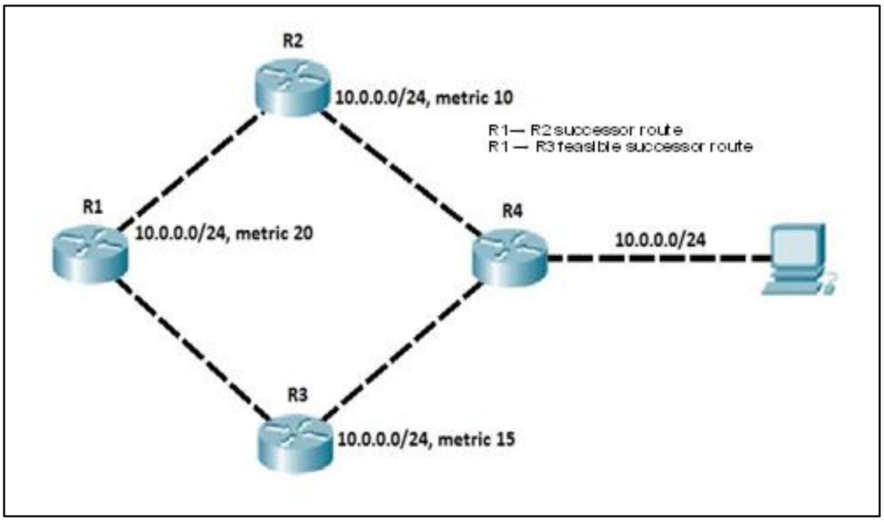


Figure 2.16: Network topology of successor route and feasible (Source: Cioara, Minutella & Stevenson, 2008:233)

2.2.4.2 Link State Routing Protocol

Link state routing provides faster convergence and a way to build a topological database in order to ensure the accuracy of inter-connecting routes. It is, therefore, more appropriate for large networks, like the Internet. Each router maintains a link state packet (LSP) that records the state of its neighbours' link information. A map of the entire network can be constructed when a router receives LSPs from other routers and the shortest paths to desired destinations are computed.

Link-state routing can be more difficult to configure, as it requires more router CPU and memory. The Dijkstra algorithm, which is merely the shortest path between nodes, is applied to calculate these routes to develop a routing table. Similar to EIGRP, each router running a link-state routing protocol creates three different tables:

- *Neighbour table* – which stores information on neighbouring routers running the same link-state routing protocol
- *Topology table* – which stores the topology of the entire network
- *Routing table* – which stores the best routes

Shortest Path First algorithm is used to calculate the best route of which OSPF is an example that utilises the cost metric (Cisco, 2011).

i) Open Shortest Path First (OSPF)

Unlike EIGRP, the link-state protocol OSPF can run on most routers, uses specific multicast addresses for routing updates and only interface cost as the metric, and uses the concept of areas. Areas are a group of immediate routers and networks, which have tables with the same topology and are unfamiliar with routers in other areas. The OSPF protocol converges swiftly, supporting many features such as variable length subnet mask (VLSM), manual route summary, incremental updates and equal cost load balancing. Before routing takes place, a neighbourly relationship is established because neighbours do not exchange routing tables in a link-state routing protocol; instead, they exchange information on the network topology. OSPF algorithms are calculated by each router to determine the best routes and are added to the routing table. Routing loops are rare, as each router is familiar with the entire network topology. The three tables of the OSPF router store the following (Cisco, 2011):

- *Neighbour table* – which stores information on OSPF neighbours
- *Topology table* – which stores topology of the entire network
- *Routing table* – which stores the best routes

In Figure 2.17, for example, each area in the OSPF network is connected to the backbone area (area 0), and each router inside an area has the same area ID in order to become OSPF neighbours. Router 3 is an area border router (ABR) as it has interfaces in more than one area (area 0 and area 1) with routers R4 and R5 inside area 1. Router 6 is an autonomous system border router (ASBR) as it connects the OSPF network to a different routing network, the EIGRP network. Should R1's subnet fail, it then sends the update to R2 and R3 as routing updates are localised in a specific area (Cioara, Minutella & Stevenson, 2008).

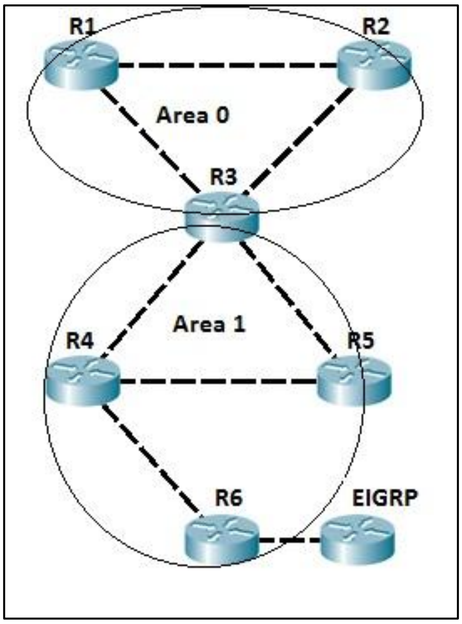


Figure 2.17: OSPF area function
(Cioara, Minutella & Stevenson, 2008:248)

ii) Link State Advertisements (LSA), Link-State Requests (LSR) and Link-State Updates (LSU)

Link state advertisements (LSAs) used by routers contain routing and topology data on a specific area of the OSPF network. This information is exchanged with neighbours. Should two neighbours decide to exchange routes, they send each other a list of all LSAs in their respective topology database. Each router then checks its topology database and sends a link state request (LSR) message requesting all LSAs that are not featured in their respective topology tables. The routers will then respond with the link state updates (LSU) that contain all requested LSAs.

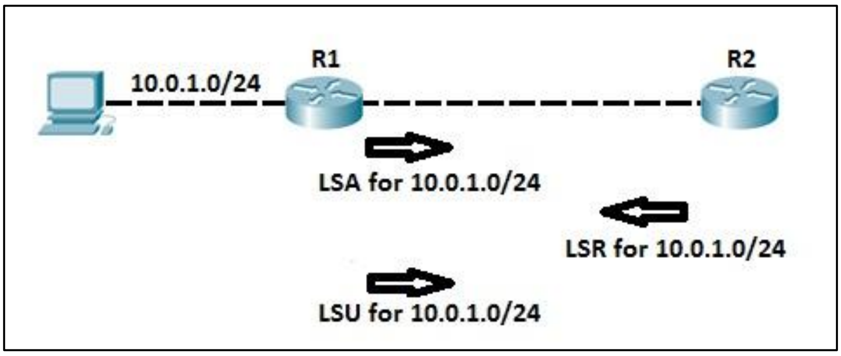


Figure 2.18: Link state function of OSPF
(Source: Cioara, Minutella & Stevenson, 2008:258)

In Figure 2.18, both routers have configured OSPF and exchange LSAs to describe their respective topology database. Router R1 sends an LSA header to its directly connected network 10.0.1.0/24. Router R2 checks its topology database and determines missing information about that network, and then sends an LSR message requesting more information. Router R1 reacts with LSU containing information about the subnet 10.0.1.0/24 (Cioara, Minutella & Stevenson, 2008).

2.2.4.3 Summary of differences between distance vector and link-state protocols

The differences between distance vector and link-state protocols are best summarised in Table 2.2.

Table 2.2: Summary of differences between distance vector and link state protocols
(Source: Cisco, 2011)

Distance vector	Link state
sends the entire routing table	sends only link state information
slow convergence	fast convergence
susceptible to routing loops	less susceptible to routing loops
updates are sometimes sent using broadcast	always uses multicast for the routing updates
doesn't know the network topology	knows the entire network topology
simpler to configure	can be harder to configure
examples: RIP, IGRP	examples: OSPF, IS-IS

2.2.4.4 *Internet Protocol Version 6 (IPv6)*

IPv6 is the latest IP protocol version developed to overcome the imperfections of IP4, particularly regarding address overload. This protocol is located on the network layer (Layer 3) of the International Organisation of Standardisation/Open Systems Interconnection (ISO/OSI) model and allows additional information to be attached to IP packets by extension headers (Krishnan et al., 2012). Similar to IPv4, IPv6 supports routing protocols that allow the change of information between connected networks that are both internal (RIP, EIGRP) and external (BGP), distance vector (hop count) or link state (cost metric). Additional features of IPv6 include enhanced security and stateless address configuration, which means that devices operating in this protocol can configure themselves automatically with an IPv6 address. The three types of IPv6 addresses used are as follows:

- *Unicast,* which is a single interface where packets are delivered to a single interface
- *Anycast* identifies several interfaces; Anycast addresses are used for load balancing
- *Multicast,* which is a group of dynamic hosts where packets sent to an address are delivered to several interfaces (Deering & Hinden, 1998)

2.2.4.5 *5G communications*

Fifth-generation (5G) technology for mobile communication was deployed in 2020 for the first time and predicted to achieve data transfer rates 100 times higher than 4G does, with a carrying capacity 1000 times greater than 4G (Andrews et al., 2014). Considering the large number of connected devices, security has been taken into account in the design of 5G as a multi-service platform, and 5G is expected to perform as an efficient and reliable communication technology that delivers ultra-low latencies and ultra-high reliability. Furthermore, 5G mobile technology will allow for direct device-to-device (D2D) communication, thus discarding any network intermediary device or base station, and has been shown to improve the multi-casting performance (Liang et al., 2017). This means that a hop gain is acquired by sidestepping a base station, reducing latency that may occur with V2V applications. MAC layer control in 5G can be centralised, allowing for guaranteed channel access, and is allocated access to spectrum in several bands in order to support various applications and requirements, including vehicular applications.

2.3 Evolution of Artificial Intelligence (AI) and the Internet of Things (IoT)

AI is able to cope with AV bid data but certain conditions such as traffic, pedestrians, experiences and routing require data to be accumulated through IoT networks such as: i) LAN, where OSI is employed and configuration includes the routing mechanism of packet switching; ii) WAN, which connects to vast networks; iii) WSN, which is the sensor network connecting the virtual and physical world; and iv) PAN, the personal area network which is a computer network for interconnecting electronic devices centred around the individual. This information requires software and network connectivity with substances such as embedded electronic devices, sensors, roadside infrastructure, buildings and vehicles to collect and share information, and AI needs assistance in uniting the mined data (Khayyam, 2020).

At present, no processor is able to outperform the human brain. The spiking neural network architecture (Spinnaker) supercomputer built by the University of Manchester's School of Computer Science can complete 200 million actions per second, which is only a small percentage of what the human brain can accomplish (Kenny, 2018). It is the power of the computer which is the human brain that has inspired researchers to understand its complex 'mechanisms' and to endeavour determining its computing ability, with the purpose of replicating a machine using similar principles, and so improve the quality of life for humans (Kenny, 2018).

2.3.1 Artificial Intelligence (AI)

The term 'artificial intelligence', coined by John McCarthy in 1956, refers to machines that can perform tasks similar to human acumen. Based on the workings of the human brain, artificial intelligence is an endeavour to develop a machine able to emulate such behaviour and even improve on the performance of complex human tasks that include planning, recognition of objects, sounds, learning, understanding language and problem solving (McClelland & Feigenbaum, 2017).

The British mathematician, logician, and founder of computer science, Alan Turing (1912-1954), first introduced the quantitative mathematical terms applied to simulation model algorithms in the 1940s. Credited with developing the first computer, the 'Turing Machine', he advanced the ideas of algorithms, computation and the 'Turing Test' in artificial or machine intelligence. Turing invented one of the key ideas behind the theory of computing in 1936 and anticipated that a machine would carry and remember its own instructions, as well as the data it was using, but he did not get the opportunity to demonstrate this. During the Second World War, Turning worked on the top-secret Colossus machines, where he was responsible for breaking the code used in the German cipher machines through an analysis of the logical structure generated by electro-mechanical rotor-coding machines: Enigma and Lorenz. It is believed that his actions were instrumental in considerably shortening the duration of the war (Flood & Wilson, 2012:334). Turing posed the question: "Can machines think?" In his 1950 paper, published in the journal *Mind,* which focused on the meaning of thought and human behaviour (Turing 1950:433-460), Turing postulated that a machine able to display the same intelligence as that of a human is, in fact, as intelligent as a human. This has brought to the fore whether imitation is sufficient to accept that a machine can think. Furthermore, conscious thought is displayed through behaviour, and if this is indistinguishable between human and computer, then one cannot be referred to as "thinking" and the other as "non-thinking" (Turing 1950:433-460). In contrast to Turing's hypothesis, Kurt Gödel (1906-1978), the Czech mathematician, developed Gödel’s theorem, describing the human mind as capable of working out truths that cannot be decided by any formal or mechanical procedure. This was later also seconded by the famous British physicist and mathematician, Roger Penrose. Artificial intelligence programs, however complex, were therefore merely seen as machines with formal finite systems in terms of Gödel’s Theorem (Stokes, 2012).

Turing agreed with Gödel and Penrose that although there were limitations to the power of a machine that uses a formal language, Gödel’s Theorem assumes, without substantiation, that human intellect does not suffer from the same kind of limitation (Stokes, 2012). The question, however, remains: “Is a computer that imitates the behaviour of a thinking human really thinking?” This has led to a massive and ongoing commissioning of research in the field of artificial intelligence and its applications, of which this investigation into autonomous vehicles is no exception for, although artificial intelligence was not the focus of the research, its inextricable and pervasive influence persists.

Of such is the magnitude of Turin’s premise that in 1990, Hugh Loebner introduced the Loebner Prize, which awards up to $100 000 and a gold medal to anyone creating a computer able to pass an extended Turing Test that includes textual, visual and auditory components which could baffle judges in determining the difference between computer and human (OCF, 2021). The prize will dissolve once awarded. During the course of the competition, the question emerged on whether physical and intellectual separation is desirable in evolving AI systems. According to opponents of the Turin Test, contestants aim to show that computers can fool people rather than being a holistic simulation of human intelligence. Some objections raised refer to (OCF, 2021):

i) Data processing objection, which states that data are viewed bit-by-bit in sequence whereas humans view data holistically. Turin regarded this as immaterial.
ii) Arguments from consciousness are perceived as a lack of feelings, thoughts and emotions, so it cannot equal a human. According to Turin, two entities can persuasively argue that the other cannot think.
iii) Mechanical objection merely follows instructions for manipulation but lacks comprehension. Turin stated that one should look at the functional aspect and exploit its capabilities as one would a person.

As much as Turin cannot prove that a machine can think, neither can his opponents prove the contrary – Turing did not set out to prove in any way that a computer is human (OCF, 2021).

2.3.2 Machine learning (ML)

Arthur Samuel (1901-1990) introduced the term 'machine learning' (ML) in 1959 and defined it as "the ability of a machine to learn without being explicitly programmed" (McCarthy & Feigenbaum, 1990:10). The majority of existing areas of knowledge consist of data from which we want to obtain the largest amount of useful information. Archiving some of this domain-specific data allows us to extract and analyse the data, and from which we can learn; therefore, the core purpose of ML to achieve AI. The two main sources of inspiration behind the various ML methods have emerged from statistical mathematics and biology, such as genetic algorithms based on natural evolution and deep learning, inspired by the way that neurons in the brain learn. Rosenblatt (1958) developed a mathematical system to simulate the basic operating principle of biological neurons and named it the 'perceptron' (Hush & Horne, 1993). Using the perceptron entailed inputs, weights and processing components that altogether progressed to nonlinearity, namely the sigmoid function, which is a mathematical function with a characteristic S-shaped curve. Linear functions are commonly used as output nodes (Neser, 2006).

$$y = \sum_{k=1}^{n} x_k w_k$$

$$u = f(y)$$

y = output of element processed

x_k= the kth input

w_k= corresponding weight to kth input

n= number of inputs to perceptron

Figure 2.21 illustrates the above equation, showing neural networks connected in parallel to form layers, which are also linked to each other. To create a network, the outputs of a layer of neurons are joined with the inputs of another layer of neurons.

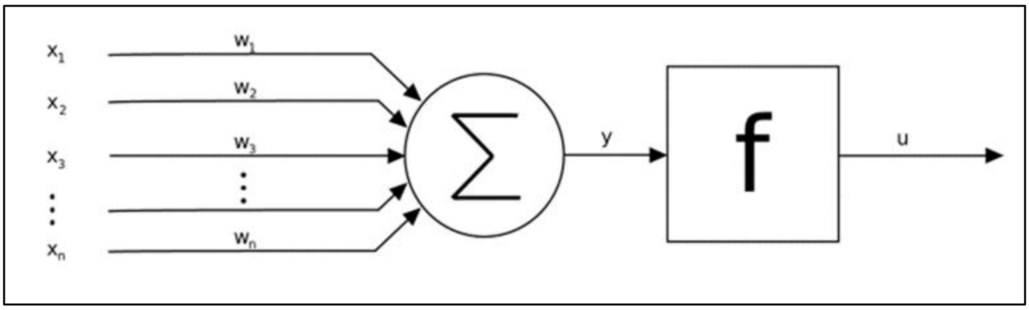


Figure 2.19: Perceptron (neuron)
(Source: Neser, 2006:38)

The multi-layer perceptron (MLP) consists of perceptrons that form a static network by being connected front-to-back, and can be used to make complex decisions as well as approximate a continuous function. The arrangement allows for a back-propagation-learning algorithm that attempts to keep the errors between output and input to a minimum (Neser, 2006).

In a more recent study, Guergiuev, Lillicrap and Richards (2017) developed an algorithm also based on neurons in the neocortex of the human brain, which are responsible for higher order thinking. They considered that inputs at the roots of the neurons differed from those of branches at the top of the tree of each neuron, prompting the researchers to build a model able to receive signals in segregated compartments. By modelling the structure of a neuron, they achieved deep learning, as these sections allowed simulated neurons to collaborate in the different layers.

To obtain AI without the use of ML, the construction requires zillions of intricate codes with rules and decision trees, and a myriad of specific instructions. ML is a method by which an algorithm fed with large amounts of data can learn, through 'training', how to regulate itself and improve in order to perform tasks similar in aptitude to those performed by humans. For example, a portrait-drawing robot designed in London that understands what it sees by making use of the software simulation of neurons, used in the context of the human brain's visual cortex where information is processed from our vision, produces sketches that cannot be discerned from those drawn by a human (McClelland & Feigenbaum, 2017).

Through the structure of ML by way of algorithms and approaches, much can be learned from data with the intention of predicting values or the future, based on what is already known. ML programs gain experience, learn from this experience, and apply it to produce accurate results (Kohavin & Provost, 1998). This contrasts with conventional problem-solving methods using predefined algorithms that gradually lead to a result. ML tasks are classified into two categories:

- *Supervised learning* – A computer is provided with inputs and is taught (and expected to learn) to map the inputs into desired outputs
- *Unsupervised learning* – No labels are given to the learning algorithm and it has to determine its own structure in order to determine an outcome (Bishop, 2013)

With relevance to this research, the latest generation of vehicles are capable of recognising images that they perceive as obstructions. They can park themselves, drive on highways and perform emergency braking. Furthermore, by studying our driving habits, the machines can adjust how they respond to us. Data processed through AI using ML algorithms allow behavioural patters to be formed for driver profiles by recalling driver behaviour and analysing driving history and the road situation. Taking this a step further, computer scientist

Haitham Baomer at University College London has developed an intelligent autopilot system able to cope in adverse weather conditions and emergencies (Baomar & Bentley, 2016). AI learns from observing (visual imagery) human pilots in minute detail to learn to cope with, and apply, the necessary skills to new situations. Similar to the multi-tasking processes of the human brain, AI uses all parts of its brain simultaneously to solve problems.

An approach to ML, referred to as ‘deep learning’, was inspired by the interconnection of neurons in the structure and function of the brain. An artificial neural network (ANN) – algorithms that mimic the human brain – is used, with numerous neural nodes arranged in tiers, and with each one solving a specific part of a task simultaneously. Instead of raw or unprocessed input, successive tiers receive outputs from previous tiers. Multiple layers acquire depth instead of using a single layer or tier. Nodes have their own store of knowledge constructed from their original programming rules and, in addition, from the knowledge that they have acquired through experience. The underlying goal is the recognition of unspecified items that the network has learnt to identify by itself (McClelland & Feigenbaum, 2017). While ML focuses on predictions based on what is already known from data training, data are extracted from previously unknown properties for analysis by means of data mining, which is the process of discovering patterns in large data sets (O’Brien & Marakas, 2011). Miniscule computer chips, more powerful sensors and enhanced manufacturing techniques have driven the converging technological advancement of the Internet of Things (IoT).

2.3.2.1 Light Detection and Ranging (LiDAR)

To enable autonomous driving in Level 3 vehicles and upwards, three types of sensor systems are required, shown in Figure 2.22, namely camera, radar and LiDAR systems, as several of each type of sensor operate at various locations on the vehicle (Roedolf & Voelzke, 2017).

LiDAR is one of four types of sensors used for AVs that feed information to the integrated AI controlled platforms which allows 3-D map construction and classification of the surroundings (Rablau, 2019). The 3-D sensing technologies include:

- Cameras for passive and visual sensing to collect 360 degree data from the local environment and for lane departure warning
- Radar for the detection, localisation and tracking of objects by means of radio waves, blind spot monitoring, and lane change assist; also for long-range detection of objects at high speed and short-range for park assist and rear-end collision warning
- Ultrasonic sensor substitutes with short-range radar

- LiDAR with good angular and linear resolution which uses long-range for detection, localisation and identification of objects from 200m onwards, and short-range LiDAR only detects and identifies the presence of objects (Rablau, 2019).

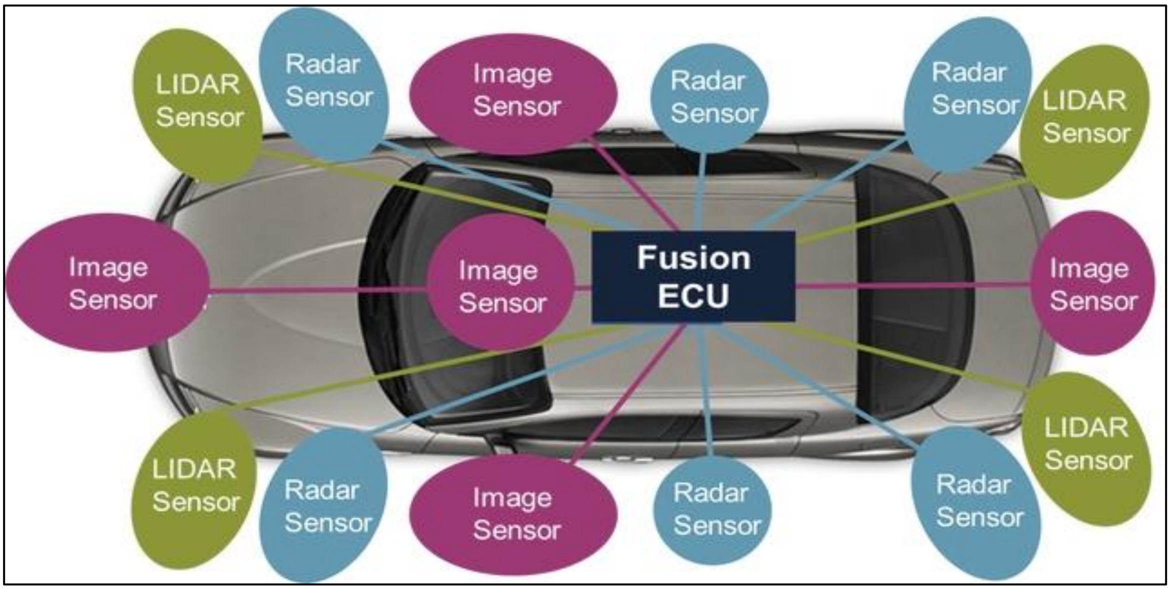


Figure 2.20: Different sensors used in an autonomous vehicle
(Source: Roedolf & Voelzke, 2017)

As a LiDAR system consists of lasers, optic scanners, a photodetector and a navigation positioning system designed to provide accurate three-dimensional information of the surroundings, it is a major component of the Google and Tesla autonomous car concept.

In addition to the transmitter, LiDAR requires a highly sensitive receiver. Current radar systems use a frequency rate of 24 GHz, or the preferred 77 GHz, as it is more accurate for distance and speed measurements. A higher frequency rate provides a more precise angular resolution, requiring a smaller antenna size and is subject to fewer interference problems. Methods are presently being developed through ML for the development of camera systems that use algorithms to detect traffic signs and signals automatically, as well as non-motorised objects such as pedestrians, cyclists or horse-drawn carts, since they demonstrate differences in physical appearance to motorised objects. These, obstacles, in addition to travelling speed, complicate the variables that must be taken into consideration. Furthermore, since pedestrians and other non-motorised objects are not fitted with devices that transmit their positions, matters become more complicated (Schippers & Socash, 2017).

A LiDAR photo detector arrangement of 64 individual infrared lasers emits light beams that it projects towards the surroundings, which enables the measurement of the distances of the vehicle from stationary and moving objects. This principle is shown in Figure 2.23.

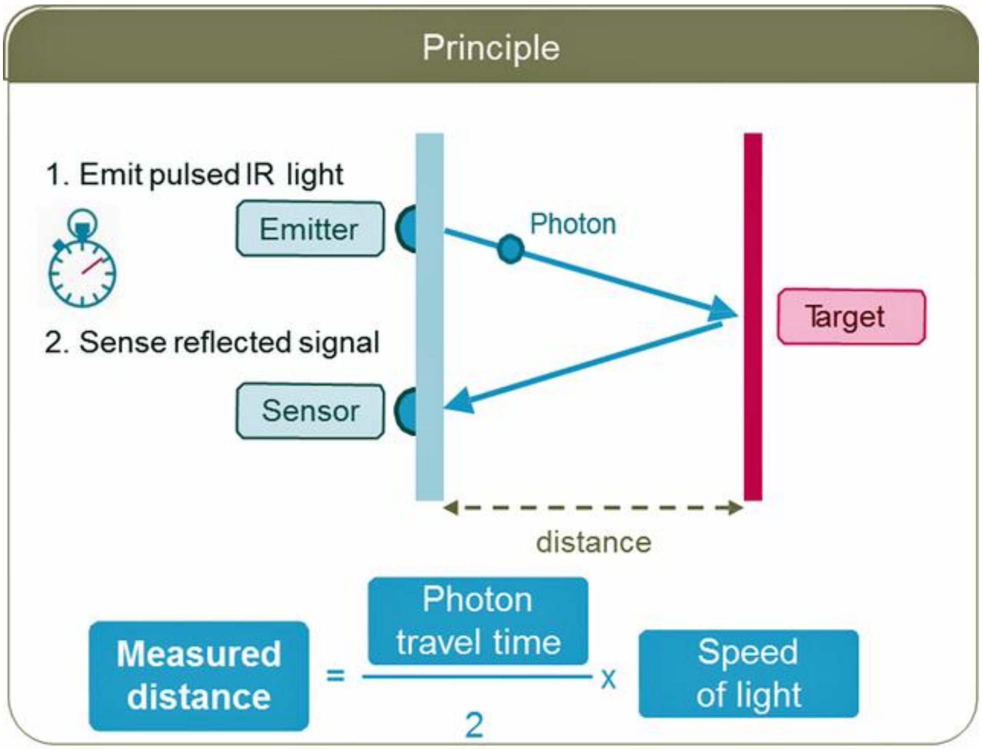


Figure 2.21: Principle of LiDAR distance measurement
(Source: www.monash.edu)

Measurement of distances is done by calculating the average time it takes for light to reach the object and return to the sensor. The data are arranged in a three-dimensional array that, when the LiDAR scanner rotates, designed to do so at 300 revolutions per minute, a map of its surroundings is created that can serve as a data model for intelligent ML. The system does not function effectively in poor weather conditions, even with current booster detection systems employed to enhance its 'visibility'. By integrating the ML and computer vision algorithms, the detection system could 'learn' to identify non-motorised objects, as well as any future data it becomes exposed to, allowing the vehicle to adapt to its environment and increasing the efficiency of the system (Schippers & Socash, 2017). This provides further evidence of collaborating systems in the technological evolution of IoT.

A 'Digital Motorway Testbed' launched by Audi introduced new technologies and in cooperation with the German Federal Ministry of Transport (Pawsey, 2017), Audi and the Ministry are concentrating their efforts on the development of piloted and autonomous aspects of driving. The test site comprises sections of the A9 motorway between Nuremberg East and Munich North. These sections of the motorway are equipped with transmitters and sensors that connect vehicles with their surroundings as well as with other vehicles. Research projects are currently being implemented to test communication technologies and to test the structural measures such as materials used for marker posts and guardrails, in order to improve the reflection of radar waves from a greater distance and in adverse weather. Standardisation introduced by IoT to wireless sensing will revolutionise other areas where environmental monitoring may be required as an early-warning system and relevant to vehicle safety, such as air pollution or a potential earthquake (Hart & Martinez, 2015).

Further research is taking place in an attempt to improve sensors on vehicles in order to detect road and roadside markings easily, thereby enabling vehicles to pinpoint their position accurately, via a camera, in relation to these markings (Pawsey, 2017).

The ‘Car2Infrastructure’ communication project connects the car to an online network, providing drivers with real-time information such as speed limits, congestion and road works, by using a mobile network connection. Sensors gather information, which is uploaded via a mobile network to the Audi cloud services, from where it is transmitted back to test vehicles to assist drivers. Cars of the future will be able to communicate directly with one another, even in regions without mobile coverage (Pawsey, 2017).

In the 2017 *Drive Me autonomous trials* in Gothenburg, Sweden, Volvo equipped test vehicles with their IntelliSafe Autopilot system and tested these vehicles on applicable routes chosen to provide trail-gathering feedback about autonomous vehicles on conventional roads in real-life situations. The Transport Systems Catapult’s (TSC) Lutz Pathfinder project in the UK offers a commuter-based system where automated self-driving pods traverse a designated path in order to ease city congestion. Data from cameras and LiDAR systems enable the pods to navigate their environment (Volvo Cars, 2016).

To reiterate, lasers and radar are applied to navigate AVs by means of radio frequency waves being reflected from neighbouring vehicles and other objects. This would indicate that prior to the technology being implemented, research will have to determine how a vehicle discerns its own reflected signal from that of the reflected or transmitted signals of thousands of other vehicles (section 2.3.2.1). The interconnected system of packet switching is boosted by multi-layered switching techniques transmiting a multitude of data requests. A router analyses incoming data, utilising variables of time and distance in the process to identify the best possible route before packets are forwarded (section 2.2). Applying the analogy to a multitude of vehicles will consequently minimise the drastic usage of signal transmission, enabling safe passage for AVs simultaneously traversing a roadway.

Sensors with the capability of sending data over the Internet by means of technologies and standards, such as Wi-Fi, IEEE 802.11 or gateways for packaging the information in Internet-based protocols, become ‘smart objects’ and embrace the vision of IoT (Mois, Folea & Sanislav, 2017). In the words of researchers Guillemin and Friess (2011:12), “this vision imagines a world in which ‘people and things’ are connected Anytime, Anyplace, with Anything and Anyone, ideally using Any path/network and Any service”.

2.3.2.2 VHF Omni-directional Radio Range (VOR)

Very high frequency omni-directional radio range (VOR) embraces the possibility of introducing a new concept regarding the use of beacons in the routing of autonomous vehicles, and is worth mentioning in a review of the literature. VOR is fitted with an omni bearing indicator that assists pilots to determine the direction of their aircraft from any position to or from a VOR beacon on a selected bearing (Australian Civil Aviation Authority, n.d.). VOR is a VHF navigational aid, although its ground to air range is limited to the 'line of sight' functionality typical of VHF transmission. The range of the transmission achieved is reliant on the position of the VOR beacon in relation to the surrounding topography. An infinite number of bearings, referred to as radials, are available that radiate from the beacon like spokes from the hub of a bicycle wheel. A radial is identified by its magnetic bearing outbound from the VOR beacon, which also transmits an identification signal of two or three Morse code letters that identify the beacon. According to the Australian Civil Aviation Authority (n.d.) the pilot can select, identify and locate a line of position from a particular VOR beacon and obtain the following information:

- The magnetic bearing of the aircraft from the VOR beacon
- The magnetic bearing from the aircraft to the VOR beacon
- The position of the aircraft, that is, port or starboard, of a selected radial
- When the aircraft is closing in on, and when it is flying along, a selected radial
- When the aircraft passes over the VOR beacon

A VOR system is accurate to within about 5 degrees (Australian Civil Aviation Authority, n.d.).

2.3.3 Internet of Things (IoT)

IoT has evolved from machine-to-machine (M2M) connectivity, links the virtual world with the real one, and is defined as a network of physical devices such as vehicles and home appliances, embedded with electronics, sensors such as GPS location data, software actuators and network connectivity that enable the items to communicate and exchange information (Brown, 2016; ITU, 2020). The relationship between AI and the IoT is similar to that between the human brain and body, in that the two are inexorably interconnected. The human body accumulates sensory input in the form of touch, sound and sight, and the brain interprets the data into comprehensible information. The many connected sensors that make up the IoT provide the raw data, as in the case of the human body. Artificial intelligence is like the brain, which makes sense of the data received and determines what action needs to be executed. The IoT-connected apparatuses, like the body, communicate with one another and perform the physical actions (McClelland & Feigenbaum, 2017). This translates into networked inputs combined into a system that integrates data, people and processes for better decision-making. There are six components involved in the multi-level data exchange

IoT-based vehicle ecosystem that, through network connectivity, work together. These components are: network infrastructure, vehicle, person, personal device, sensing device, and roadside device (Khayyam, 2020).

According to Calum McClelland, managing editor of *IoT for All*, IoT may require correlations to be identified between myriads of sensor inputs and external components constantly producing trillions of data points. Unlike data analysis that requires a model built on historic data and the expertise to establish relationships between variables, ML commences with an outcome variable (e.g. congestion reduction) and, subsequently, automatically searches for forecasting/predictor variables and their interactions (McClelland, 2020). This means that a machine is given a learning algorithm in the form of a goal and, subsequently, 'learns' from the data what is required to accomplish that goal. ML's predictive analysis is valuable to numerous IoT applications. Goldcorp Mining uses IoT to predict when their machines need maintenance, which has considerable economic benefits (Goldcorp, 2018).

ML and its deep learning facet involve the processing of enormous amounts of data to be effective, collected by the trillions of sensors that constantly come online in the IoT to improve AI. In turn, improving AI drives the adoption of the IoT in a continuous beneficial cycle. Industry can benefit, since AI can assess the cost-effectiveness of a manufacturing process or determine when machinery will need maintenance. Consumers can have access to technology adapted for simplicity, such as simply asking for information, a weather report, for example, instead of searching for the information. Data that are pre-processed to be accurate allow for faster and easier machine learning (ML), and with this, allowing the ability to choose the right approaches and algorithms. Storage facilities such as cloud services allow for almost unlimited storage for data sent in high volume via wireless sensors (McClelland, 2020) (section 2.3.3.2). IoT connectivity has also been demonstrated in situations such as home automation and security systems for the home, where homeowners can have a visual link to the inside of their homes via their mobile phones from anywhere in the world (section 1.3.1), or by having doors lock 'themselves' on request. Criticism, however, has been directed at some companies that have plunged into IoT without effective security measures being considered when creating devices (Porup, 2016).

Applications of IoT are also considered in the medical realm. For example, voice control can help persons with limited sight and mobility, and sensors that monitor medical emergencies can be installed (Mulvenna et al., 2017). The term Enterprise Internet of Things (EIoT) (section 1.3.1) is used in business and corporate environments, while the term for industry is Industrial Internet of Things (IIoT), the foremost objective being to increase productivity by reducing downtime and, thereby, maintenance costs (Daugherty et al., 2015; Greenough, 2015). Organisations need to monitor trends in IoT and discover emerging technologies to increase business competitiveness, thus, maximising profits by reducing operating costs.

Figure 2.24 shows the Hype Cycle and provides an overview of emerging trends in organisations for 2018, drawn up by Gartner Incorporated, the world's leading research and advisory company. The graph illustrates emerging technologies that should constantly be monitored by business leaders (Panetta, 2018). Gartner confirms that AI is critical for autonomous vehicle technology, the development of which has led to an increase in ML algorithms. Related technologies have also advanced in the fields of sensing, imaging and mapping; but that high costs and the complexities involved are challenging (Panetta, 2018).

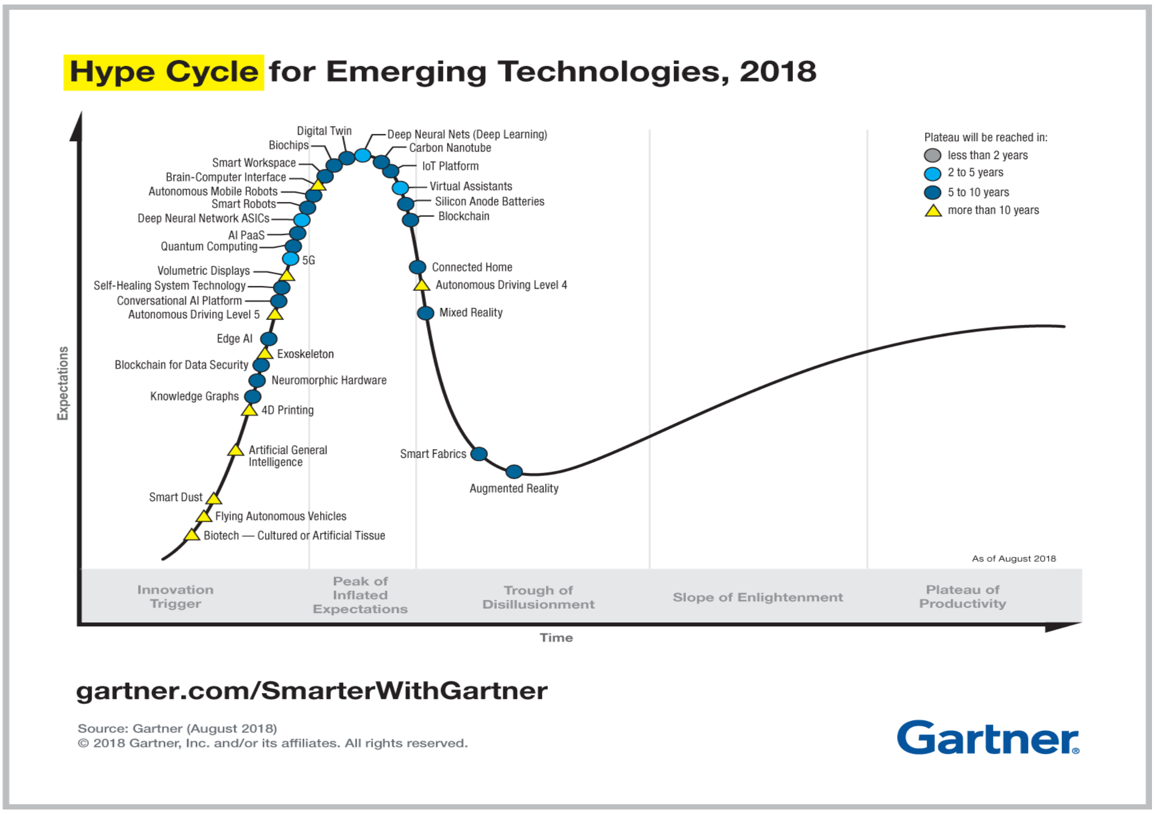


Figure 2.22: Gartner's 2017 Hype Cycle for emerging technologies (Source: Panetta, 2018:3)

Further to this, the interaction between humans and the cyberworld in cyber-physical systems is based on 5C architecture, namely connection, conversion, cyber, cognition and configuration, as illustrated in Figure 2.25 (Lee, Bagheri & Hung-An, 2015). Industrial data analytics in the manufacturing industry can play a major part in profit margins, such as predictive maintenance, as accumulated data are transformed into actionable information that interacts with the physical assets to optimise processes. This concept was first demonstrated in 2014 on a band saw by the National Science Foundation Industry at the University of Cincinnati (Lee, Bagheri & Hung-An, 2015).

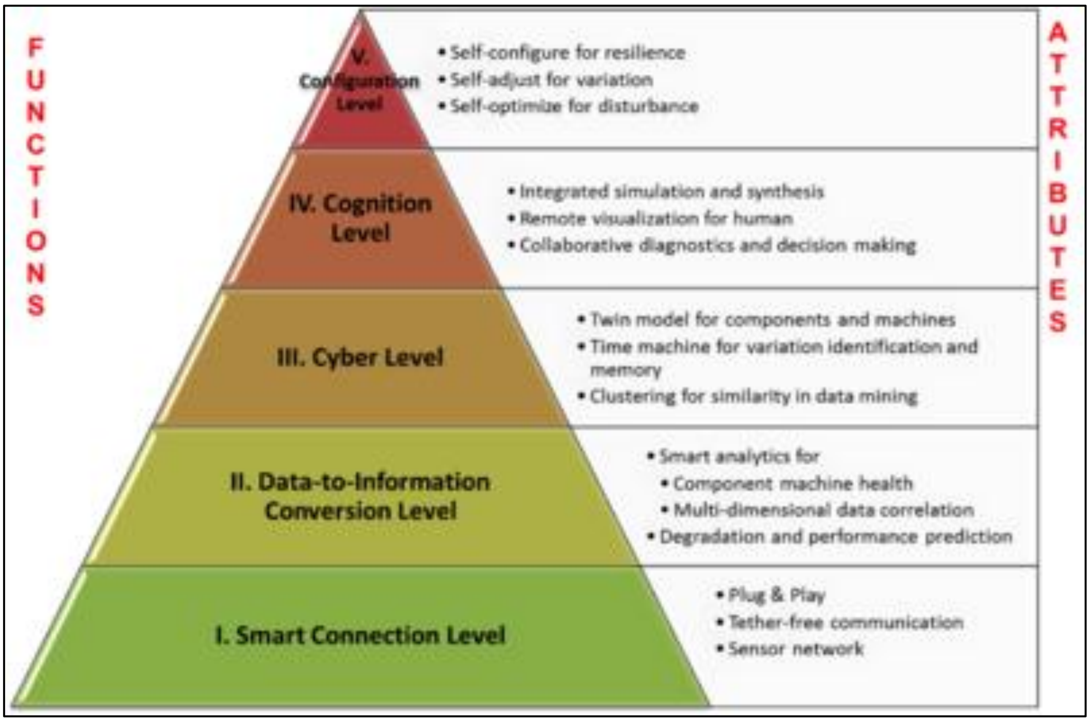


Figure 2.23: Design architecture of cyber-physical systems-enabled manufacturing system (Source: Lee, Bagheri & Hung-An, 2015:19)

Integration of IoT involves all areas of the automotive industry, from transportation, vehicle enhanced features like directing a vehicle into a parking bay, infrastructure and the driver, through the interaction of all elements, to enable communication. Aston Martin, BMW, Tesla and Volkswagen are integrating principles of ML and AI into their design processes. IoT successfully applied to an autonomous vehicle routing system would allow vehicles to exchange data via network connectivity, while each is uniquely identified by means of its embedded computing system to improve efficiency and safety, since human intervention is omitted. Information collected from the surrounding environment by means of networking-embedded devices and IoT applications aids the routing of vehicles. Meanwhile, IoT infrastructure management continues to be used to monitor the changes in structural conditions that can impede safety, such as maintenance to bridges (Jayavardhana et al., 2013).

It is interesting to note that architecture of IoT reflects a bottom-up approach, which means it is founded on processes and operations in real-time, with operational approaches synchronised with new approaches. IoT is considered a complex multi-agent system due to the myriad of links and interactions between autonomous components and IoT's capability to integrate new components (David, 2010; Hamid, 2017).

Whereas M2M connects systems of sensors to servers autonomously or with little human interference, IoT utilises M2M connectivity, integrates Web applications, and connects this to cloud computing systems (Khayyam, 2020).

2.3.3.1 Cloud Computing

Cloud computing is a service that allows IT services delivery, which, through Web-based tools and applications, can be retrieved from the Internet. Cloud computing allows IoT to operate with 24-hour accessibility, and it is flexible and cost effective (Zhang & Voas, 2013); it can furthermore allow AVs to connect to the cloud and update their software and maps.

AVs pose numerous challenges to IoT services regarding cloud computing resources and storage. Vehicular communication requires an Internet communication network platform with huge data information facilities to operate and deploy all the gadgets that require information gathering (Satyanarayanan, 2017). IoT can support millions of device connections, which can generate an enormous amount of data to be transferred and processed in cloud computing that is permanently accessible, flexible and cost-effective. Cloud computing incorporates four components:

i) Hardware and sensors that collect data, such as Radio Frequency Identification (RFID) and Wireless Sensor Networks (WSNs), which are among the technologies that enable the interconnection of the virtual world and physical objects. These result in highly efficient service, scalability and privacy enforcement and are able to carry cloud outages;
ii) A communication network centred on wireless technology such as Wi-Fi or cellular technologies, 3G, 4G and 5G is offered, but processing large amounts of data increases the workload of providers and the Cloud Data Centre;
iii) Big data that represents volume, velocity, and variety of data generated, which are stored, transferred and processed, and which incur overheads in terms of throughput, energy, time and cost; and
iv) Cloud, where the data are stored and processed by means of processing, analytics and storage. Services must not be located in different geographical areas in order to prevent latency and throughput problems (Satyanarayanan, 2017).

To address problems such as latency, network bandwidth and security, Edge computing, proposed by Mehdipour et al. (2019), introduced a model whereby process and planning sections are divided into two modules, handled by edge and cloud collaboratively. To reduce communication overheads, data analytics can be conducted at a network edge near the location where data are generated, and is referred to as Edge computing (Fog computing). Edge devices use fixed hardware resources that can be configured by users for each application, whereas the allocated resources in the cloud are intangible and generally inaccessible to users (Mehdipour et al., 2019). In addition, tasks can be migrated to another edge in the vicinity in case of failure, which is important for AVs that need a high reliability factor.

AV data applying AI are collected and transferred to the edge node for pre-processing before transferred to the cloud to be offline globally processed, whereas data collected from IoT sensors are analysed locally in the edge. This means that time-sensitive decisions such as crash avoidance can take place in the edge node in a shorter timeframe (Mehdipour et al., 2019). Edge devices can also collaborate with other edges in the vicinity and a local network beneath the cloud is created as an intermediate layer between data source and cloud. IoT devices interact in a spread environment together with the cloud, which reduces traffic and bottlenecks, addresses security and privacy concerns, and consequently prevents private data from being shared across other cloud environments (Mehdipour et al., 2019).

2.3.3.2 IoT security

Efficiency of any system depends on storing, processing and accessing the data and cloud networks are used for this purpose by companies working on IoT. When operating multiple systems, the question of vulnerability arises when data are collected from multiple sources and forwarded to a single destination (Thompson & Mattalo, 2015). Government legislation also comes into effect, designed to protect consumer choice and ownership of data (Anon, 2016). The National Highway Traffic Safety Administration (NHTSA) is in the process of developing cybersecurity guidelines and a best practice database to help secure automotive the computer systems applicable to their IoT industry (Wallace, 2016).

The ultimate goal for IoT is for virtual objects (avatars) to act autonomously, or to collaborate with other virtual objects, depending on the circumstances. These objects must gather reasoning information and need to be able to detect changes in the environment and initiate suitable mitigation measures (Alippi, 2014). Given the developments and on-going research in IoT, IIoT and EIoT, this study integrates the packet switching concept with VANET technologies in order to execute an effective routing mechanism for real-life autonomous vehicles. The surge in devices in IoT calls for huge scalability with IPv6 to handle the network layer safely and to benefit from stateless address auto-configuration, since it reduces the configuration overhead on the hosts (Arpan, 2015) (section 2.2.4.4).

2.4 The impact of AI and IoT on society

Gordon Moore, the founder of Intel, made the observation in 1965 that gave rise to Moore's Law stating that the number of transistors on a microchip doubles about every two years, though the cost of computers is halved. Another precept of Moor's Law says that exponential advancement of technology is set to accelerate the pace of job losses, often with one group being technologically favoured over another, according to Erik Brynjolfsson, Director of MIT Initiative on Digital Economy. He also expressed the opinion that low- and high-skilled jobs are presently not as threatened by automation as first thought. Jobs such as robotics, engineering and programming, however, are expected to be in demand as AI and automation will need maintenance and improvement. This is in accordance with Daugherty et al. (2015) of the Accenture Institute of High Performance, who contended that IoT is expected to establish new workforce needs, particularly with respect to skills in data science, software development, testing, operations and hardware engineering, as well as marketing and sales skills. New growth opportunities, and the technology challenges that need to be overcome, require a skilled workforce to accomplish. Three critical spheres are expected for the new talent base:

i) *Creating a new IIoT service sector* requires the skills and talent necessary to create and sell products, including: the product managers and software developers, required to create and test new information services; the hardware designers, required to develop the products; the data scientists, required to develop and analyse systems; and designers for planning infrastructure. Managers and marketers will be required to direct sales.

ii) *Supporting users of industrial products* requires companies offering IIoT tools to ensure that these are simple and convenient to use, in order to provide the technical expertise. Process engineers will be required to integrate services, as will data science and quantitative analysis skills be required to monitor incoming data.

iii) *Mastering new ways of working* means that the new technologies will change the way people do their work, for example, robots will be working alongside people to perform certain tasks more effectively (Daugherty et al., 2015).

The merger of Information Technology (IT) and Operations Technology (OT) are expected to take some time – different business functions own these technologies, and are served by different vendors operating under different technical standards, as explained in Table 2.3. Further positive aspects of AI have arisen in that automation can assist with advances in medicine and accurate diagnosis, and help produce better quality food to feed the masses.

Table 2.3: Merging IT and OT under differing technical standards
(Source: Pettey, 2017)

Present realities	Future vision
Software, sensors, and controls running facilities and equipment are outdated and difficult to upgrade. Incorporating new features and improvements is difficult	Services, communications and other operational technologies collaborating with information technologies
Limited integration between internal systems	Standard, fast software techniques create intelligent industrial products
Older operating systems are vulnerable	A common data model and sensing and control architecture to support flow of insights and action in an organisation and its environment
Limited embedded computing	IIoT infrastructure trustworthy and resilient to negotiation

Humans trust other humans before placing their trust in machines; therefore, companies use people to help 'train' their autonomous software and to encourage productive employees through human-robot engagement. Human assistance is used as part of the customer experience in various companies such as Amazon, for example. It was predicted that by 2020 price and product will take second place to customer experience (Walker, 2013) and this trend has been verified by the Temkin Group that discovered investing in customer experience allows companies earning $1 billion per year to expect to earn an additional $700 million within three years (Kulbytẽ, 2020).

McClelland (2020) asserted that because of our inherent human nature, we would continue to build intelligent machines to an even higher level of intelligence than that of human intelligence, which begs the question: "Can machines be considered intelligent?" As machines will be maintained and improved by other machines, our education systems need to be revised to incorporate lifelong learning in order to accommodate the constant technological change of the new millennium. Realisation of the impact of AI and IoT on society depends, therefore, on personal choice and as such, automation gives us the opportunity to pursue meaningful careers.

2.5 The potential impact of self-driving vehicles on society

IoT and AV Industry 4.0 is set to improve quality control and, in the process, increase productivity, lower costs and reduce machine failure. Morgan Stanley Research (2018) reported that up to nine industrial manufacturers stand to benefit through provision of technologies, features and services.

A research team from eight universities reported in *arXiv* that 5% of automatically controlled vehicles on a simulated single lane test track were able to eliminate stop-and-go wave actions. They simply adjusted the speed of, and distance between, the AVs by estimating the average speed of the vehicles in front of the AVs. This also helped to reduce braking and fuel consumption. It was evident from this and other research that any technology implemented towards attaining full autonomy, such as connected vehicles and adapted cruise control, were already improving the safety and quality of life of both humans and the environment (AccessScience Editors, 2017).

According to a report in *Rethinking Transportation 2020-2030* by *RethinkX*, by 2030 about 95% of US passenger miles will be covered by fleets of autonomous electric vehicles (A-EVs) that people will not independently own. Transport-as-a-service (TaaS) companies such as Uber, Lyft and Didi have spent considerable amounts of money enhancing services and technologies in order to attract passengers (Arbib & Seba, 2017a).

A-EVs are longer lasting and cheaper to operate than combustion engine vehicles. Automation is expected to affect the economy and society in the following ways:

- Transportation costs will drop, mainly due to competition, thereby saving on travelling costs for the average family, boosting disposable income
- Increased mobility for the elderly and disabled
- Number of vehicles will drop by one fifth, as fewer cars will travel more kilometres
- Demand for new vehicles will plummet by 70%
- A serious decline in oil production by oil producers
- Gross domestic product (GDP) will improve as less time will be spent driving
- Air pollution and greenhouse gases will be drastically reduced and public health improved
- Globally, 1.2 million lives will be saved annually
- New participants from other industries will enter the market, for example:
    - Sectors in vehicle operating systems
    - Alternative fuel industry, such as electrical energy that takes lighter vehicles into account, built as a result of reduced accident potential and environment friendly composites
    - Computing platforms, with other participants entering the market and
    - TaaS providers providing a level of service equivalent to, or better than, current car ownership, without the need to own a vehicle. This new industry is expected to revolutionise the transport sector and create wealth even greater than that generated by the computer, Internet or mobile industries. Affiliated players such as technology companies and battery manufacturers will enter the market, providing an array of economic and social incentives to potential employees. Further employment will be generated in extracting lithium, cobalt and cadmium required to produce batteries for AVs. Job losses felt by taxi and truck drivers will be affected by the TaaS industry. For example: navigation and mapping, software development for operating systems, and services such as on-board advertising and entertainment will become the norm and generate more employment prospects (Arbib & Seba, 2017b).

According to Arbib and Seba (2017b), electricity demand will increase because of the need to charge the A-EVs. However, this does not mean a need to increase the capacity (kW) of present electricity-generation infrastructure, as this infrastructure is designed for peak demand and not for efficiency. Charging of A-EVs during off-peak periods will accommodate the demand for electricity within the current power-generation infrastructure.

2.6 Previous studies conducted on routing mechanisms

The concept of full self-driving automation has been used for a number of years to manage traffic via conveyor belts and the Internet by means of packet switching technology. This concept has been introduced into the realm of transportation by researchers such as Fiske (2002:1) who suggested a transport system that involves packet switching to route magnetic levitating (*maglev*) vehicles travelling on a track. Later studies show that packet switching technology has been considered in accident-prevention studies by researchers such as Fawaz and Artail (2010), who incorporated a safety protocol into a conventional motor car's on-board computer software that forewarned of braking vehicles ahead through a forwarding system of packets.

2.6.1 Vehicular Ad Hoc Networks (VANETs)

Although vehicular ad hoc networks (VANETs) are not the main theme of this research, they play a major role. VANETs are inevitably found in IoT to provide communication channels between vehicular nodes accessed through cellular gateways and have been receiving serious attention for some time due to their importance in the development of ITSs (Bilal, Madani & Khan, 2011). VANET communication includes V2V as well as V2I communication by means of wireless links and offers services such as accident avoidance, on-road internet access, information on parking locations, locations of refuelling stations, and entertainment applications such as listening to the radio. Further to this, processing Advanced Driver Assistance Systems have accelerated the need for efficient hardware and software using sensors and actuator devices that produce data containing information such as time, date, navigation, motion detection, fuel consumption, voice recognition, eye tracking, driver monitoring, and image recognition, among others. Data total over 100 terabyte per annum for 100 000 vehicles (Sherif, 2017; Xu, 2018). Over the years, projects involving inter-vehicle communication have been initiated by companies such as Toyota, BMW and Daimler-Chrysler as steps towards the development of intelligent transportation services (Bilal, Madani & Khan, 2011; Liu, Wan & Wang, 2016).

Characteristics of VANETs, including intermittent connectivity, network partitioning, high mobility, predefined roads and obstacles in city environments make routing challenging and, therefore, reliable communication is questionable. Even so, the importance of the vehicular ad hoc network for providing safety-related applications in Intelligent Transport Systems (ITSs) has gained momentum, prompting the Institute of Electronic and Electrical Engineers (IEEE, 2018) to develop a specific wireless access in vehicular environment (WAVE) standard named IEEE 802.11p for VANETs because of their safety-related applications in ITS. The Federal Communications Commission (FCC) has also allotted 75 MHz of bandwidth in 5.9GHz band for licensed dedicated short-range communication (DSRC) (Abdalla, Aburgheff & Senouci, 2008; Bilal, Khan & Ali, 2017).

A study undertaken by Abbasi and Kahn (2018) on V2V communication using VANETs describes several position-based protocols in urban environments. The authors provide insights into routing in forwarding techniques, methods of junction selection and strategies to handle a local situation optimally. Their study was conducted on conventional urban roads and, although vehicles were not fully autonomous, their efforts to advance routing protocols were significant. Abbasi and Kahn (2018) presented a simulation study of both dynamic and static junction selection routing protocols and discussed their findings with comparative existing studies. They determined that a position-based unicast routing protocol based on location information, as opposed to a topology routing protocol, would be the most appropriate approach in VANETs for V2V communication in an urban environment. Unicast describes communication between a single sender and a single receiver over a network, whereas to multicast is communication between a single sender and multiple receivers.

Their decision to favour the position-based protocol was determined because of the frequently changing network topology experienced in VANETs. This results in protocols that are not scalable under highly mobile large-scale VANETs, as they incur high latency in seeking a route, owing to having stored all the unused routing paths. Furthermore, maintenance of routing paths in highly mobile environments results in high routing overheads for topology-based routing protocols (Abbasi et al., 2014; Fiore et al., 2007). Abbasi and Kahn (2018) also established that urban real-life environment cannot be effectively reflected in present routing protocols for VANETs and position based protocols. They confirmed that the high mobility and intermittent connectivity characteristics of VANETs might cause linkage problems, resulting in packet loss, rendering the communication unreliable. They proposed that a secure optimal path-based reliable and stable routing protocol is needed that encompasses real-life urban characteristics such as high mobility, intermittent connectivity and the sparse and dense nature of the network, in order to enhance the performance of packet delivery, end-to-end delay, routing overhead and hop count in VANETs.

Research by Rashdan, de Pont Müller and Sand (2017) describes the effective application of the new 5G mobile technologies, destined to be utilised in the form of ‘platooning’ in vehicular routing (section 2.2.4.5) currently using the ITS-G5 communication parameters. A platoon is a convoy of vehicles following in close proximity to each other, under the control of the leading driver. A cooperative adaptive cruise controller (CACC) in each vehicle establishes acceleration and speed, based on information from other platoon vehicles. On-board radar sensors on board each vehicle determines distance and relative speed of preceding vehicles so that the vehicles involved can adapt to speed and time gaps between them. The concept allows for less aerodynamic drag, allowing more efficient traffic flow and capacity (TRL, 2016).

Simulation results showed that, due to considerable time delay between transmitting and receiving messages and the low rate of packet delivery, the ITS-G5 communication system was unreliable for platooning, especially during peak traffic periods. This was also confirmed through simulations by Fernandes and Urbano (2012) and Chenxi et al. (2012). The study envisioned that the 5G communications system for vehicles travelling in close cooperation would increase safety and road capacity, while reducing fuel consumption (Rashdan, De Pont Müller & Sand, 2017). Nevertheless, while heavy goods vehicles (HGVs) make use of V2V communications and operate as platoons to reduce fuel costs, risks are presented as human drivers join platoons, and when obstacles such as potholes are encountered at high speed by a platoon participant (Metz, 2018). In addition, the platoon is restricted by the speed of the slowest member and problems arise when a platoon wishes to overtake another platoon. Furthermore, unequipped conventional vehicles attempting to participate in a platoon equipped with speed management systems would render the concept ineffective (ETSC, 2016).

In accordance with the above, the results of a survey conducted by Cavalcanti et al. (2018) show that VANET simulation-based studies lack credibility. Technical information on aspects, such as models involved, mobility and transmission, and input/configuration parameters tend to be omitted. They further claim that dataset or code repositories are absent. Further research shows that VANET nodes may be able to communicate in an infrastructure-less environment, because integrated advanced wireless technologies reduce the cost of infrastructure deployment (Gupta, Prakash & Tripathi, 2017; Bilal, Khan & Ali, 2017). Cavalcanti et al. (2018) also conceded that 5G mobile technologies were expected to be integrated with vehicular networks by 2020 (section 2.2.4.5).

The current research, however, highlights routing of autonomous vehicles with the intention of doing away with the human factor and, thus, the intervention of any decision-making by the driver, who can override any routing protocol that research on VANETs has yet to take into account.

2.6.2 Other research on routing mechanisms

Research by Alazab et al. (2011) on how to route a vehicle optimally through a conventional transport network based on traffic flow information in real time is also relevant to this study, as it considers one of two states, namely, congested or un-congested, which are common occurrences in packet switched networks. The authors examined traffic real-time information collected through geographic information systems (GIS) in order to achieve optimal vehicle routing in a stochastic transportation network. They took into account the fact that standard short-path algorithms such as Dijkstra's algorithm (section 2.2.4.2) are not effective in determining minimum short path on a dynamic stochastic network. The reason is that standard short-path algorithms cannot be computed as a simple path requiring the optimal route to be determined based on policy (Delling & Wagner, 2009).

Alazab et al. (2011) then drew attention to the fact that to be able to determine the best route to a destination, arrival time at the node would have to be determined by other factors such as distance, road works, driver's speed, other diversions and traffic congestion. To incorporate traffic routing into real-time traffic flow information, the team consequently developed their Web-based transportation routing application, integrated into real-time GIS services, and suggested modifying Dijkstra's algorithm to implement an optimal routing algorithm.

The researchers stipulated that they were able to reduce total costs and vehicle usage, and increase driver productivity during heavy congestion periods by implementing the following attributes (Alazab et al., 2011):

- *Node Labelling Method,* which is the shortest path obtained from the termination of the repetitive procedure
- *Network Representation,* which is the representation of stored data associated with arcs and nodes in a specific sequence
- *Selection Rules,* which are strategies for selecting nodes that are impacting on efficiency and speed
- *Data Structures,* which involves the manipulation of labelled nodes to support strategies

In a study on adaptive routing in congested networks, Cox, Jennings and Krukowski (2014) put forward a reinforcement learning approach for routing autonomous vehicles in a multilevel network on conventional roads, with relevance to operating, for example, a fleet of taxis. The strategy is an extension of Q-routing developed by these researchers for packet routing in communication networks. Q-routing is described as being similar to reinforcement learning, where the Q function is a matrix that maps states (S) and actions (A) to rewards (R).

$$S \ X \ A \ \rightarrow R$$

Q qualifies the time estimated at each entry that an agent takes to traverse the network from node x to node d via neighbouring node y. On traversing node x to y, an agent can update,

$$Q_x \, (d, y)$$

as follows:

$$Q_x\,(d,y) := Q_x\ (d,y) + a(s + t - Q_x\ (d,y)$$

d - destination node

x - current node

y - neighbour node of x

a - learning rate

s - travel time between nodes $x$and y

t - $min_{z\in\{neighbours\ of\ y\}}Q_y(d,z)$

Taxis use the information on the time duration of a specific leg of a journey to update prior node estimates. A table of estimates is saved for each node and turn direction to every probable destination node. Once updated the next optimal decision is sought at that node,

$$argmin_{y\in\{neighbours\ of\ x\}}\, Q_x(d,y)$$

An extended update to the Q-routing was proposed by Cox, Jennings and Krukowski (2014) with each taxi's path history stored in memory and used to update previous nodes. Each node in a taxi's history is updated. This information consists of the time it takes for a taxi to travel from one node to another, and the estimated time to travel from a particular node to the destination,

$$Q_{x^i}(d,y^i) := Q_{x^i}(d,y^i)\ +\ a(p_i +\ min_{y\in\{neighbours\ of\ x^j\}}Q_{x^j}(d,y) - Q_{xi}(d,y^i))$$

i – element in path history, $i\ \in \{1, \ldots, j$

j – last element in path history

pi – travel time between nodes i and j

To facilitate a congestion model, Cox, Jennings and Krukowski (2014) were of the opinion that no model was able to integrate traffic with throughput completely. They applied a sigmoid function representing the relationship between the number of vehicles on a road and mean velocity of the vehicles. Sigmoid function is often used for models designed to predict probability and has a characteristic 'S'- shaped curve. The authors allowed for a minimum velocity [V(1) = V_{max} and $lim_{x\to\infty}$] in their simulation to ensure that each vehicle was able to complete its route in a fixed time.

The main purpose of the congestion function was to demonstrate the algorithm performance and to derive a significant relationship between the distribution of vehicles and mean velocity on different road types by using a learning algorithm.

$$V(x) = V_{max} \frac{(1 + e^{\beta - \mu})(e^{\alpha - \beta x})}{1 + e^{\mu - \beta x}}$$

x is the number of cars on a road
β and μ are constants

The Q-routing process was evaluated against a search algorithm (A*) using an extension of the Dijkstra algorithm (section 2.2.4.2) to find the shortest path between two nodes. On receiving a customer, a taxi utilises its knowledge of road rules and geometry to apply the search algorithm (A*) in order to calculate its path from origin to destination. The taxi stores the results of this calculation in memory and uses the information to its destination. Although taxis have knowledge of all static system parameters, the locations of other taxis and the congestion on the roads are unknown.

In considering three test cases, the simulation revealed that, in multivehicle routing, Q-routing outperformed A* in a congested traffic network. A downside of the concept is the fact that Q-routing is unsuitable from a dimension perspective, as the Q-matrix grows with number of nodes at $O(n^2)$ (Cox, Jennings & Krukowski, 2014).

A study by Zhang, Rossi and Pavone (2016) focused on congestion, and involved routing and rebalancing a fleet of autonomous vehicles, providing on-demand mobility within a capacitated transport network environment. Capacity, in this instance, refers to the maximum flow-rate, and balancing signifies empty vehicle trips. They proposed routing and rebalancing algorithms for an autonomous vehicle fleet that, through intelligent routing, the operator could avoid congestion by means of a network flow model.

Their study introduced autonomous vehicles on conventional roads and addressed the concerns of other researchers (Templeton, 2015; Barnard, 2016) that any additional vehicles would disrupt present infrastructures. In order to avoid congestion, capacity constraints are implemented on nodes (representing intersections and locations) and road links, where vehicles travel at the speed limit at which the link is able to contain congestion. In the event that flow reaches a maximum, the flow-rate decreases to avoid congestion. The effectiveness of the network is dependent on whether the overall capacity entering each node is equal to the capacity exiting all nodes. Once customers have been transferred to their destinations in minimum time, vehicles would be rebalanced to realign with customer-demand and adherence to vehicle-flow forced constraints. Furthermore, the authors postulated that their autonomous mobility-on-demand concept would reduce travel costs, pollution, and demand for parking (Zhang, Rossi & Pavone, 2016).

International Business Machines (IBM) proposed that by 2025, autonomous vehicles would carry self-healing mechanisms to reduce human intervention and maintenance further. In accord with IBM's suggestion, Mahmoudi and Griffor (2018) demonstrated by means of testing and simulation how enhancing safety for autonomous vehicles is done through robust in-vehicle networking architectures that incorporate in-built resilient mechanisms' particularly with automatic emergency braking and lane centring. By utilising software defined networking (SDN), once a primary link fails, a standby link automatically takes over. SDN program features allow manipulation of routing tables and routing engines, and the implementation of features to protect in-vehicle networks from failure. The researchers claim that the proposed software defined in-vehicle Networking (SDIVN) does not increase overhead when compared to legacy In-vehicle networks (LIVNs) that are not programmed for failure conditions.

In their study Halba, Mahmoudi and Griffor (2018) demonstrated how the SDN application populates the SDN switches, with flow and group tables, with essential rules and actions to manage link failover, installed after running the application. Halba, Mahmoudi and Griffor (2018) stated that safety issues are critical when an electrical control unit (ECU) or in-vehicle bus becomes inoperative. They stipulate that the SDN networking paradigm manipulates the separation of network programmability and the control and data plane, so that complex features such as fine-grained forwarding, quality of service (QoS), load balancing and failover are enabled (Belyaev & Gaivoronski, 2014; Bianco et al., 2010; Tomovic, Prasad & Radusinovic, 2014). A threat to LIVNs's controller area network (CAN) would be critical, as it constitutes all data transmitted between components and received by all ECUs (Tuohy et al., 2015). This indicates that access to one ECU could change the behaviour of a vehicle. In contrast, the SDVIN design is set to switch from a failed path to an effective one.

Figure 2.19 shows CAN1 ECU and CAN3 ECU generating various messages with different frequencies, each message carrying a precise automotive function. For example, CAN3 hardware detects obstacles in front of a vehicle and sends messages to the antilock braking system (ABS) at CAN4. Based on road conditions and information from the obstacle detection ECU located at CAN3 ECU, the ABS comes into operation. CAN1 is a sensor determining the road conditions and sends messages to an alarm system at CAN2 ECU.

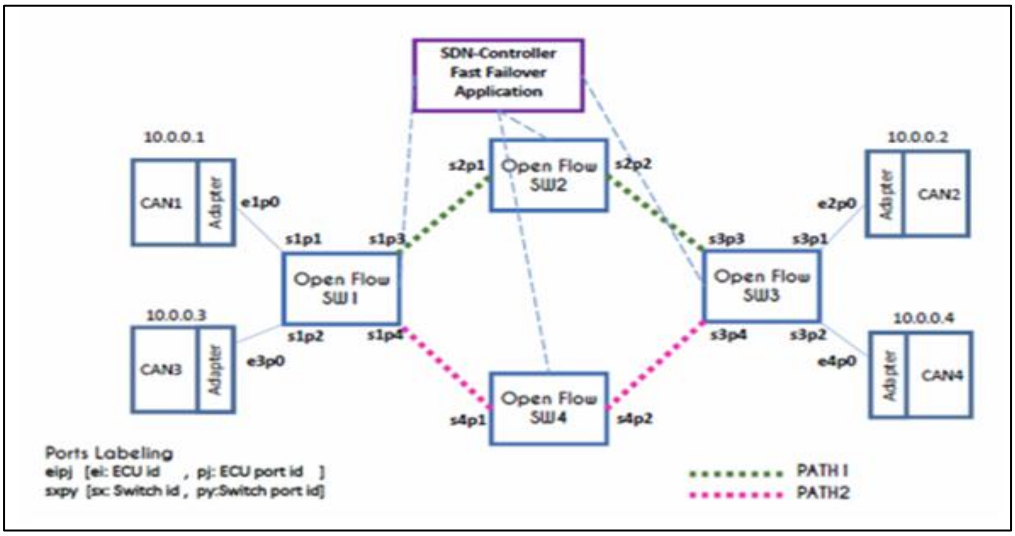


Figure 2.24: ECUs exchange in-vehicle messages and control the SIDVN (Source: Halba, Mahmoudi & Griffor, 2018:52)

The unicast characteristic of SDIVN allows for rule-generated messages generated by the application, running in the SDN controller, to be sent to their specific destinations, decreasing the risk of bus outage or replicating any message. The researchers also concluded that should link failure occur, the fast failover mechanism would not cause any noteworthy delay overhead but would guarantee timely message delivery and message frequency integrity (Halba, Mahmoudi & Griffor, 2018).

A source routing mechanism developed by the Internet Engineering Task Force (IETF) Source Packet Routing in Networking (SPRING) working group, called segment routing, is an alternative suggestion to the shortest-path-first strategy (Reuter & Cordeiro, 2016). The IETF SPRING working group defines a segment as "an instruction a node executes on the incoming packet" which, together with its segment identifier (SID), is advertised in the routing domain, assisted by the interior gateway protocol (IGP) (Reuter & Cordeiro, 2016:35). With the help of these extensions, segment routing signalling information can be executed by IGP protocols such as OSPF and IS-IS. Their source routing method provides interoperability with present non-source routed networks. Packets are forwarded based on routing information attached to the packets themselves, rather than stored at the intermediate routers. Three kinds of segments are introduced (Reuter & Cordeiro, 2016), namely:

i) *IGP-Node Segments,* which have a unique SID and advertise the segment via the IGP. All nodes register entry in their FIB, with the IGP algorithm determining the shortest path. The incoming node can impose a source route on a packet by stipulating another node to be traversed, which is achieved by prepending the corresponding node to that packet. Prepending is defined as adding something to the beginning of something else (Lexico.com, 2020). A prepending label specifies the relationship of a packet to a group of IP packets forwarded over the same path in the same way. In Figure 2.20, Node R advertises its node SID 70 and, by prepending this

Node S, instructs incoming packets to the domain to reach Node R via the shortest path that it has investigated, either {S, A, B, R} or {S, C, D, R}. Intermediate packets merely swap the SID. As the last node on the route closest to R (B or D) is also directly connected to R, it can remove the SID because it is no longer required (Reuter & Cordeiro, 2016).

ii) *IGP-Prefix Segments* are specific nodal segments that advertise a prefix of full address length SID, advertising a route within the network and therefore have global relevance.

iii) *IGP-Adjacency Segments* have only local unique SID installed to the RIB of remote nodes and impose a route over specific links.

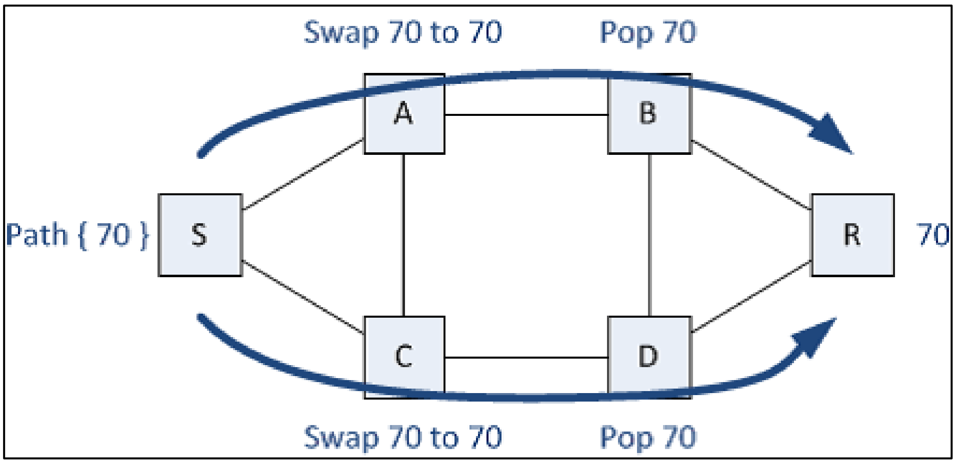


Figure 2.25: Node segments
(Source: Reuter & Cordeiro, 2016:33)

The SPRING working group maintain that source routing is destined to become a major technology used to optimise traffic flow in large-scale networks, with segment routing as a technique to accommodate this. Segment routing, however, poses a challenge, as attacks from within a segment routing domain will have to be considered (Reuter & Cordeiro, 2016).

As noted from the above studies, research has shown that AV complexities include localisation, map-building, route planning, route tracking and sensors for detecting and classifying objects that need to be avoided. However, in considering obtaining simplification and reduction of data when state information is minimised, Khayyam et al. (2020) determined that reducing data limits the vehicle’s navigation and manoeuvrability. The prescribed routing inputs of packet switching were considered by the researcher as better equipped to allow for minimising information through abandoning redundant data as well as reducing duration of retentive information in their buffer regions once employed (section 2.2) (section 4.4).

2.7 Theory defining the study

In order to qualify the investigation, the author delved into theories that could support the direction of this research, rooted, as it is, in an association with computer science and the theorists behind the technology. A theory is a structured set of logical principles that explain a specific phenomenon by presenting and expressing the relationship that exists between the phenomenon and other phenomena (Zikmund et al., 2010:39). Artificial intelligence (section 2.4) and ML (section 2.3.2) theory were dealt with earlier, and incorporated into the 'mechanics' of the concepts under their respective headings above. Reference is made here by the researcher to the alternative theories applicable to this research. Of the various theories applicable to Information Systems (IS) support, and to an extent, both the positivist and constructivist aspects of this study, only the two theories most pertinent in outlining the course of this study are discussed below, namely: i) design theory, and ii) systems theory.

2.7.1 Design theory

Human culture, expressed in the art of living and the production of tools, is established mainly through design as demonstration of man's ability. Formulation of design within the complexity of IT systems is based on the evolution of continual refinements to traditional design approaches (Östman, 2005) and based on the theory of design.

2.7.1.1 Seminal work

Herbert Simon (1996:13) recognised the need for a science of design that required a certain order to attain goals and to be expressed as theory. Walls, Widemeyer and El Sawy (1992) extended this concept into IS design theories (ISDT), while Gregor (2006) highlighted design and action theory as the theory of 'how to do something' by designing and developing an artefact. These researchers proposed that a design problem is encountered when humans want to change or improve a complex circumstance involving workings that include aspects of the human and natural environments. They also maintained that design involves assessment and ethical considerations.

More recent research by Hevner and Gregor (2013) reveals how Design Science Research (DSR) can demonstrate and contribute to our knowledge base. DSR is divided into descriptive knowledge, Ω, which is 'what' knowledge about natural phenomena and their associated laws, and prescriptive knowledge, Λ, which is 'how' knowledge of human built artefacts (Hevner & Gregor, 2013). The intention is to build a foundation of knowledge to evaluate the idea of a new artefact for a proposed study. With a belief that things are either made from something else of built on a previous idea, research contributions vary according to the project's starting point of knowledge. These starting points are as follows: Invention – this requires new solutions for new problems; Improvement – new solutions for known problems; Exaptation – known solutions extended to new problems, and Routine design – known solutions for known problems (Hevner & Gregor, 2013).

2.7.1.2 Definition

Design is described as a controlled creative action, while ISDT is defined as “a prescriptive theory which integrates normative and descriptive theories into design paths intended to produce more effective information systems” (Walls, Widemeyer & El Sawy, 1992:36).

2.7.1.3 Characteristics

Similar to systems theory, design theory alludes that the components are to be viewed as part of a whole. Design originates in products already in existence and forms the basis from which further research can be undertaken to improve on this information. The principles of design theory, modified by Gregor and Jones (2007:323) from the original by Codd (1970), are as follows:

1. *Purpose and scope* is stated relevant to the research
2. *Principles of form and function* incorporate fundamental constructs
3. *Artefact flexibility* is taken into consideration by allowing adaptation and change
4. *Testable* suggestions to verify whether concepts or theory is feasible
5. *Justification* of the theory by offering an explanation for the design
6. *Principles* define the process for implementing the method or process
7. *Explanation* of the artefact demonstrated in a physical working system

2.7.1.4 Advantages

In constructing an artefact that works based on existing theory and knowledge, design theory in information systems allows for identifying deficiencies in constructed prototype software systems and developing creative solutions to rectify them (Gregor & Jones, 2007).

2.7.1.5 Disadvantages

A disadvantage with design theory lies with the fact that the larger the number of components, such as an increasing number of devices connected via the IoT, the more design alternatives there are, and the more adaptability measures need to be made, with new connections between components (Gregor & Jones, 2007).

2.7.1.6 Extent of application to this study

Design theory can exist in the abstract world to include algorithms and models, which once instantiated could become part of the physical real world. The researcher’s interest was initiated by Gregor and Jones’s (2007) theory on the phenomena of interest for design research, where material artefacts (instantiations) are used as part of the process, while abstract artefacts (theories) are communicated as constructs, methods or models. The development of the design for this study guided the building of instantiation – the development of a transport system by using abstract artefacts that include methods, models and principles. IT artefacts are vital outputs of IS research, and design science research incorporates the following artefacts:

- *Constructs* – meaning the vocabulary and symbols that define and communicate problems and solutions
- *Models* – which are statements expressing relationships between constructs representing the real world
- *Methods* – the procedures implemented towards a solution and 'best practice' approach, such as through the development of formal algorithms
- *Instantiations* – which are the implementation of models, constructs and methods to develop a prototype or an add to an existing system in order to demonstrate the effect of these implementations on people (Hevner et al., 2004; March & Smith, 1995; Nunamaker, Chen & Purdin, 1991; Simon, 1996). This would inform researchers of the interactions between people and the technology that would need to be developed to improve the current human transport arrangement

Taking DSR into consideration, this research focuses on the improvement of a situation through new solutions for known problems and draws from understanding the problem and building innovative artefacts to serve as a solution.

2.7.1.7 Limitation of theory relative to this study

The author attempted to limit the number of components required to incorporate in the simulation experiment by focusing on the basic requirements of the working system.

2.7.2 Systems theory

The 19th century and early 20th century were perceived by Ludwig von Bertalanffy, a German biologist, as 'chaotic', since it was believed by people at the time that aspects of the living world were the product of chance (Laszlo & Krippner, 1998). The gradual emergence of new disciplines such as cybernetics IT and systems analysis provided a different outlook on this worldview, thus changing the trend of scientific thought. It is acceptable that assumptions and techniques may differ from one another and it has been established that all these assumptions and techniques are concerned with "systems", "wholes" or "organisations" that require a new approach, and systems theory can be considered as a field of inquiry (Lilienfeld, 1978:7-8).

2.7.2.1 Seminal work

Ludwig von Bertalanffy advanced the systems theory in the 1940s, noticing a need to integrate various sciences in his work. Later Russell Ackoff (1981) incorporated technology into physical science and biology in the definition, in view of Norbert Wiener's introduction of the cybernetics concept in 1949.

2.7.2.2 Definition

Systems theory is defined as the trans-disciplinary study of the abstract organisation of phenomena, independent of their substance, type or specific characteristic existence. Systems theory investigates principles common to all complex entities, as well as the (generally mathematical) models, which can be used to describe these entities (Von Bertalanffy, 1973). Based on Russell Ackoff's suggestion to modify systems theory's definition, it is now explained as a "complex of interacting components together with the relationships among them that permit the identification of a boundary-maintaining entity or process" (Ackoff, 1981:15-16). This modification is more appropriate to this study due to the complexity of interacting components and relationship among them via IoT on the AVs routing process.

2.7.2.3 Characteristics

To understand the principles of systems theory, two important aspects associated with the theory – reductionism and holism (also referred to as a systems approach) – need to be explained:

- *Reductionism* asserts that in order to understand a phenomenon, we need to study the properties of its individual parts; while
- *Holism* affirms that we cannot understand the behaviour of the whole by merely studying the behaviour of its various components (Laszlo & Krippner, 1998).

A system includes two or more interrelated elements with the following properties:

- Each element affects the function of the whole;
- Each element is affected by at least one other element within the system; and
- Subgroups of an element also have the two above properties (Ackoff, 1981).

Holism refers to the general organisation and relationship of the different components of a system functioning as a whole, since the behaviour of the system is seen to be independent of the individual properties of the elements making up that system (Bode et al., 1949).

Cybernetics was subsequently introduced into systems theory, and is described as the study of the communication and control of a reaction in living and lifeless systems (organisms, organisations and machines), and in combinations of these systems (Ackoff, 1981). Emphasis is on how anything (digital, mechanical or biological) that controls behaviour, processes information and responds to information, and how it changes or can be changed to improve the system. This is referred to as *closed systems thinking*, enabling diverse disciplines to communicate, and a unity of science and organisation can be reached at all levels. Conflicting philosophies often arrive at similar conclusions; therefore, unconnected fields of study are important for understanding reality.

The interchange between the system and its environment is referred to as *open systems theory* as the systems constantly evolve and adapt to the needs, resources and threats of their environment. Quantitative values are expressed in physical science and non-physical concepts are explained in qualitative terms; however, biological laws can be reduced to physical ones and vice versa (Carnap,1934), and as mentioned previously by Trochim and Donnely (2008) (section 1.8.1). Reductionism, in this respect, refers to the principle that biology, behaviour and social sciences are ultimately reduced to concepts of the physical.

2.7.2.4 Advantages

Systems theory drives the following positive characteristics:

- Allows trans-disciplinary exploration of relationships between perceptions and conceptions in the integration of the sciences
- The closed systems thinking of cybernetics permits diverse disciplines to communicate with each other
- Open systems theory focuses on environmental influences and enables research to adapt to situations that arise and to respond to threats (Laszlo & Krippner, 1998)

2.7.2.5 Disadvantages

The cybernetic closed systems model as the intellectual hub for customary cost and quality management approaches could lead to an increase in entropy or havoc, resulting in inevitable destruction of the model. Regulation of the model occurs only after the concept has been developed. Cost and quality are viewed as internal variables, managed within the confines of an organisation. This could ultimately lead to incurring excessive costs (Wiener, 1988).

2.7.2.6 Extent of application to this study

From the onset of the theoretical stage of the research, systems engineering was applied to the building of the simulated theoretical model, and the study incorporated concepts and principles applying to all systems (Von Bertalanffy, 1973). Systems engineering includes “scientific planning, design, evaluation and construction of man-machine systems” (Von Bertalanffy, 1973:91).

It is not correct to serve only one aspect of the theory (either closed or open), given the vast scope of the system under study. The realm of IoT characterises an interconnection of “mini-systems” into a whole and this is taken into account in line with the interaction of relationships between components, as described by Ackoff (1981:15-16). The researcher took cognisance of the notion that systems may not be studied entirely in isolation because of environmental factors, including competitiveness, among others. While it was considered that thorough knowledge of individual components of a system was important to the unified whole, this was not possible because of the study's time constraints; but their functions, to a lesser degree, were integrated into the study. This indicates that relevant and even unconnected fields of study could be incorporated into a 'mix' (both closed and open) of the theory.

A problem question in respect of this research (section 1.6.2) refers to the scientific adaptation of systems efficiency to reach a level that would benefit society in the process of applying human engineering across disciplines. Consideration was given to the economic impact of introducing the concept that was the focus of this research, as well as its impact on the employment factor.

2.7.2.7 Limitation of theory relative to this study

A limitation of the theory is that the complexity of a system cannot be explained through employing one modal point of entry, even to an in-depth extent. It was understood that an assembly of interrelated and interconnected components result in the behaviour of the whole being different, for a system cannot unshackle itself from the relationship of its parts (Von Bertalanffy, 1973).

This study in real-life integrates many fields, from systems engineering and topography to biological and social sciences, which were taken into account together with related connectivity via IoT. Given the scope of the influencing factors, the researcher focused on the one specific area of the whole that dealt with the routing of AVs. In focusing specifically on the routing aspect, the nature of interdependence of components becomes somewhat neglected.

2.8 Summary

In Chapter 2, the researcher provided an overview of the research by describing in-depth components and introducing terminologies that contributed to the ultimate purpose of this research. The focus of the research was on the routing of autonomous vehicles in a manner analogous to the concept of packet switching in the inclusive realm of IoT VANET technologies. The chapter offered insight into existing and proposed routing algorithms used to direct traffic, and highlighted the problems in these areas, in order to qualify the development of an appropriate algorithm for the simulation model that describes the real world of autonomous routing of vehicles.

The conceptual framework of the research is grounded in design theory, viewing components as a unified whole; and the research is grounded in systems theory, which allows diverse disciplines to collaborate. The literature elaborated on new vehicular technologies in IoT, how these technologies are set to enhance the business realm and provide a mechanism to accommodate employment prospects and a better quality of life.

In Chapter 3, the researcher will provide an in-depth description of the research design and methodology employed in this investigation, including the culmination of philosophy, strategy and data analysis.

CHAPTER 3: RESEARCH DESIGN AND METHODOLOGY

3.1 Introduction

To bring this study into an existing body of knowledge, literature was cited from numerous sources. Available research on the specific core of the topic is limited; nonetheless, this research draws on what has already been recognised in the discipline. While the study is inherently embedded in the field of IT, it is inevitably interconnected to environmental sustainability and human wellbeing (Gregor & Jones, 2007) (section 2.7.1.4).

The research design and methodology employed in the study is described in this chapter in order to justify the research and to explain the framework and theory on which the research was founded. Designed by the Michigan Simulation User Group Technical Committee on Simulation Methodology, the process of design and methodology identified for this research is commonly used in the building of simulation models (Ulgen et al., 2014).

3.2 Real-system problem defined

Research is the step-by-step process of acquiring knowledge through various means (Welman, Kruger & Mitchell, 2005), which generates understanding that contributes to solving problems and producing new ideas that lead to further research in the quest for constant improvement (Clough & Nutbrown, 2012). Alternatively, research design is interpreted as choices made by the researcher in preparation for, and progression of, the study in order to attain a predefined goal (Fouché & Schurink, 2011, Mouton, 1996)**.**

The goal of the experiment was to develop a routing system for autonomous vehicles analogous to the packet switching concept (sections 1.3.1 & 2.2). A simulation was considered, taking into account the notion that this simulation could accurately mimic a complex phenomenon, since it contains a considerable amount of information about the phenomenon. Computer simulation provides the most applicable method employed to demonstrate the real-life environment described in this research, since it can realistically model experiments and observations in a manner that is simplistic, comprehensive and unambiguous (Olivier, 2011) (section 1.8.2).

A simulation of the theoretical active model of the experiment under study, namely the routing of autonomous vehicles, is explained herein. In this chapter, the researcher describes the preliminary process of constructing the prototype, as illustrated in Figure 3.1, using algorithms analogous to packet switching routing protocols and based on theory for the perception of the new model.

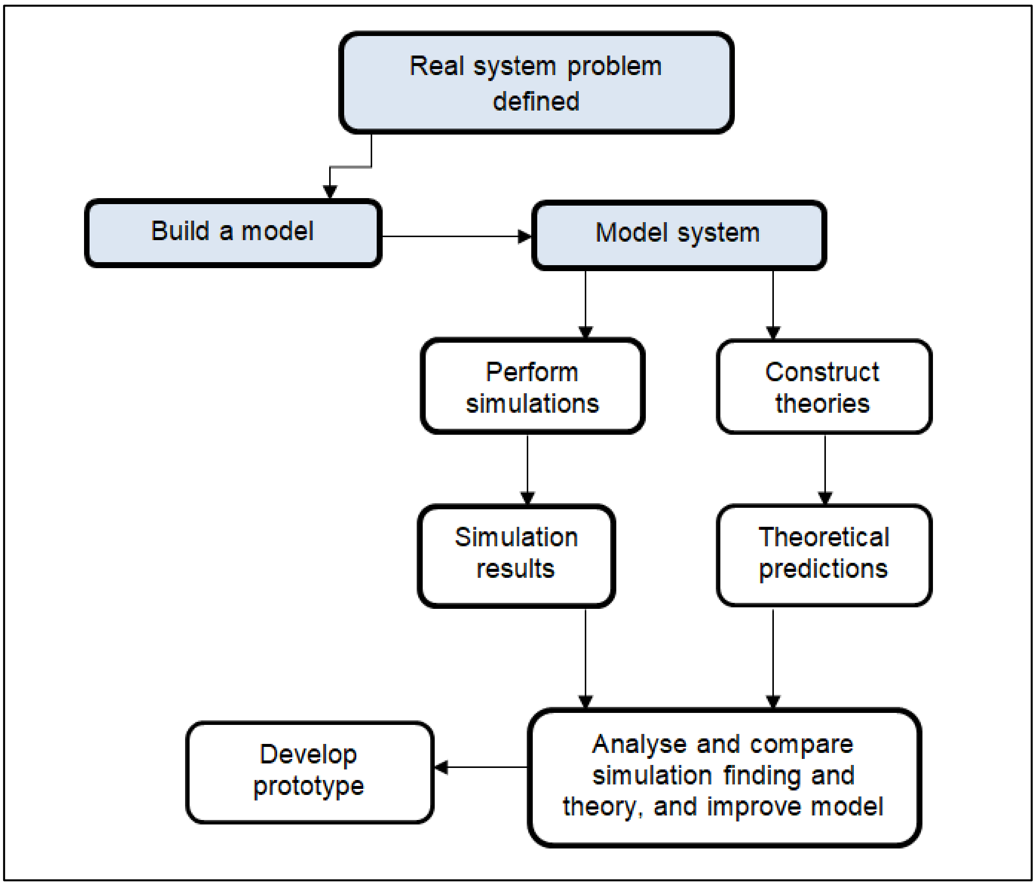


Figure 3.1: Preliminary process (highlighted) of constructing proposed computer model (Section 1.8.2)

3.3 Build a model

In order to construct the simulation model it was imperative to meet the proper conditions of the physical system. To simulate realistic input and operating conditions, considerable care and attention to detail were essential. The relationship between the base model, the modelled physical system, and the simulated algorithm is called realisation, and enables the simulation to be an instrument for understanding real life phenomena (Norton & Suppe, 2001).

In deciding to build a simulation model, the following steps drawn up by the Michigan Simulation User Group Technical Committee on Simulation Methodology (Ulgen et al., 2014) were considered:

Step 1: Decide on static, discrete, continuous or combined discrete-continuous modelling

Step 2: Determine the elements that drive the system

Step 3: Determine the entities that should represent the system elements

Step 4: Determine the level of detail needed to describe the system components

Step 5: Determine the graphics requirements of the model

Step 6: Identify the areas that utilise special control logic

Step 7: Determine how to collect statistics in the model and communicate results

3.3.1 Step 1: Deciding on static, discrete, continuous or combined discrete-continuous modelling

An algorithm is defined as a systematic method of accomplishing a goal (Merriam-Webster, 2020). In the process of altering existing algorithms characteristic of specific software and according to a certain modelling arrangement, the events and variables engaged to describe the system are discussed below.

Discrete event simulators (DES) require simulation software for process analytics to allow for the creation of virtual process replications of real-life operations. These replicas consist of two main categories, namely: *static* tools and *dynamic* tools; the differences between the tools are located in the model building, simulation interaction, analysis, and connectivity (CreateASoft, Inc, 2018).

Static simulation represents a system at a particular point in time, initially developed for testing engineering systems. The concept is to develop a computer-based analytical model to predict the behaviour of a system. Static tools provide animation during the simulation but do not physically interact with the model, as metrics need to be collected and coded before the model is run. This means that analysis of the simulation performance can only be reported on after the simulation, as data feedback during simulation is difficult. Although time is not a factor in this model, it serves as a reliable technique that goes beyond mathematical clarification (CreateASoft Inc, 2018).

In the *dynamic DES* model, the values of the state variables change at discrete (or distinct) instances, while remaining constant between event times. This means that each event marks a change of state in the system, while no change of state is assumed to take place between consecutive events, thus allowing the system to jump from one event to another (Robinson, 2004). A systems state is a set of variables that encapsulates the relevant properties of the system and can be represented mathematically over time **S(t)** (Banks & Carson, 1990; Ulgen et al., 2014). A DES model makes it possible to simulate events in time that are of the logic-test and fault-free types. This implies that a series of events can be managed and sorted within a certain timeframe, and processed before activating a new event. Typical DES mechanisms could therefore be applied to simulate the routing of autonomous vehicles by means of packet switching and discrete equations could be directly implemented in an algorithm (Ulgen et al., 2014).

The state variables of a system constantly change in a ***continuous model*** in both time and difference to reflect changes in quantities; or differential equations that may contain a function in one independent variable and its derivatives, are used to describe the state variables. Continuous simulation is activity-based as the simulation continuously tracks system dynamics over time, which is broken up into small time segments (Robinson, 2004). The system state updates according to the activities taking place in each time segment and as a result is slower than discrete-event simulation.

In a ***combined discrete-continuous model***, both types of variables are employed to describe the system, which is neither entirely discrete nor entirely continuous. Based on the objective of the study, a discrete model may be used for a continuous system or, similarly, a continuous system may be used as a discrete model (Ulgen et al., 2014). For example, the movement of the vehicles may be described in terms of continuous variables, while pickup and drop operations may be described as discrete events.

Taking the above into account for this research, the use of a ***combined discrete-continuous model*** was also considered a possibility because of the following factors:

i) Discrete-event

a) *System events* -> The events on a high level, and of significance to the study were:
   - Hops, which can be described as the conveyance of a packet from its immediate source to the very next device en route to its destination
   - Route adjustment, which occurs in network redundancy where an alternate route is selected in the event of network failure (Ulgen et al., 2014).

b) *System entities* -> represented by the waypoints (encompassing all endpoints) and routing devices are key factors and integral to the system design (Ulgen et al., 2014). The different states of the simulator could imply the number of hops to destination, the destination availability, or route availability. A time factor would depend on each state at a specific time, with the following states outlining the change in state, relative to the above events:
   - **Number of hops to destination:** Influenced by the hop event; the number of hops was expected to decrease by increments of 1 for every router or switch arrived at by a packet. Hence the number of hops would be indirectly proportional to distance
   - **Destination availability:** A packet may be in a *pending* or *waiting* state if the destination is temporarily unavailable, to which point it would proceed once the destination became available
   - **Route availability:** In a redundant network, should a failure occur along a predetermined route, an event can be triggered that adjusts the routing path in order to make use of an alternate route. Consequently, the event would warrant a change in the hop count state (Banks & Carson, 1990; Ulgen et al., 2014)

c) *System variables* -> represented by:

- *Distance* **–** which can be quantified by hop count and *time between hops*
- *Traffic* – which has a direct impact on the *time-in-transit* (Ulgen et al., 2014)

A more efficient method, developed later in discrete-event simulation, is the ***three-phased approach,*** where the first phase jumps to the following chronological event, the second phase executes all events (B events) that unconditionally occur at that time, and the third phase executes all events (C events) that conditionally occur at that time (Pidd, 1998). This approach has been adopted by numerous commercial software packages. The order of three-phase discrete-events in the context of the three-phase simulation was considered as follows:

- **Phase 1:** Output interface determination – check each output interface sequentially in determining the correct routing path
- **Phase 2:** Unconditional events, for instance, congestion control and system health checks
- **Phase 3:** Certain network technologies, such as voice-over-IP (VoIP), make use of packet prioritisation, which, by definition, only allows certain traffic through an interface. The same principle could be applied to routing vehicles, should the need arise

ii) Continuous-event

The continuous-event attribute of the model could be taken as the interaction of input/output router interfaces. Output interfaces in a conventional model are considered constants, and are thus unchangeable given their 'static' configuration. Output interfaces can however change automatically if configured with the appropriate protocol such as the EIGRP protocol, which is designed to detect changes in the state of each interface automatically, together with the interfaces of neighbouring networks. The routing table is then updated with this newfound information (sections 2.2 & 2.2.4.1).

Output interfaces are directly dependent on the input variable interface; for example, when a destination network address is analysed upon the arrival of a packet, it will determine the output interface it needs to traverse. There are several considerations during this process, such as the router's IOS, which coordinates the invocation of packet forwarding events within a predetermined timeframe, with each interface selection method coinciding with a specific event assigned to it (Banks & Carson, 1990).

The final procedure towards building the model was to opt for the use of open-source software (OSS), as OSS is a form of computer software with unrestricted access to the source code, where the copyright owner permits users to modify and distribute the software

(St. Laurent, 2008). Open source software is usually easier and cheaper to obtain than proprietary software, and with the adoption of available implemented standards, it has resulted in increased usage. It has been recorded that technical users of open source far outweigh commercial users of the software (Wigan & Drain, 2002).

The researcher took cognisance of reasons provided by the Open Source Business Conference survey (Guseva, 2009) as to why individuals or organisations choose open source software, as follows:

1. Lower cost – particularly of marketing and logistical services
2. Security – continuous follow-up and updating
3. No vendor 'lock in' – easier to obtain than proprietary software
4. Enhanced quality – many users keep abreast of technology developments
5. Reliable – a myriad of independent programmers test and rectify problems
6. User friendly – tools support development of product and process

Constructed in similar fashion to a mathematical model (section 1.8.2), the Psimulator2 is a graphical network simulator developed by researchers at the Faculty of Information Technology of the Czech Technical University in Prague for learning the basics of IP networks (Linkletter, 2016). Psimulator2's specific open-source simulation software features also conformed to the desired building steps described herein, and it is considered an appropriate simulation model for the purposes of this research. In addition, the model's characteristics allow for easy design adaptation and manipulation.

In summarising the justification for applying discrete-continuous methods, the researcher deliberated on the fact that Psimulator2 requires manual configuration for entry to network routes due to its static protocol. As coding is done prior to running the system, it remains a reliable method to model a scenario where human lives are concerned. It was noted that continuous simulation continually tracks the response of a system, whereas a discrete event moves forward by change taking place at a certain time, such as at a particular waypoint or node where everything that needs to be done is carried out (CreateASoft Inc, 2018). A continuous variable exists in the movement of vehicles with their independent IoT connectivity that continually 'informs' the system, whilst pick-up and drop-off procedures reflect as discrete simulation in their programming that changes the behaviour of vehicles at each event. These events equate to system entities such as waypoints, endpoints and routing devices operating independently as discrete mechanisms that respond accordingly. In this environment, discrete simulation software allows virtual replication of real-life situations by means of static tools and dynamic tools (CreateASoft Inc, 2018).

3.3.2 Step 2: Elements that drive the system

With a minimal subset of networking instructions, only static routing is supported by Psimulator2. A static protocol requires manual configuration and static routing, as described in the literature, which means network routes are manually entered into a router's route table, stipulating the next hop router IP address in forwarding traffic to the next destination (Osterloh, 2001) (section 2.2.3). Information is received in the routing system via all connectivity incorporated into the IoT that include elements such as ubiquitous wireless communication, machine learning, real-time analytics, embedded systems and LiDAR, which facilitates the measurement of distances through photo-detection (section 2.3.2.1).

3.3.3 Step 3: Determine the entities that should represent the system elements

In determining the tools available in the simulator's cache and the simulation language for manipulation and improvisation, the Psimulator2 simulator was found to be the most suitable model (section 1.8.2). Initially written in Java (version 7) as a multiplatform application, it is also supported in Java (version 8) (GitHub Inc, 2008). Psimulator2 comprises the necessary networking structures, such as routers, switches and links, required for designing, configuring and implementing a basic network topology, with commands that typically feature in Linux and Cisco routers, which are configured by means of the command line over telnet protocol (sections 2.2.2 & 2.2.4). The telnet client is included in the GUI, or alternatively, a different client can be used, such as PuTTY or Linux telnet.

It is of interest to note that Java is a language designed to run swiftly; it is fully portable, the Java environment is also designed to be object-oriented, and is generally used in transport modelling initiatives (Hornick, Marcadẻ & Venkayala, 2006). Transport modelling, in this instance, refers to characteristics such as interchange of data between models or different conceptual data structures in different models. Systems that support Java include Windows, MacOS and Linux. The researcher decided that this study would operate Java on the Windows platform.

As mentioned in Step 1 above, system entities encompass waypoints that include all endpoints as well as the routing devices, which were the prime factors in this research and which are intrinsic to system design. The integration of IoT technologies in the realm of V2V and V2I involving exchange information and autonomous control required a model that allows an operation that permits entities to be independent of each other in finding a solution. Accumulated data would have to be extracted and analysed by the objects by means of AI, and the objects were expected to identify defects and detect changes in the environment and/or the sensors, and to respond appropriately as 'trained' to do so through ML and achieve AI in their independent decision-making and subsequent performance (section 2.3.2). Tools were, therefore, required to demonstrate that a chain of events during a certain time needed to be transmitted to certain objects by other objects and that solutions would be effectively applied, either by means of self-determined decisions or by means of a collaborated set of instructions (Alippi, 2014; Moore & Lu, 2011:6) (section 1.3.1).

As position and reliability were considered factors critical to the system design, the need for more algorithm adaptations was contemplated in order to connect components in a multi-threaded configuration (section 2.7.1.5).

3.3.4 Step 4: Determine the level of detail needed to describe the system components

The researcher determined that detail would be added to the model in stages, beginning with a basic macro level model of the system, namely a roadway infrastructure and characteristic environment. Simple commands available to generate traffic would be entered simply to allow for clarity in inspecting ARP messages and internet control message protocol (ICMP) messages (section 2.1). ICMP is a TCP/IP network layer protocol service offering troubleshooting, control and error messages. The Psimulator2 characteristically allows for expansion or 'building on', and permits the simulation of incremental procedures in extremely precise split-second timing.

Care was taken to ascertain that the amount of detail introduced met the objectives of the study sufficiently to sustain the validity of the model, while at the same time eliminating unnecessary detail that would otherwise serve to clutter and to confuse.

Challenges, such as redundancy and the security of the system, were taken into account through the simplicity of the system's operation. The functioning interdependent/autonomous systems in operation were not permitted to prioritise functionality as the researcher incorporated redundancy and security in the design (sections 2.3.3.2 & 3.3).

An alternative routing algorithm for shortest path selection from source to destination was also considered in place of Dijkstra's algorithm (sections 2.2.1 & 2.6.2).

3.3.5 Step 5: Determine the graphic requirements of the model

Animation requirements (dynamic and static) were determined as vital tools in the simulation process and needed to correspond appropriately to what the researcher intended to depict. Modifications, new operating systems and instruction set architectures were also considered.

The Psimulator2's network topologies designed using the GUI, which is the front-end component of the simulator that also views the transmission information of packets, was considered sufficient to fulfil the graphic requirements of this research. A simulated virtual network was swiftly created using the GUI and was designed to capture and display packets in a virtual network, with the virtual network able to connect to the real network. The emulator, replicating the real-life situation, was operated by the back-end component and worked independently of the front-end, while it controlled the internal functions of the simulation. Communication between components and other nodes was established via the back-end emulator only once the design was completed. The front-end and back-end were then expected to work together to run a simulation (Churchman, 2017; Law & Kelton, 1991) (section 1.8.2.1).

3.3.6 Step 6: Identify the areas that utilise special control logic

Complex control logic was taken into account given the multiple-entity characteristic of the system; however, for the purpose of this research with limited components, it was not deemed necessary to take any such action.

3.3.7 Step 7: Determine how to collect statistics in the model and communicate results

Winsberg (2010) conceded that successful simulation studies utilise a variety of techniques to draw inferences from computed numbers. This means that in order to make simulation results reliable, considerable expertise is required. A quantitative model functions on measures of quantity and uses physical values such as amount and size. Further to this, the IFAC SAFEPROCESS technical committee views a quantitative model-based approach as "one in which static and dynamic relations between system variables and parameters are used to create a model of a system using quantitative mathematical terms" (Isermann & Balla, 1997:710).

Results were to focus primarily on the back-end output as it routes a request from source to destination, together with information stored in routing tables. Information was logged in the start-up console for recording purposes and was a standard feature of the simulator. Configuration amendments and real-time feedback were included when operating the routing animation feature, which, in turn, could have provided useful information on how a packet is routed. The routing information was of particular interest because of the visual display of next hop addresses, time in milliseconds and pertinent network information.

System variables would be conveyed to indicate the following:

- *Distance* can be computed by hop count and refers to *time between hops*
- *Traffic* density directly influences the *time-in-transit* and consequential *time of arrival* (Ulgen et al., 2014)

Statistical data collected and results were to be communicated effectively and displayed in the form of tables and graphs.

3.4 Model system

The model system describes the manner in which the above were incorporated into the system to enable the simulation to be performed using the multi-faceted Psimulator2 (section 3.3, Step 1). The Psimulator2 project was sourced from the open-source portal, GitHub Inc (2008) to run in a Windows environment that supports Java version 7. The model was initially made available from a ZIP archive in Google Code and the files were unpacked into a folder using the Archive Manager. The simulation model required:

i) Java to run in Windows from Oracle Java web site; and

ii) Installing a telnet client compatible with Windows, such as PuTTY telnet/SSH client in order to connect to each virtual node once simulation commenced and configuration and commands were entered. From the basic network diagram depicted in Figure 3.2, the researcher proceeded to gradually construct the simulation model, applying the step-by-step process described above (section 3.3).

In order to build the model, the Psimulator's architecture was taken into account. The front-end was started and a network topology, using the GUI, was created with nodes and links. The topology represented destinations and points of origin by way of: (i) conventional homes, metropoles or waypoints; (ii) wireless homing beacons, which replaced both routers and switches; and (iii) network cables connected to each component, which represented a road or guideway. These substances in reality all form part of the network connectivity of IoT.

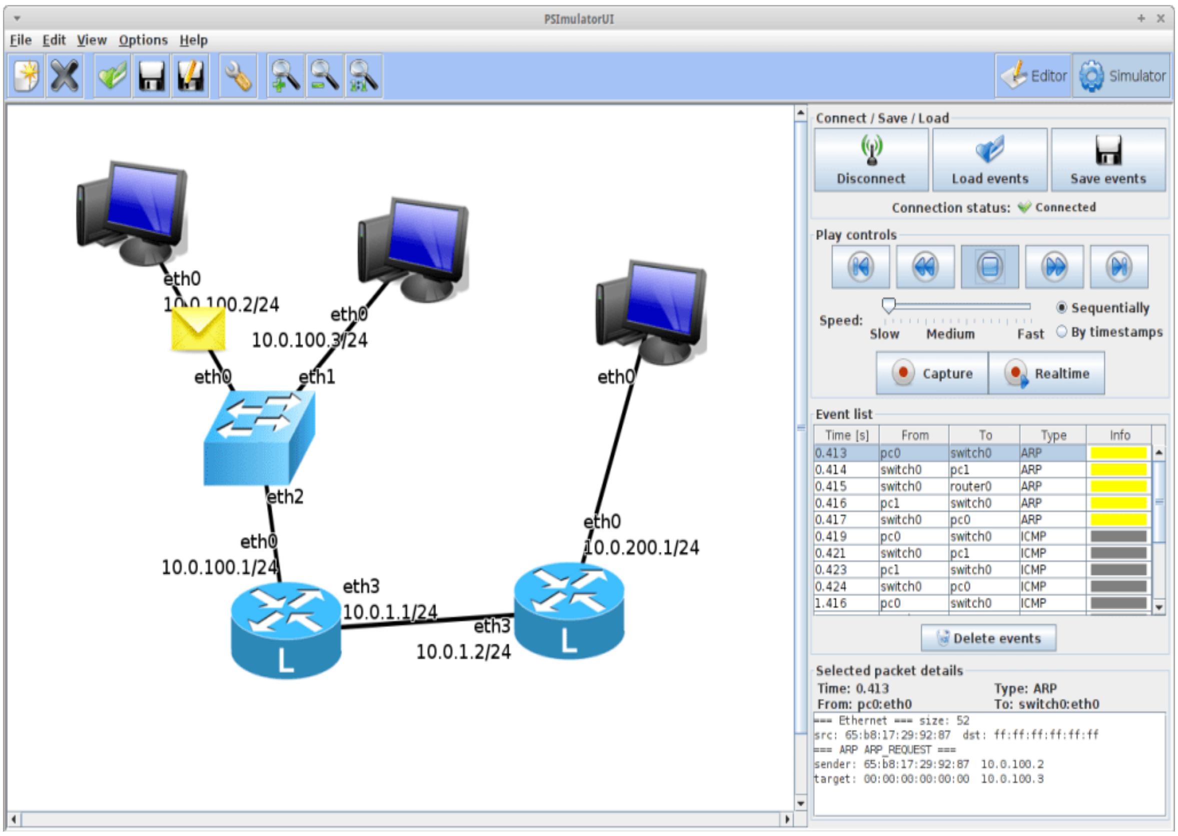


Figure 3.2: Network diagram of basic nodes of Psimulator2 system (Source: GitHub Inc., 2008)

The tools for organising elements on the canvas (section 3.3, Step 2) were configured to represent the following motion and static elements of the system:

- ***The moving elements*** in the system consisted of ARP signals and ICMP signals (vehicles) (section 2.2.3). These were the driving forces of the model and the key constituents for ensuring consistent mobility of passengers from one point to the next. ARP signals were sent to adjacent devices, which, in turn, ensured that vehicles traverse the network via valid routes. ICMP signals were, in essence, the vehicles themselves and were forwarded to the relevant routes depending on the feedback received from the ARP requests. It is worth noting that APR requests were only sent once until all adjacent networks were discovered or in the event that routing tables needed to be recreated (section 2.1).
- ***The static elements*** comprised all routers, switches and nodes. Network routing devices could be considered static elements, as they reacted only to the instant a packet needed to be routed within that specific broadcast or collision domain. Nodes could be considered as both the packet conception point and endpoint, and remained dormant while there was no packet activity (section 2.2.3).

The tools referred to as the entities that represented the system elements (section 3.3, Step 3) are best described by the researcher in Table 3.1 below.

When describing the system components (section 3.3, Step 4), detail was introduced in stages, commencing with components that depicted basic roadway topography and the typical characteristics of such a setting. A dynamic component enabled the animation of graphic items, such as vehicles traversing the roadway. Components such as V2I communication within IoT vehicle communication technologies were further introduced that effectively associated the autonomous vehicles' routing with packet switching, which entails moving small packets of data to a large set of nodes and integrating the system by way of switches and routers.

To ensure that the security of the system was not compromised in any way, because of the fact that human life was dependent on the effectiveness of its functionality, redundancy was considered in the event that a link to a device was lost. V2V accountability was also considered by querying the destination information of vehicles in the vicinity and updating routing information accordingly. In addition, system security was taken into account by encrypting communications between vehicles and devices.

Table 3.1: Tools representing system elements (Source: Researcher)

Tools		Purpose
Platform	JAVA	Required to run all components of the simulator
Design tools	NetBeans IDE 7.4	• Redesign Psimulator components. Transformation to autonomous vehicle simulator by replacing original network entities with those hypothesised for routing autonomous vehicles • Events to be simulated coded (possibly with parameters) according to: 1) Time of occurrence 2) Type
	Psimulator modified version	• Adaptations to connect components for position and timing • Designing a network topology for autonomous vehicles • For identifying defects and changes (in environment/ sensors) • Simulation time tracked, measured in milliseconds • Discrete-event – time hops (as events are instantaneous) clock skips to next event start time (synchronisation of time and event) • Decide on which objects are to transmit a chain of events to other objects for solution, such as, routing devices – from conception node to arrival at each router (similar to a trace-route command) • Instantaneous events that extend over time are modelled as sequences of events. The start time and end time of the event is specified as an interval • Continuous event – automatically configured to change when detecting changes in state of each interface
Configuration tools	Built-in terminal client Alternatively, terminal emulators such as PuTTY can be used	• To configure interfaces on the network element • To telnet and configure IP addresses of each interface • Router configurations • Ping/ICMP requests
Implementation tools	Gradle Save topology XML file	• To compile Java files and run each component individually • Contains all configuration information and meta-data about the created network topology

The tools (section 3.3, Step 5) were determined appropriate to a simulated roadway setting in accordance with the intention of the study and used to introduce animated graphic objects and background. The final procedure (section 3.3, Step 7) meant performing a preliminary simulation to ensure that quantitative data in the model would be gathered from routing table information, which would be translated into mathematical calculations based on distance and factors concerning time.

The input algorithm considered the value of all the system's variables at time t and proceeded to calculate the state of the system at time t+1. From the characteristic values at this state (t+1), additional values would then be calculated at time t+2 and t+3 to t+n. The model was expected to begin conceptualising a numerical representation of the system's evolving status produced by the algorithm. The succession of values for these variables was to be the data that was saved by the model to be observed on a screen to mimic the real-world objectives of the research. Results were to be communicated in the form of tables, graphs, and output data generated by the emulator.

3.5 Conclusion

In Chapter 3, the researcher described the methodology applied in the research and the artefacts used for the successful construction of a prototype computer model with the intention of simulating the real-life process of routing autonomous vehicles. Choice of the model system was substantiated and variables defined to prepare a new algorithm (section 1.8.2). Challenges such as computer security were also addressed.

In Chapter 4, the researcher presents an in-depth description of the simulation and a step-by step data analysis of components. The new algorithm to route autonomous vehicles, which was developed in consequence of the interaction of IoT VANET technology with ITS applications, simulation and theory, is described.

CHAPTER 4: RESEARCH AND DESIGN OUTCOMES

4.1 Introduction

The purpose of this chapter is to provide the researcher with an opportunity to address the outcomes of the research methodology outlined in Chapter 3, and to focus on the design processes that were used, including but not limited to the failure points, internal interlinking of processes and the time factor. A step-by step discussion of the various mechanisms and adapted software to develop the new algorithm is followed as a way of verifying and validating of the process.

A network topology was created exclusively for autonomous vehicles, based on the research conducted that aimed to observe the behaviour of autonomous vehicles as they traverse the simulated network, as explained in Chapter 3.

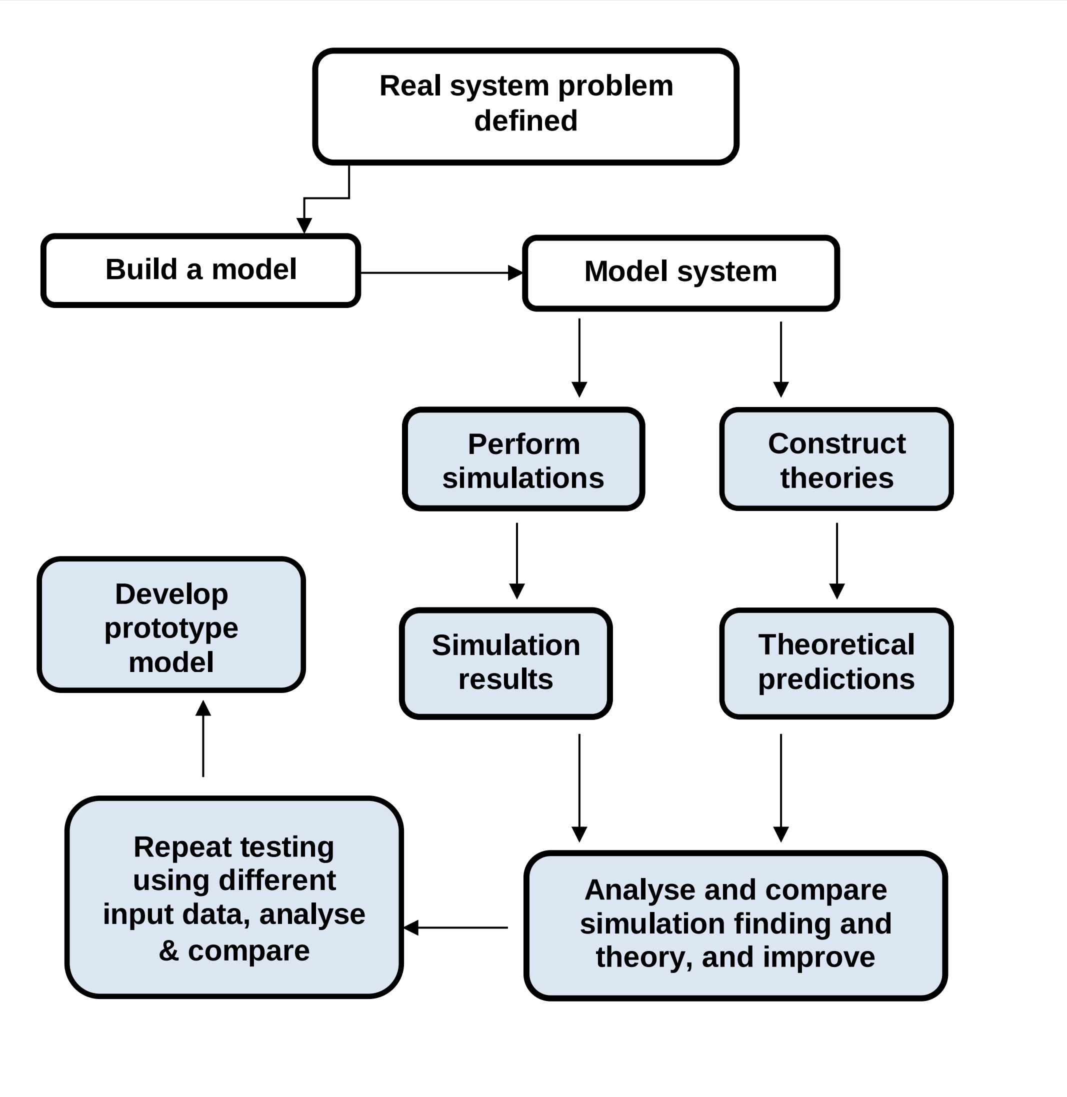


Figure 4.1: Second stage (highlighted) of proposed computer model (Section 1.8.2)

4.2 Building and setting up a network in Psimulator

Prior to performing the simulation that demonstrated the purpose of this research, it is important to describe the functioning of the activities associated with the Psimulator2. Simulation can reveal outcomes of the interactions of unfolding strategic processes of the real world and is a potent method for verifying existing theory (Repenning, 2002). The activities in a simulation model consist of events, actuated at certain points in time that influence relationships between constructs, which affect the state of the system. This allows the model to function as an analytical tool, an aid to experimentation, and for planning and scheduling (Repenning, 2002).

The graphical network, Psimulator2, was employed in the simulation design of this study and the basic requirements of the working system (section 2.7.1.7) were considered sufficient without the need to adopt unnecessary measures and new connections between components. As a mathematical model, Psimulator2 represented the behaviour and the logical and quantitative relationships between network elements.

The topology was designed with the intent to utilise each of the networking components made available by the simulator, with the exception of the Linux routing components. Figure 4.2 illustrates a sample network topology running in a vanilla, or otherwise unmodified, version of Psimulator.

All endpoints in Figure 4.2 differ marginally and are essentially network nodes that constitute notebooks, workstations and desktop PCs. The design called for the section of a network IP ranging between 1 and 7 within a *class A* network configuration, therefore consisting of 8 network bits and 24 host bits. There are also eight broadcast domains. The generic user interface (UI) properties of each node can be seen in Figure 4.3.

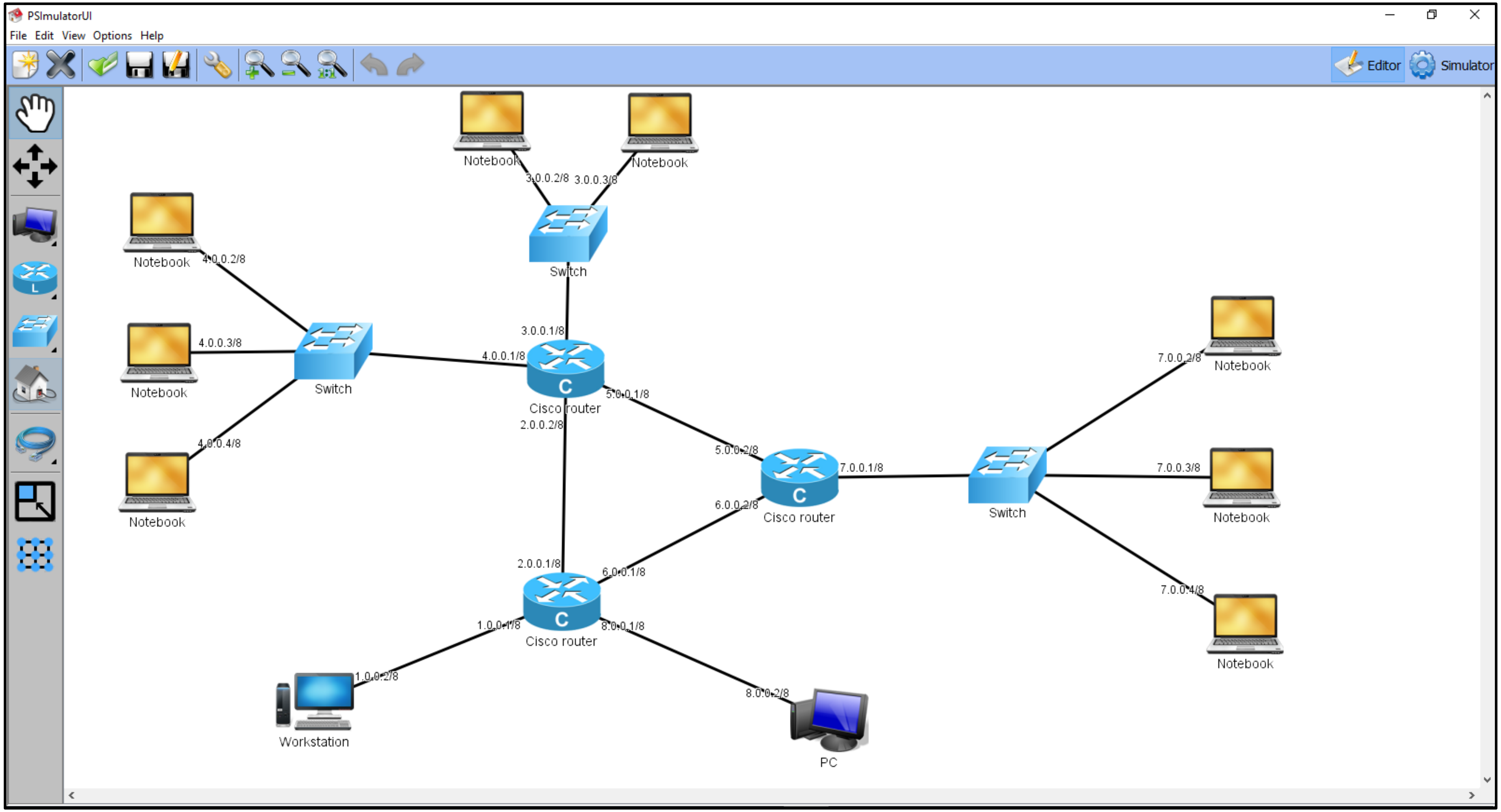


Figure 4.2: Unmodified Psimulator network topology from available simulator routing components

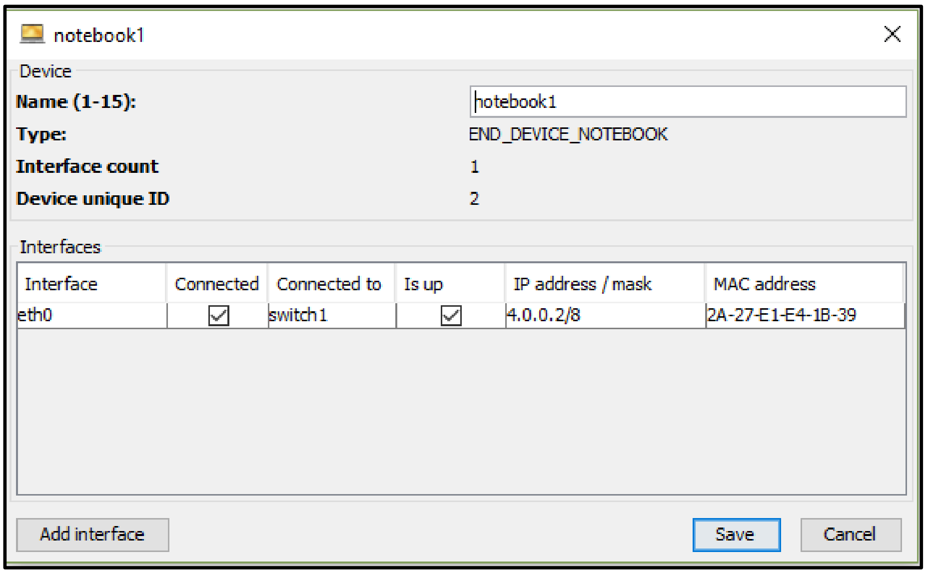


Figure 4.3: Interface from simulation with configured node IP address details depicting UI properties

The MAC addresses and interface port numbers were auto-generated by the simulator, while default component names were also generated, which could be changed at the user's discretion. It was noted that workstations consisted of a minimum of four Ethernet network interfaces. On completion of the network design, the final step was to configure static routes, via the console window, for each router by issuing standard Cisco commands in order to add the necessary static routes to the routing table. Once this process was completed, the user could confirm interface statuses and routing tables on each router, which were viewed by executing basic show commands, as indicated in Figure 4.4. A telnet session to each node was also required for creating a default route to the respective nodes' local networks.

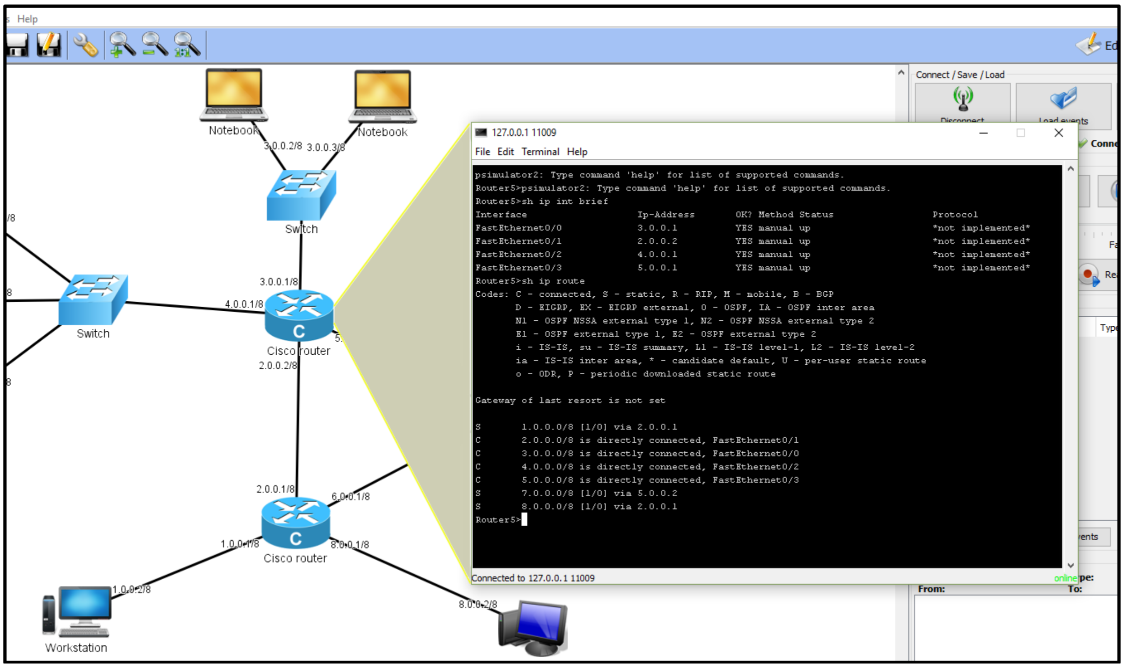


Figure 4.4: Sector of console window (Figure 4.2) of Cisco router from available simulator routing components displaying routing table info and interface status

Once completed, it was important to save the project, allowing it to sequentially build an extendable mark-up language (XML) file containing critical information and metadata pertaining to the design and specific to the saved session. The file was then fed into the emulator, which read the data and created a logical network that imitated the design. In order to view network interactions between nodes, as well as to monitor all routing activity in real time, the back-end emulator had to be started.

For this experiment, the researcher started the emulator via command prompt and used Gradle to compile the code; the boot process could be observed in the command prompt window, as illustrated in Figure 4.5.

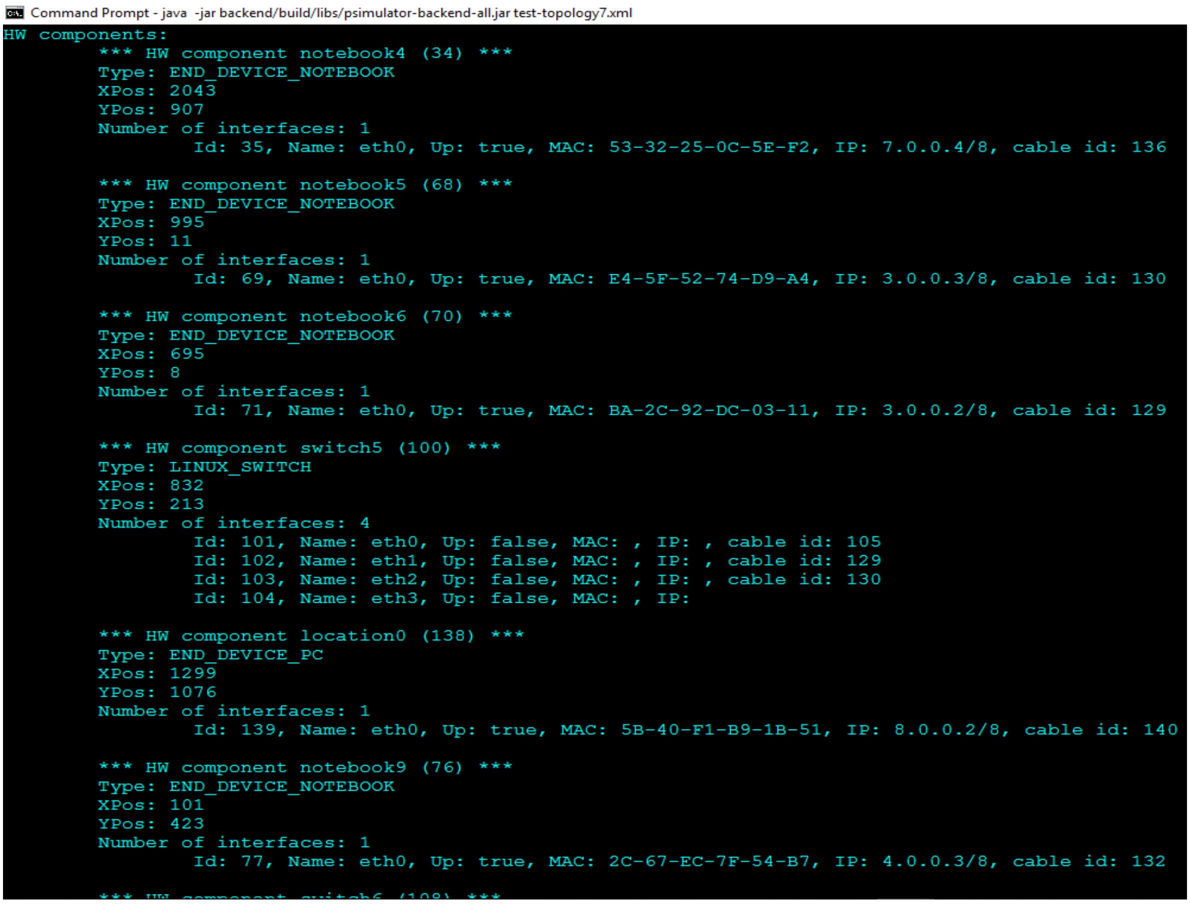


Figure 4.5: Start-up screen of Psimulator network emulator for commencing simulation experiment

The final step was to establish a connection between the simulator front-end and the emulator, which was done by a simple click of a button on the UI, at which stage the connection testing could commence by sending a series of ICMP requests. Once the user started recording the network activity and played back the animated recorded session, it was immediately observed that ARP requests were sent to each device along the path bound for the target node, followed by the payload (ICMP packet).

For ease of reference, the developers color-coded the different packet types, using yellow envelopes to denote ARP packets and grey for the payload, as shown in Figure 4.6.

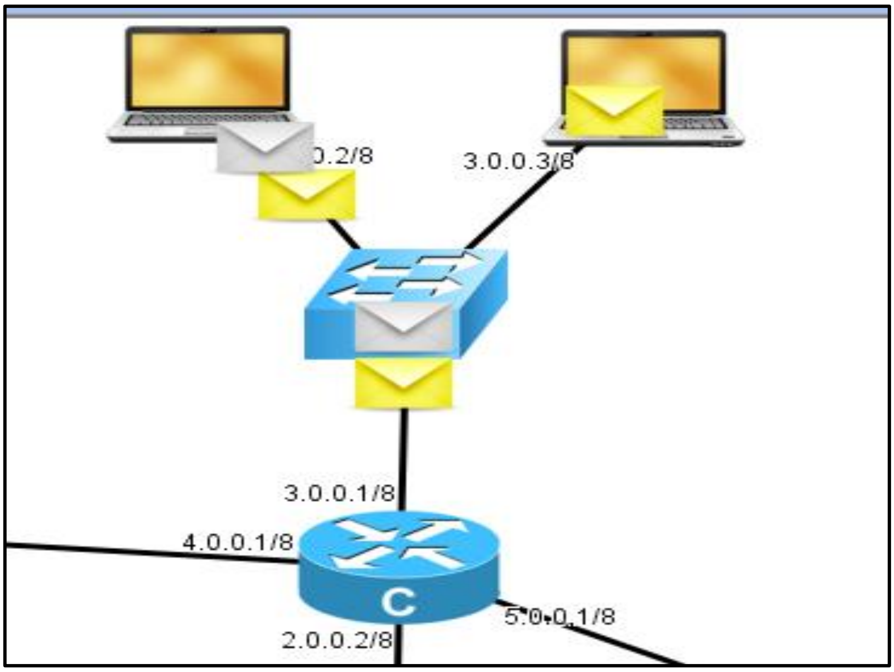


Figure 4.6: Accessible animated envelopes for simulation indicating colour-coded data transmitted

4.3 Perform simulation

Psimulator2 required manual configuration since it sustains only static routing (Baltatu et al., 2000), and static simulation is more rigid and structural in respect to the behavioural characteristics of the dynamic model.

By means of the GUI, a virtual simulated network was created with a similar configuration, as discussed above (section 4.2), in order to carry out the experiment for this research. The network topology, configuration and XML file used were all identical, with the exception of the naming of each component that needed to represent typical roadway topography. The experiment portrayed the concept of routing autonomous vehicles in a real-life scenario if we were to apply the exact same principles to this scenario as are applied to packet switched networks; thus, instead of moving an envelope across the network, the researcher customised the code to display an autonomous vehicle. In addition, all other routing components such as routers, cables and switches, were modified to exemplify wireless beacons that served the same function as their predecessors in their original state, prior to being modified, and which depicted the vital constituents of the system and relationship with IoT connectivity.

Communication was established via the emulator or back-end component to mimic the real-life situation, allowing the two independent components to work in synchronisation. Network routes were then manually entered into a router's routing table, instructing the next hop router IP address to forward traffic to the following destination (Osterloh, 2001) (section 2.2.3). Figure 4.7 illustrates a legend of components, adapted and introduced to the simulator, that were required to illustrate the experiment of routing an autonomous vehicle from its point of origin to its destination, expounded on in points 4.3.1 to 4.3.6 below.

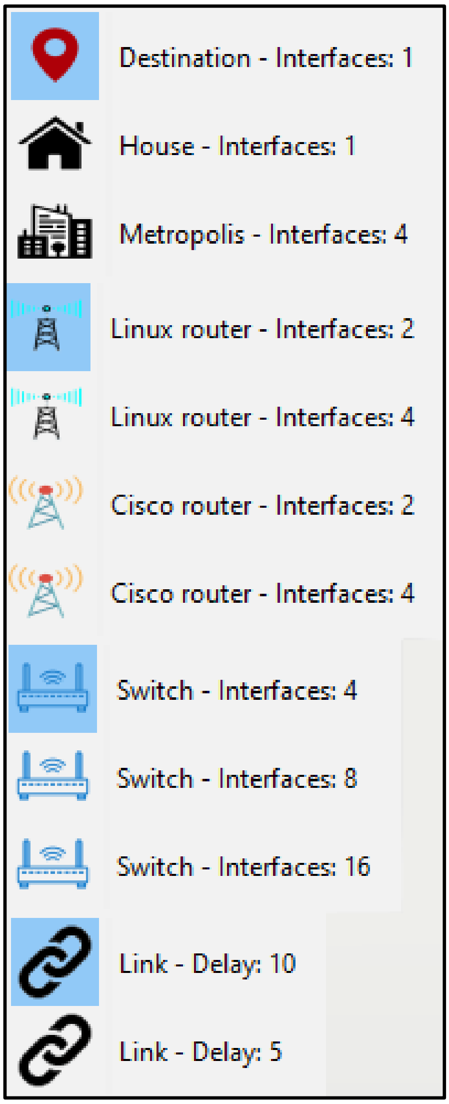


Figure 4.7: Psimulator customised legend for illustrating the experiment in this research

Expanding and customising the model was possible by modifying the code, but additional control logic was unnecessary at this stage of the experiment.

4.3.1 Destination/location component

The location component was analogous to a standard PC in the original simulator and was, essentially, an endpoint or point-of-arrival in the context of the modified simulator, consisting of a single interface by default, although more location components could be added at the user's discretion. The setting up of a location component required that the user follow the conventional Psimulator steps and procedures used to set up a PC component. These endpoints/nodes could be viewed as Network Interface Cards (NICs), where an IP address and accompanying subnet were assigned. The researcher excluded Domain Naming Server (DNS) details in the configuration process, since this was outside the scope of this research.

4.3.2 House component

Similar to location, the house component consisted of a single interface, by default, with the option of adding more interfaces at any time. The house component was analogous to a notebook in the original Psimulator. The researcher only used the necessary amount of house components needed for the demonstration.

4.3.3 Metropolis component

The metropolis component consisted of multiple interfaces by default. The researcher refashioned this component to represent a conurbation or hypothetical hub, expecting higher traffic flow or throughput. The primary function of the metropolis component, therefore, served as both an endpoint and route, akin to vehicles either bound for the component or just passing through it. It was noted that the location and house components could well consist of several interfaces, which come as a standard feature in Psimulator. Figure 4.8 illustrates the properties UI of a metropolis, where name and interface details are configured and observed in the simulation.

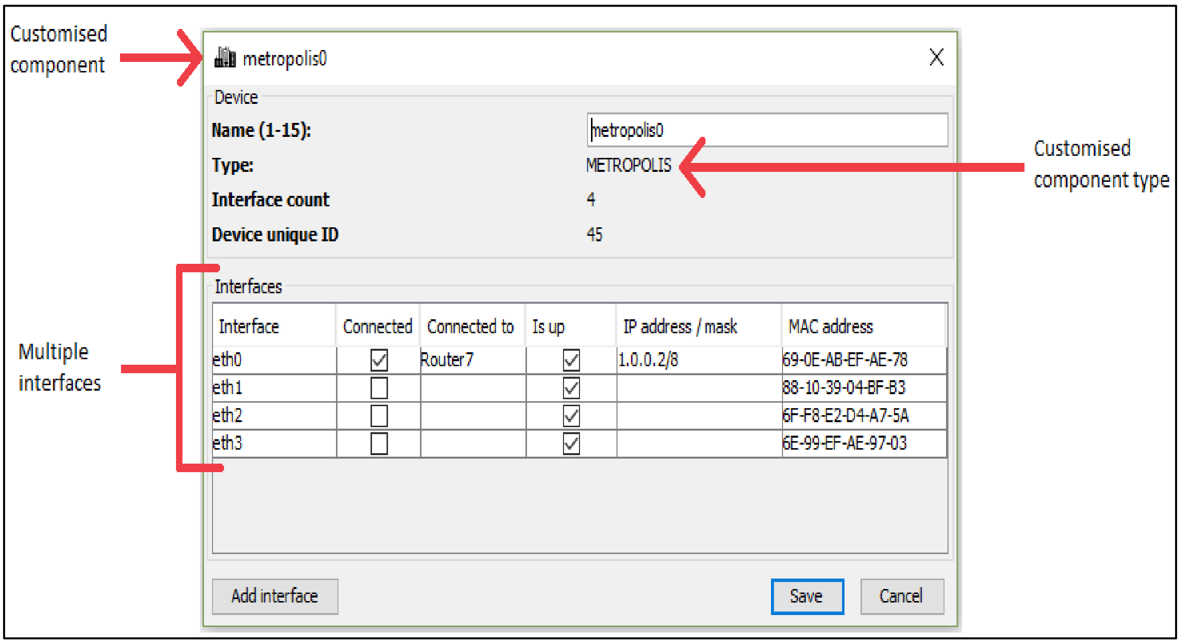


Figure 4.8: Properties configuration user interface from the experimental simulation

4.3.4 Router component

The router component was analogous to a standard Cisco or Linux router, which the researcher transformed into a wireless homing beacon, leaving most of its internal processes unchanged, as the standard function of a router was imperative to emphasising the true nature of packet switching in a data network.

4.3.5 Switch component

The switch component was used to expand a local network, as was the routine behaviour of a network switch. These components required the least amount of configuration as the interfaces are automatically detected. For the purposed of this research, the switch component was transformed into a wireless homing beacon, similar to that of the router component.

4.3.6 Link component

Initially a cable component, the link component, was reengineered to act as a wireless interlinking mechanism required to link network components together via Ethernet interfaces, and was converted from a solid to a broken line. The link delay, another standard feature in Psimulator, could also be configured. The purpose of the delay feature was to analyse events when running in simulation mode.

4.3.7 Observation

The simulation was observed to determine whether it mimicked the predicted characteristics of the real world in order to apply suitable mathematical representations as a consequence with the intention of conveying the results effectively. The computational demonstration needed to represent accurately the essential theoretical reasoning of the simulation that links constructs of the virtual and real world together (Sastry, 1997). This was done to determine whether it was possible for the artefacts to be demonstrated in a physical working system (Gregor & Jones, 2007) (section 2.7.1.3) and reveal the effect that the technology would have on humans, as recognised in design theory, stated in Walls, Widemeyer and El Sawy's (1992) extended ISDT (section 2.7.1.6).

The network setup was created to be entirely static, requiring the researcher to instruct each device explicitly as to which route and interface to use (section 4.2). Configuration is often required on standard network devices, irrespective of the routing protocol used. The exception is the switches that automatically query the end-device once plugged into an interface. Based on this hypothesis, every house component added to the network topology was to remain unrecognised or invisible until a default route was configured.

4.3.7.1 Broadcast domain

Based on the data collected (section 4.3), it can be seen that the broadcast domains were only discovered once routers had identified the requested network in their individual routing tables. It was also observed that the target MAC address would remain unset until the required route was identified by a router.

4.3.7.2 Timestamp

It was further observed that the start-up phase of packet routing took longer because of the ARP requests that were distributed through the network; thereafter, time increments were reduced once the route was 'learned' by the routers along the lines of ML.

4.3.7.3 Security

The manual characteristic of static routing limits harmful wireless transmissions from infiltrating the model system (section 5.3.3), while the security afforded by open source software is valued by organisations (Guseva, 2009) (section 3.3).

To secure the system, routing information was structured to be updated constantly via the routing tables, and any link that was lost by a device at any time was to be made redundant (section 3.4). This meant that, because of the continuous updating of routing information, an instant alternative route would be acquired.

4.4 Simulation results

The simulation displayed packets in a virtual context, as the emulator replicated the real-life situation via the simulator's back-end. Information was logged at the start-up for recording purposes, with real-time feedback when operating the routing animation feature. A visual display of routing information was recorded in the form of next hop addresses, time in milliseconds and relevant network material. It was observed that the number of hops (Nh) to the destination decreased proportionately on arrival of packets at a switch or router. This indicated that the number of hops (Nh) was indirectly proportional to distance (s) Nh = 1/s computed s (hop count) (section 3.3, step 1).

A larger datagram was noted for the transmitted (Tx) packets, since they must locate the routing path through the network. The receiving (Rx) packets were buffered and their datagram was then unpacked to determine whether they had arrived at the correct destination. When the destination that the packets arrived at was incorrect, the receiving packets were not processed further but rather re-packaged into a new (Tx) packet for further routing through the network. A numerical representation of the developing status of the system was produced and the information was observed on the screen (section 3.4).

4.4.1 Routing outcomes of experiment and events

The network topology selected for this experiment, as shown below in Figure 4.9, was a replica of the layout illustrated in Figure 4.2.

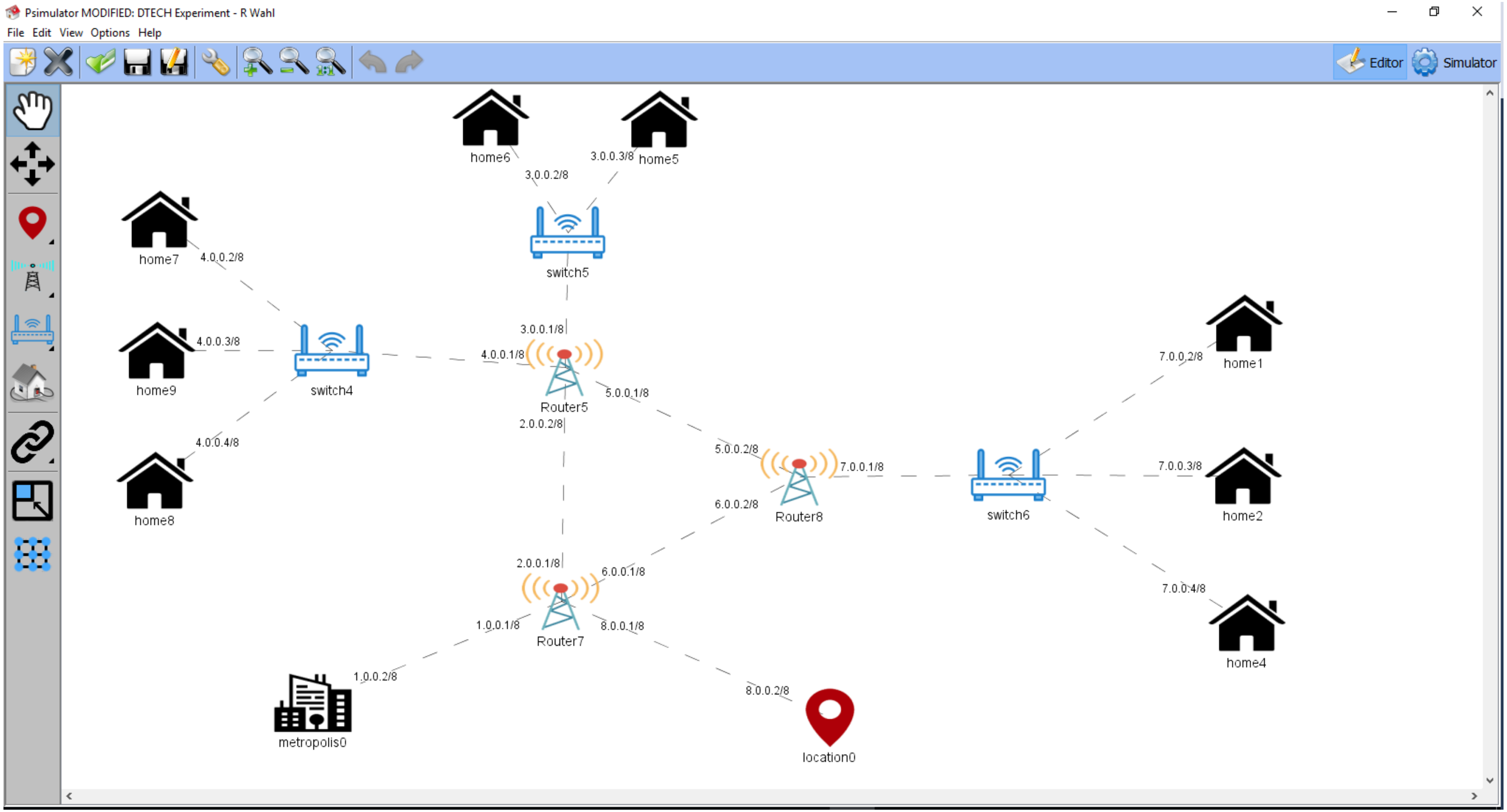


Figure 4.9: Experiment network topology portrayed during simulation

In order to demonstrate effective routing dynamics, the researcher decided to route from opposite sides of the canvas, i.e. route starting from network 4.0.0.0/8 and ending at network 7.0.0.0/8. It is important to note that the simulator was further modified to omit response packets from being sent back to the requesting node, which is the typical behaviour of a ping request. The reason for this was to focus on exclusively routing vehicles to their destination without a reverse effect, since the aim of the research was to explicitly route a vehicle from a place of origin to its destination, and not back again in the same session.

House address 4.0.0.4 was selected as the place of origin for the experiment, and address 7.0.0.2 as the destination. We can see this in action in Figure 4.10 where an animated autonomous car can be seen in transit through the network en route to its destination.

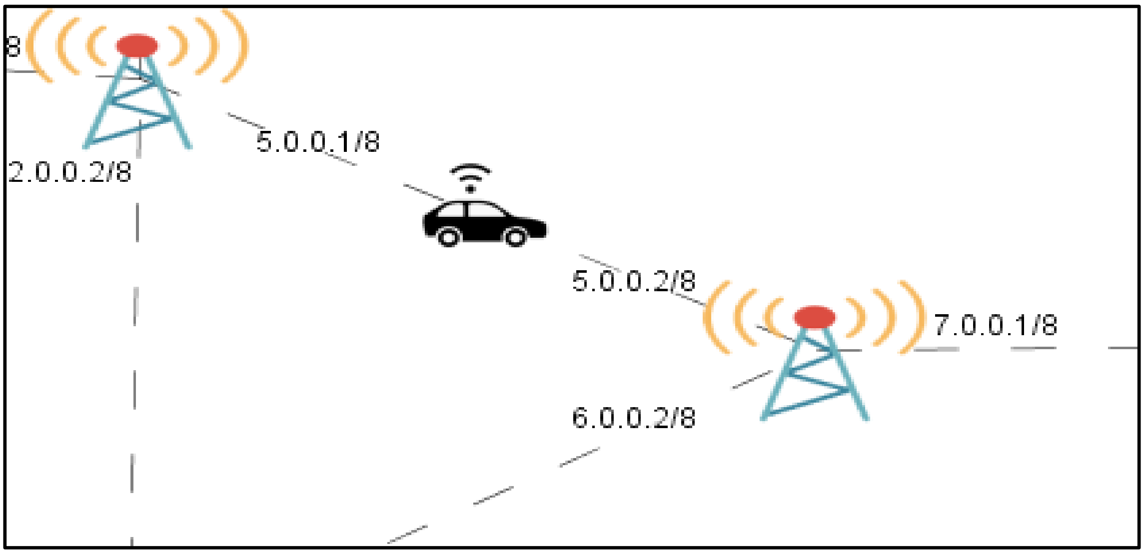


Figure 4.10: Routing simulation of autonomous vehicle

4.4.1.1 Recording events

The back-end process logged each event via the console as it occurred. The following table shows a snippet of the events that transpired during the initialisation of the routing process.

The data captured in Table 4.1 highlight the routing events that transpired between house and router components. As mentioned above, a 'send response' event was also part of this cycle of events, which was supressed in the modified version by the researcher (section 4.3.1). A complete log of events can be viewed in Appendix A.

A more comprehensive log file was created by the emulator during the recording/capturing process when the researcher saved the session, and which could be used to reload the session again. A snippet of this log can be seen in Table 4.2 and focuses on comprehensive communication between devices, with emphasis on ARP and ICMP packet interaction between routing devices. The complete log can be viewed in Appendix B.

Table 4.1: Console logs of captured events

Route Info	Packet Info	Details
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Received IP packet destined to be mine.

4.4.1.2 Component events

With reference to Table 4.2, it can be seen that ARP requests are the first to be transmitted, as one would expect. When neighbouring network devices respond to these requests by announcing that they contain the required route to the destination network, the payload (ICMP packet) is then transmitted and the cycle continues until the payload eventually arrives at the intended destination house component. Note that ARP requests only occur at the start-up of a network. The payload will therefore be routed continuously to the destination without further ARP requests sent for every echo request within the same session.

4.4.1.3 Event sequence

The component identifiers give the observer a sense of how the sequence of events is being processed, the time taken per event measured in milliseconds and its accompanying status. Some of this data appears in the detailed text of Table 4.2. Furthermore, the details field encapsulates detailed information and can be broken down into the following constituents:

- **Type of transmission:** In this case Ethernet is the type of transmission used, which touches both Layer 1, the physical and Layer 2, the data link layer of the OSI model
- **Size:** The logical size of the packet being transmitted measured in bytes
- **Source physical address:** The Layer 2 address (MAC address) of the transmitting device
- **Destination physical address:** The MAC address of the next hop device. It can be seen here that packets are allocated the default MAC address: ff:ff:ff:ff:ff:ff when first discovered on the network. Only once a router identifies the destination network in its routing table does the MAC address of that router get populated
- **Time-to-live (TTL):** TTL is a value that tells a network router whether or not the packet has been in the network too long and, if it has been too long on the network it should be discarded, which is typically the case in the event of a loop
- **Packet type:** Packet type can be one to two types: ARP or ICMP
- **Sender:** Attributed to both the MAC and the IP address of the interface of the source device or the device sending the packet
- **Target:** Attributed to both the MAC and the IP address of the interface of the neighbour/next-hop device where the packet originated from

4.4.1.4 Component identifiers

The component identifiers (Cable ID, Destination ID and Source ID) are auto-generated as they are incorporated into the canvas and added to the simulator log. In addition, these identifiers are key elements in the forging of a visual reference in the form of animation within the simulation canvas, thus giving the observer a high-level view on the vehicle routing operation.

4.4.1.5 Timestamp

As mentioned above, the timestamp is measured in milliseconds and is an innate feature of the Psimulator emulator (sections 4.4.1.3 & 3.3, Step 1), which gives the observer an added perspective on the efficiency of routing packets, based on specific packet types and distance between devices. The timestamp value, as seen in Table 4.2, is increased sequentially per event in the session. This implies that time measured in milliseconds increases in chronological order for each event in the routing episode.

A numerical expression of an event extracted from Table 4.2, for example, can refer to each of the component IDs, such as cable, destination and source ID, which are increased chronologically with respect to the event. A sizable time gap can be seen between cable ID:134 to source ID:28 at 712 ms, and the subsequent event with cable ID:133 to source ID:95 at 1 660 ms. The latter is a new request originating from the requesting component. Thus, the time gap is indicative of a completely new vehicle being sent to its destination.

Table 4.2: Extract of log created by the emulator used for replaying the simulator

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
132	76	=== Ethernet === size: 52 src: f9:c8:d5:68:47:e5 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: f9:c8:d5:68:47:e5 4.0.0.4 target: 00:00:00:00:00:00 4.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	95	656
106	95	=== Ethernet === size: 52 src: 44:0e:7c:1c:fa:6d dst: f9:c8:d5:68:47:e5 === ARP ARP_REPLY === sender: 44:0e:7c:1c:fa:6d 4.0.0.1 target: f9:c8:d5:68:47:e5 4.0.0.4	SUCCESSFULLY_ TRANSMITTED	ARP	80	658
133	74	=== Ethernet === size: 52 src: 44:0e:7c:1c:fa:6d dst: f9:c8:d5:68:47:e5 === ARP ARP_REPLY === sender: 44:0e:7c:1c:fa:6d 4.0.0.1 target: f9:c8:d5:68:47:e5 4.0.0.4	SUCCESSFULLY_ TRANSMITTED	ARP	95	659
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	74	666
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	95	668
125	118	=== Ethernet === size: 52 src: 53:f4:72:6a:4d:9e dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 53:f4:72:6a:4d:9e 5.0.0.1 target: 00:00:00:00:00:00 5.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	80	678
125	80	=== Ethernet === size: 52 src: fd:9b:74:b9:5f:d6 dst: 53:f4:72:6a:4d:9e === ARP ARP_REPLY === sender: fd:9b:74:b9:5f:d6 5.0.0.2 target: 53:f4:72:6a:4d:9e 5.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	118	692

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	698
698	108	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	118	702
134	28	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	108	704
135	30	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	108	705
136	34	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	108	705
134	108	=== Ethernet === size: 52 src: 49:07:cb:76:58:bb dst: a3:db:32:49:14:19 === ARP ARP_REPLY === sender: 49:07:cb:76:58:bb 7.0.0.2 target: a3:db:32:49:14:19 7.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	28	707
128	118	=== Ethernet === size: 52 src: 49:07:cb:76:58:bb dst: a3:db:32:49:14:19 === ARP ARP_REPLY === sender: 49:07:cb:76:58:bb 7.0.0.2 target: a3:db:32:49:14:19 7.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	108	708
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	118	711
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	108	712
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	74	1660
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	95	1663

Figure 4.11 illustrates the historic number of events that occurred from the source to the destination during the simulation, which reflects six echo requests that were sent to the destination, providing the reader with further insights into the routing process of an autonomous vehicle and the relationship of the process to packet switching.

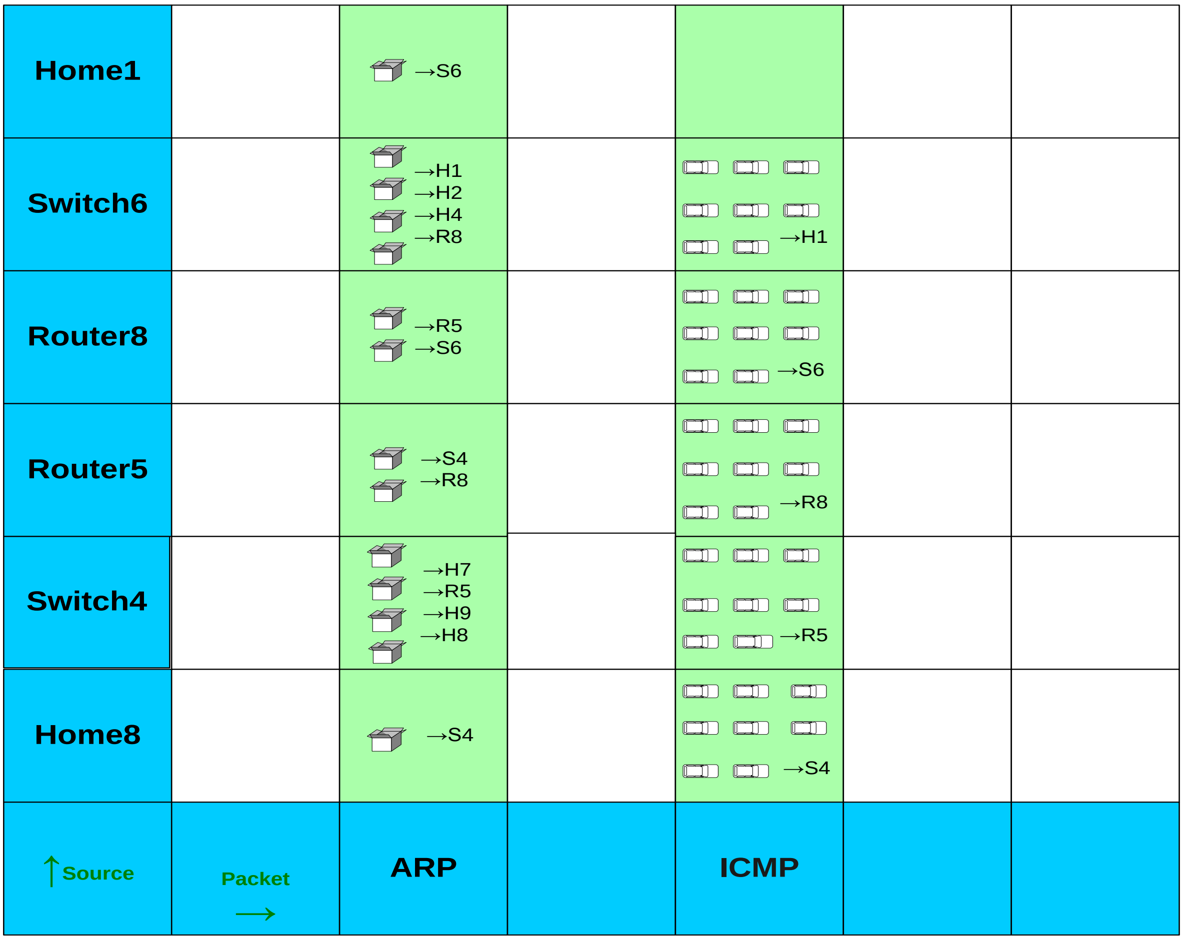


Figure 4.11: Graphical log of events in simulated routing of autonomous vehicles

In line with work on computer simulations conducted by Gulyås and Kampis (2015), repeat testing or replication is an underlying requirement for conducting an experiment and is critical for yielding effective results. A second test was therefore carried out by the researcher using different input data, as illustrated in Figure 4.12 below, where the source address *Metropolis0* and the destination address *Location0* were selected for a second case study to be run.

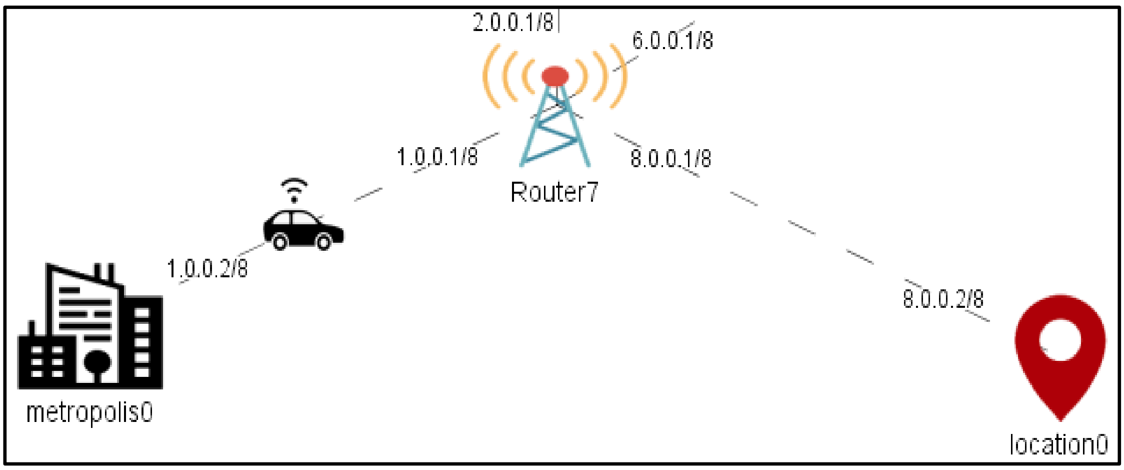


Figure 4.12: Replication test sampling

4.5.1 Failed case study

The researcher enacted a failed-case study to obtain a perspective on how the simulator would handle failures of the status quo. A rudimentary approach to impose a failure effortlessly would have been simply to send a vehicle to a non-existing destination; however, for added appeal, the researcher decided to route a vehicle to a fictitious destination via an *existing* network. The reader will recall that routers are concerned with the network portion of the address and not the destination itself; therefore, in theory, an ICMP request would have made the trip right up to the network it was bound for, before discovering that the physical destination did not exist.

4.5.2 Replication results

A segment of the output data can be viewed in Table 4.3. The data reflects ICMP requests only and the complete log can be found in Appendix C.

Table 4.3: Output data for case study 2 of the experiment

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO C3id: 1025 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	45	1139
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	1142
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 3 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	45	2135
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO C6id: 1025 seq: 3 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	2138
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 4 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	45	3134
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 4 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	3136
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 5 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	45	4136
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO C10id: 1025 seq: 5 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	4139

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 6 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	45	5135
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO C12id: 1025 seq: 6 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	5137

4.5.3 Failed case observation

The output data confirmed accuracy of the simulation by illustrating a vehicle routing from *location0* to fictional address *3.0.0.6* could not complete the trip. On further observation, it could be seen that a request was sent to the router hosting the destination network on three occasions before returning a failed response. Thus, in compliance with current routing standards (IEEE 802.11p) (IEEE, 2018), the respective journeys of the first two vehicles were suppressed at the final network portion, and any subsequent vehicle was routed back to its originating address. A snippet of the output data can be viewed in Table 4.4.

Table 4.4: Failed case study depicting unreachable host/destination

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
105	100	=== Ethernet === size: 52 src: 9f:90:89:6f:5c:e6 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 9f:90:89:6f:5c:e6 3.0.0.1 target: 00:00:00:00:00:00 3.0.0.6	SUCCESSFULLY_ TRANSMITTED	ARP	80	3400
130	68	=== Ethernet === size: 52 src: 9f:90:89:6f:5c:e6 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 9f:90:89:6f:5c:e6 3.0.0.1 target: 00:00:00:00:00:00 3.0.0.6	SUCCESSFULLY_ TRANSMITTED	ARP	100	3401
129	70	=== Ethernet === size: 52 src: 9f:90:89:6f:5c:e6 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 9f:90:89:6f:5c:e6 3.0.0.1 target: 00:00:00:00:00:00 3.0.0.6	SUCCESSFULLY_ TRANSMITTED	ARP	100	3402
-1	-1	=== IP === src: 8.0.0.2 dst: 3.0.0.6 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 1 payloadSize: 56	LOST_IN_DEVICE	ICMP	80	3413
124	113	=== Ethernet === size: 108 src: b2:4b:6f:1c:99:57 dst: d4:5d:04:04:cb:d3 === IP === src: 2.0.0.2 dst: 8.0.0.2 ttl: 255 size: 84 === ICMP === type: UNDELIVERED code: HOST_UNREACHABLE id: 1025 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	3417
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 2.0.0.2 dst: 8.0.0.2 ttl: 254 size: 84 === ICMP === type: UNDELIVERED code: HOST_UNREACHABLE id: 1025 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	3421

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
140	113	=== Ethernet === size: 108 src: 5b:40:f1:b9:1b:51 dst: 39:18:99:3c:2b:8f === IP === src: 8.0.0.2 dst: 3.0.0.6 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 5 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	138	4395
124	80	=== Ethernet === size: 108 src: d4:5d:04:04:cb:d3 dst: b2:4b:6f:1c:99:57 === IP === src: 8.0.0.2 dst: 3.0.0.6 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 5 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	4397
105	100	=== Ethernet === size: 52 src: 9f:90:89:6f:5c:e6 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 9f:90:89:6f:5c:e6 3.0.0.1 target: 00:00:00:00:00:00 3.0.0.6	SUCCESSFULLY_ TRANSMITTED	ARP	80	4399
129	70	=== Ethernet === size: 52 src: 9f:90:89:6f:5c:e6 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 9f:90:89:6f:5c:e6 3.0.0.1 target: 00:00:00:00:00:00 3.0.0.6	SUCCESSFULLY_ TRANSMITTED	ARP	100	4400
130	68	=== Ethernet === size: 52 src: 9f:90:89:6f:5c:e6 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 9f:90:89:6f:5c:e6 3.0.0.1 target: 00:00:00:00:00:00 3.0.0.6	SUCCESSFULLY_ TRANSMITTED	ARP	100	4401
-1	-1	=== IP === src: 8.0.0.2 dst: 3.0.0.6 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 2 payloadSize: 56	LOST_IN_DEVICE	ICMP	80	4401
124	113	=== Ethernet === size: 108 src: b2:4b:6f:1c:99:57 dst: d4:5d:04:04:cb:d3 === IP === src: 2.0.0.2 dst: 8.0.0.2 ttl: 255 size: 84 === ICMP === type: UNDELIVERED code: HOST_UNREACHABLE id: 1025 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	4405
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:515 === IP === src: 2.0.0.2 dst: 8.0.0.2 ttl: 254 size: 84 === ICMP === type: UNDELIVERED code: HOST_UNREACHABLE id: 1025 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	113	4408

4.6 Construct theories

The use of simulation as method for developing theory has increased significantly, as documented in the literature, particularly when insight needs to be gained with respect to challenging empirical data limitations (Repenning, 2002; Zott, 2003; Davis, Eisenhardt & Bingham, 2007). Considering this, based on theory development initiatives by scholars, theory is defined as "consisting of constructs linked together by propositions that have an underlying coherent logic and related assumptions" (Davis, Eisenhardt & Bingham, 2007:485).

Although the static characteristics of a system in a static model are associated with managed data, structural shape and architecture, the main value of the simulation was in the use of creative experimentation to produce theory (Davis, Eisenhardt & Bingham, 2007) (section 2.3.1), and this process began as the underlying relationships between constructs were seen to influence the behaviour of the system. In order to enable the simulation to answer the research questions, assumptions and theory, the computational representation was developed to include the following:

- Putting theoretical constructs into operation
- Building an algorithm that mimics the theoretical logic
- Stipulating the assumptions that bind theory and results – namely predictions

4.7 Theoretical predictions

It was expected that the simulation would conform to expectation in demonstrating the effective routing of autonomous vehicles based on the packet switching concept as, according to Norton and Suppe (2001:93), "simulations can warrant belief because they are literally experiments". The small data ecosystem required less modelling effort and, as predicted, was more viable economically, since infrastructure costs increase with network size. This is consistent with design theory in that the larger the selection of components, the greater the array of design options and degree of adaptability needed (Gregor & Jones, 2007) (section 2.7.1.5). It was also assumed that an event is an occurrence at a point in time (Hanseth & Lyytinen, 2010) that changes the state of a system.

Additional data from other sources were expected to increase the capacity for more accurate modelling, which is affirmed by systems theory in that the behaviour of each element contributes to the purpose of the whole, or is influenced by another element(s) within the system (Ackoff, 1981) (section 2.7.2.3). It was also anticipated that the model would show that the database was likely to continue to grow significantly as the closed systems rationale of cybernetics allowed diverse disciplines to communicate with each other in the ever-emergent milieu of the Internet of Things (IoT). Emergent properties, as articulated by Bedau (2011), are needed to explain how the whole depends on its parts and how the whole is independent of its parts. The emergent properties of the initial basic simulated system would ultimately require numerous complex mechanisms consisting of independent interacting parts and, as posited by Bedau (2011), the only means of understanding the behaviour of such a system would be to run a simulation and observe the outcome. Further to this, and in accordance with open systems theory of cybernetics, the model was expected to show how it adapted to situations that develop and how it reacted to threats (Ackoff, 1981) (section 2.7.2.3).

It was assumed that an algorithm developed as code would perform as expected, since, in order to verify the conclusion, a comparison of the computed output is required, along with clarified analysis (Parker, 2015). In the final confirmation, negative testing the model and comparing the implemented results to those of the preliminary run, outcomes were expected to coincide.

4.8 Analysis

A static model characterises a system independent of time and represents the static software constituents, such as classes, objects and interfaces, and their relationship with one another (Wichmann et al., 1995) and an indication of their employment in the demesne of IoT. Static analysis provides reasons for program behaviour and is undertaken prior to running the simulation. Analysis in this research concentrated on the modified code.

In order to focus on the objectives of the study, features were kept to a minimum, with the basic macro level model illustrating a roadway infrastructure and typical environment. Detail was provided only to those entities necessary to exhibit the purpose of the research and to demonstrate that elements could be added in stages. Commands were simple, allowing for clarity and accuracy in examining ARP requests and ICMP messages (section 2.1). In principle, ICMP signals were the vehicles that were forwarded to the specific routes based on ARP information (section 3.4). By modifying the code, the model could be expanded and adapted; but extra control logic was deemed superfluous during the initial phase of the simulation (section 3.3, step 6).

Static routes, which were the routes to destination hosts or networks and which were manually entered into the router's route table, outlined the IP address of the next hop router and local interface to use when forwarding traffic to a certain destination (Osterloh, 2001). This meant that, following manual configuration, information was received in the routing system via all connectivity used in the experiment. This also indicated that the real-world connectivity incorporated in IoT would effectually include elements such as sensory devices and LiDAR that would facilitate the measurement of distances through photo-detection (section 2.3.2.1).

The safety aspect of simulating the routing of an autonomous vehicle was evident in the data gathered from the broadcast domain, which showed that routers react only to networks in their respective routing tables, and target only the MAC address identified by a router. The ability of routers to 'learn' routes, to memorise and become adept at effectively managing constantly updated network information, substantiates the concept of ML to achieve AI (McCarthy & Feigenbaum, 1990) (section 2.3.2).

Following initialisation, the simulation mirrored the anticipated characteristics and behaviour of the real world, evidenced in the computation and endorsing the theoretical rationale that consolidates the virtual realm of simulation and the real world (Sastry, 1997) (section 4.3.7); and, in order to confirm the results, analysis was also conducted on the repeat test.

In replicating the model, the results and interpretation of the original simulation were compared to those of the enactment. Repeat testing entailed using different data input via the existing network, from an alternative source to a fabricated destination. The response to the request sent on three occasions to the router hosting the destination network was to disallow, each time, the first two vehicles from continuing their journey on reaching the final network section. Since subsequent vehicles were routed back to the address of origin, which is probably not an ideal situation, it is recommended that a virtual request be sent to the destination address first, awaiting a reply before mobilising the vehicle towards its destination.

Failure of the first vehicles to reach their fictitious destinations was in accordance with routing standards (section 4.5.3) and because of the static nature of the route when the router or interface defined became unrecognisable. Furthermore, in harmony with open systems theory of cybernetics, the model showed how it could adapt, and act accordingly, to (adverse) situations that develop (Ackoff, 1981) (section 2.7.2.3). This is parallel with ML's ability to learn and respond accordingly to an incorrect instruction and also verifies independence of the system to function effectively.

Routing vehicles in a different configuration to the initial experiment using negative testing demonstrated the overall effectiveness of the simulated second case study and showed how real-life routing of an autonomous vehicle to an inaccessible host can be dealt with. In the final analysis, the simulation model chosen was an appropriate instrument to demonstrate the routing of real-life autonomous vehicles within the concept of packet switching. Comparing computed output with analytic solutions is part of the solution towards the verification and development of a prototype.

4.9 Develop prototype

The initial basic simulation was run and the outcome compared to a replication test, choosing a different route to demonstrate the routing of an autonomous vehicle from a source to a destination, in a manner analogous to the packet switching theory. An animated autonomous vehicle was observed in the simulated prototype to traverse the network successfully en route to its destination.

4.10 Validation and verification

Validation deals with physics and choosing the correct equation in the process of determining whether the chosen model effectively represents the real-world system for the purpose of the simulation, and involves comparing model output with observable data (Parker, 2015; Oberkampf & Roy, 2010; Roy, 2005). According to Parker (2015), a simulation can provide more reliable knowledge of a system than an experiment can when relevant background knowledge is accurate. Validation confirms whether there is no major difference between the model and the real system and, therefore, the model reflects reality (Barberousse, Franceschelli & Imbert, 2009).

With reference to this research, there is no historical data available; therefore, the outcomes of this research cannot be validated against an existing system; however, validation processes that are applicable to this research involved:

- Replicating the experiment by varying input data values through negative testing of alternative routes so as to analyse the results and propose changes to the experiment
- Adding new features such as removing response packets from the simulation ensured a structured understanding of theory
- A structured walk-through technique showing how the arrival of a packet at a router is an event that alters the state of the port buffer in the router, thereby changing the state of a system and validating an event
- Observing behaviour of the simulated representation to ensure it fulfilled the real-world purpose of the research, and establishing whether simulation results matched empirical evidence (Campbell & Stanley, 1966)

Verification entails solving correct mathematical equations and is the process of ensuring that the model behaves as intended. Verification can be divided into solution verification and code verification. Solution verification verifies whether the simulation output is similar to the original model's differential equations solutions, while code verification verifies whether the written code executes the intended algorithm (Roy, 2005).

The protective quality of static routing (Baltatu et al., 2000) was applied to verify the structure of the design for this research, and through verification, decisions were made based on demonstration runs and the objectives of the study in order to confirm existing theory. In this way, validity of the findings was strengthened and the experiment contributed creatively to building new theory (Davis, Eisenhardt & Bingham, 2007). According to Davis, Eisenhardt and Bingham (2007), an important part of developing theory is to verify the computational representation in order to ensure that it represents the theoretical logic. Davis, Eisenhardt and Bingham (2007:494) further postulated that "theory can be incremental in its advance provided insight is enhanced".

The experiment has demonstrated by way of the relationship between the real-world system and simulation, the feasibility of routing autonomous vehicles in a manner analogous to packet switching through IoT connectivity.

4.11 Conclusion

In Chapter 4, the researcher described the simulation process and the process of the replication as well as the development of construct theory and theoretical predictions in respect to the experiment. Results of the findings, compared with theoretical concepts and expectations, were analysed with the notion of developing a prototype. Sufficient indicators served to validate and verify the experiment and to develop a new algorithm, contributing to the pool of IoT, simulation and theory.

In Chapter 5, the researcher expounds briefly on the integration of the conceptual phenomenon into the working mechanisms of the model that epitomises the study designed to illustrate the virtual characteristics of routing of packets in a manner analogous to the real-life routing of autonomous vehicles.

CHAPTER 5: MODEL INTERPRETATION

5.1 Introduction

In this chapter, the researcher presents an overview of the concept that directed the development of the model characterised by this research. Prompted by theory, the model demonstrates that in alignment with the virtual packet switching routing mechanism that traverses data, a routing system could be created for traversing real-life autonomous vehicles. Figure 5.1 illustrates the consolidation of the concept within the design and operation of the model.

5.2 Theoretical background to model

Data converted into information through interplay among elements in the virtual and mental realms can become manifest in the physical world and is applied to the concept in this research. A model showing the dynamic and collaborative relationships in these realms that accomplish tasks and solve problems via IoT is depicted in Figure 1.1 (section 1.3.1). The method of communication using packets to transmit data across digital networks was considered viable as an IoT routing solution for real-life autonomous vehicles because of certain conceptual similarities (section 1.3). In addition, the effective reliability in delivering data during any packet switching process is evident, and became a mitigating factor in developing a concept where human lives are involved.

The autonomous vehicle domain is augmented by communication technologies within ITS through mechanisms such as AI and ML, albeit governed by legislative restrictions (section 2.1). The researcher determined that the multi-facet functioning of IoT vehicular technologies, particularly in routing autonomous vehicles, could be manipulated to correspond with the efficiency of the packet switching concept.

The researcher also resolved that due to the magnitude of the research, a cost-effective approach would be to develop a computer simulation model that effectively qualifies the theory. Simulation models can be used as instruments to understand real-life phenomena (Norton & Suppe, 2001) (section 3.3) and, when appropriately utilised, are employed as tools to verify engineering theories, as verification confirms that the implementation of the simulation model corresponds to the model (Monette, Sullivan & De Jong, 2002; Ulgen et al., 2014). Validation verifies that the model accurately mimics the real-world environment (section 1.9.1). Validity was later established with the running of the simulation. The model's reliability meant that it was able to produce consistent statistical results, and that the system would correct itself, should a unit fail, and continue its efficient operation (Monette, Sullivan & De Jong, 2002:117) (section 1.9.2). These aspects were confirmed with the consistency demonstrated by the duplication of the outcomes of the experiment (section 4.5.3).

Design theory and systems theory were perceived by the researcher as the most appropriate theories to guide this study (sections 2.71 & 2.72). Design theory deals with the desire that humans exhibit in wanting to better a complex situation by designing and developing an artefact (Simon, 1996:13) (section 2.7.1.1), with design within IT systems based on the progression of this refinement (Östman, 2005). Design theory can include abstract components, such as algorithms and models, which can be implemented into the real world as reflected in this research. Systems theory involves a complexity of components functioning as individual entities but interacting with one another to ensure that the system functions effectively as a whole (Von Bertalanffy, 1973; Ackoff, 1981:15-16) (section 2.7.2). Systems theory presents two schools of thought – one arguing that all properties must be studied in order to understand the whole, and the other proclaiming that the behaviour of the whole cannot be determined by studying the behaviour of individual parts. The multi-system characteristics of IoT interconnect into a whole, and could therefore result in different behaviour than that displayed by individual parts. As the research focused on a particular aspect of routing analogous to packet switching, the researcher chose to become familiar with the characteristics and functions of each element in its individual capacity, and to tackle issues of the whole as they arose. This was demonstrated throughout the building process of the model as each entity was individually dealt with, culminating in the successful completion of the operational system. The computer simulation model was constructed incorporating IoT vehicular wireless technologies to demonstrate the real-life environment described in the research, and to mimic a complex phenomenon accurately (Olivier, 2011) (sections 1.8.2 & 3.2.3).

5.3 Outline of simulation objective

Simulation brought to effect the consolidation of the concept in a real-life scenario through the operation of the model as shown in Figure 5.1.

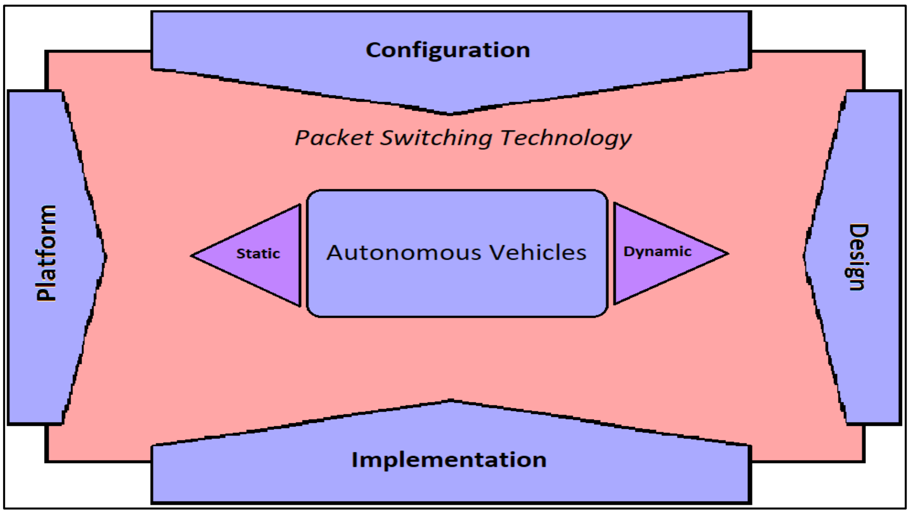


Figure 5.1: Diagram representing a concept consolidation model (Source: Researcher)

Psimulator2 is a graphical network open-source simulator and the instrument chosen to build the simulation model, as it comprises appropriate tools relevant to this study (section 1.8.2.1). The researcher hypothesised that the virtual packet switching routing mechanism that moves data across a network could create, through simulated adaptation, an analogous routing system for moving real-life autonomous vehicles.

The simulator only supports static routing protocol, where network routes specifying the next hop router IP address are manually entered into a router's routing table to forward traffic to a predetermined destination (Baltatu et al., 2000) (section 4.3).

The GUI (front-end component) and emulator (back-end component) of the simulator work in parallel to each other (section 1.8.2.1). The GUI is the virtual network of the simulation model, capturing and displaying packets, which, together with a network of building blocks, were used to design, configure and implement a network topology through simple animation. These simulated components fundamentally comprise organising tools, graphic components and element tools/entities allowing the researcher scope for manipulation. At the same time, dynamic and static elements permitting graphical depictions of existing network components can be substituted by those elements characteristic of a road system, all merged within the realm of IoT vehicular technology, allowing the transmission status of packets to be viewed.

The specific functions of these components include two elements:

i) **Dynamic:** ARP signals sent to devices to 'confirm' the route for vehicles, and ICMP signals, which were essentially the actual vehicles themselves that were forwarded along designated routes.

ii) **Static:** These included routers, switches, nodes and links that allowed for designing, configuring and linking network components, and implementing the uncomplicated network topology in the form of wireless beacons, creating static routes that were loaded into the routing table via Cisco commands.

Element tools translated into the following elements (see table 3.1):

i) **Platform:** Essential scaffolding, developed in Java to operate on the Windows platform, was available for the study (section 3.3, step 3) (Figure 5.2).

ii) **Design:** Redesign: Allowed for the incorporation of IoT vehicular technologies to be accommodated within the design:

- **Adaptation –** end devices were transformed into suitable *en route* places or destinations for autonomous vehicles to head for
- **Modification –** routing devices were modified for autonomous vehicles travelling to a destination along a suitable roadway infrastructure (Figure 5.2)

iii) **Configuration:** This was done via the built-in terminal client; the emulator used to allow telnet sessions to configure the IP addresses of each interface, and to run ICMP requests. Built-in tools were used to capture a set of instructions, which became the accumulated data to be analysed (Figure 5.3).

iv) **Implementation:** For the simulator, Java files were compiled and each component was individually run. All configuration and topology meta-data were saved in the system in order to build an XML file, which was subsequently fed to the emulator to imitate the design (Figure 5.3). Careful consideration will need to be exercised when implementing the project on a large scale, which should begin with planning and deciding on the necessary building blocks required for implementation.

v) **Packet Switching:** The foundational component that encompasses the other four components and is the driving force behind the system by way of applying the fundamental principles of packet switching technology and thus, allowing AVs to navigate an unabridged infrastructure (Figure 5.3).

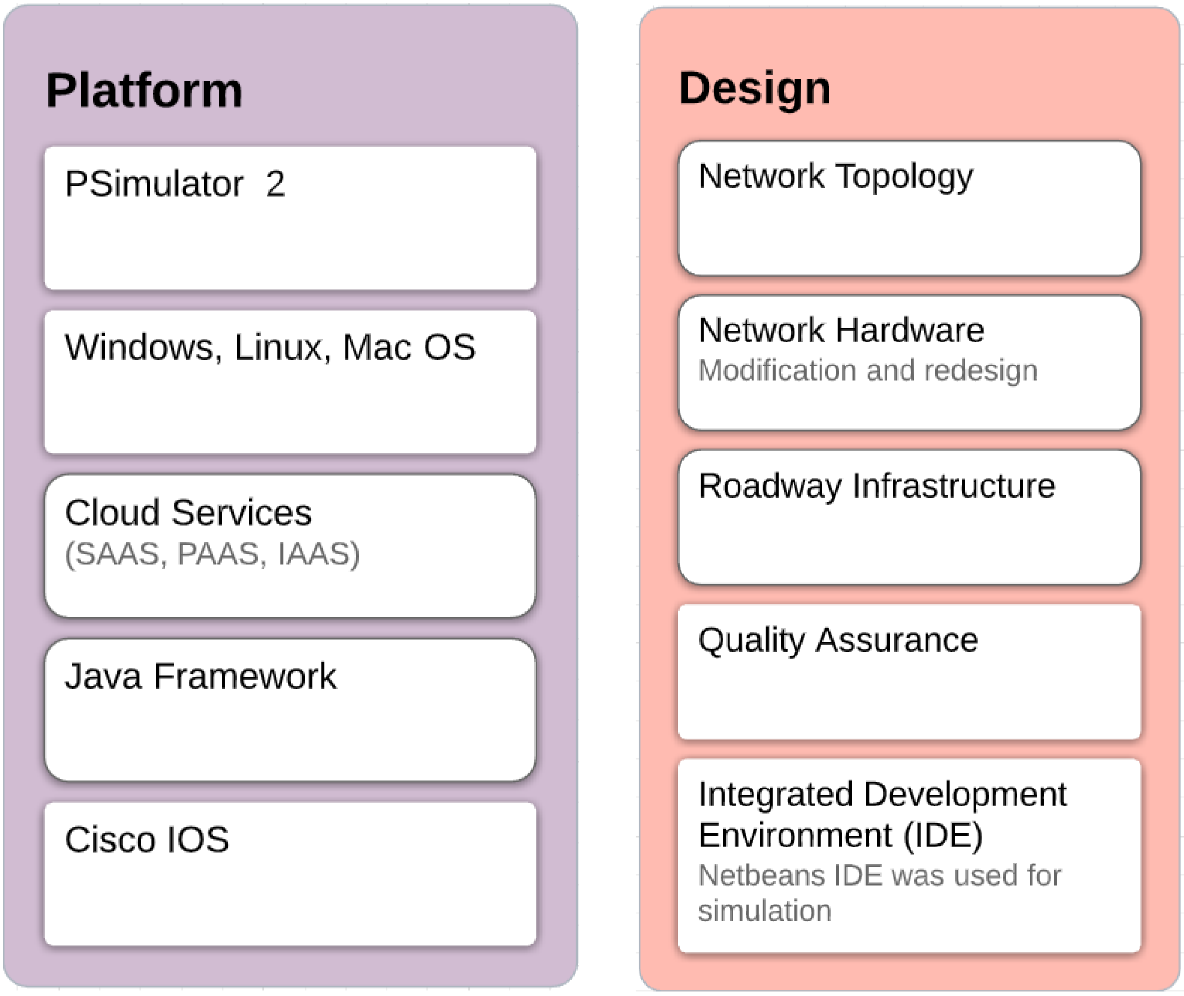


Figure 5.2: Platform and Design constituents of the consolidation model
(Source: Researcher)

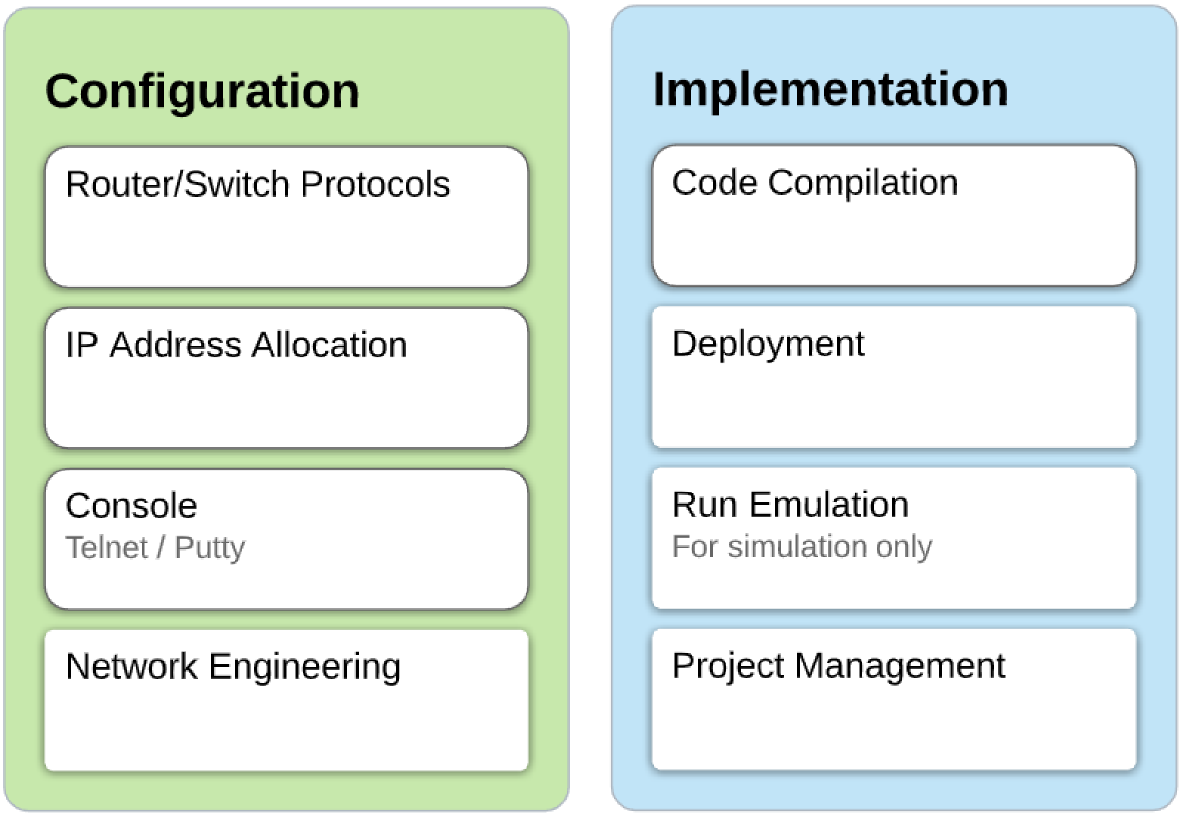


Figure 5.3: Configuration and Implementation constituents of the consolidation model (Source: Researcher)

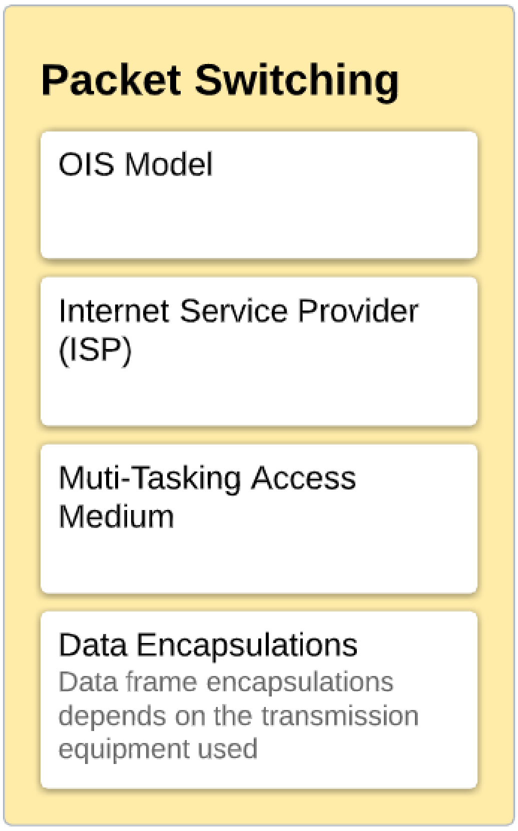


Figure 5.4: Packet Switching constituent of the consolidation model (Source: Researcher)

Once design was completed via the GUI, communication was established with the emulator, which imitated the real-life scenario under study.

5.4 Overview

Once the GUI virtual design was completed, the code for the emulator was compiled using Gradle and the emulator was subsequently started. Communication was then established between the front-end and back-end components and the two components began working together to run the simulation through the emulator, replicating the real-life situation (section 1.8.2.1). The simulation mimicked the concept of routing autonomous vehicles in a manner analogous to the principles of packet switched networks, verifying the conceptual validation against that of the operational validation.

Data were captured by the emulator as requests were routed from source to destination. Next, the hop router IP address and the transmission status of packets were viewed via the routing table (section 3.3, step 5). Variables in the system were regarded as distance, noted as time between hops, and traffic density affecting time of arrival. The quantitative data represented in the output logs were saved by the model and observed to emulate the real-world aim of the research.

A subsequent test conducted using different input data to purposely route a vehicle to a non-existent destination along a prevailing network was demonstrated and reflected in the hop count state (sections 3.3, 3.4 & 4.3.7.3). This also served to verify the effectiveness of the model in reacting to threatening situations that could develop in a real-life routing situation.

5.5 Conclusion

An overview of the conceptual-to-operational mechanisms of the model was provided in this chapter and the 'transition' from virtual to real world was substantiated through the instruments of simulation, observation and theory.

In Chapter 6, the researcher addresses the research objectives and consequent findings of the research. The advantages of a constructed prototype is also discussed, together with future research suggestions and the influence of the findings on individuals and society.

CHAPTER 6: FINDINGS, RECOMMENDATIONS AND CONCLUSION

6.1 Introduction

The researcher has presented a solution to the routing of an autonomous vehicle by adopting packet switching technology. The objectives of this research, as stated in the first chapter, were to include the description of packet switching and the subsequent design of a model towards the development of an algorithm that represents a real-life environment and the implementation of a solution resulting in the routing of autonomous vehicles. Ancillary regard was afforded to the complexity of sensor systems and other IoT-connected elements integral to autonomous vehicle technology in order to home in on the core aspect of the research, which was to demonstrate routing in a manner analogous to packet switching, while capturing the critical properties of reality.

In this final chapter, the researcher provides an overview of the research results and findings, with recommendations for further research presented and limitations considered.

6.2 Overview of thesis

The background and objectives of the study and the research methodology employed were presented in Chapter 1. The need for appropriate solutions to the reduction of road fatalities and of congestion were emphasised and a solution was proposed for the operation of autonomous vehicles. In the introductory chapter, the researcher explained that a simulation model in the form of Psimulator2 (Figure 1.2) would be used for the experiment, since the researcher considered this approach suitable to the nature of the research.

In Chapter 2, the researcher delved into the literature available on packet switching technology and autonomous vehicles. In addition, marrying the concepts encompassing IoT VANET technologies that included communications, the concepts of AI, LiDAR, VOR and V2I within the analogy were discussed. In order to realise the concept, the focus was on the routing of such vehicles, given that a route is any path from a source to a destination, whether it entails network traffic or vehicles on a road network. A review of the literature found that research into this matter was limited. The technological impact on society was also discussed and the chapter culminated in a discussion about theory that characterises the study.

Analyses of characteristics of various models were discussed in order to determine the most feasible for simulating packet switching and the routing of an autonomous vehicle. To address the objectives of the research, a model was built and a systematic description of the research design and methodology was presented in Chapter 3.

In Chapter 4, the researcher described the simulation process and subsequent simulation results. The researcher further elaborated on the procedure and outcomes of replication when testing the model. An analysis was conducted for the purposes of developing a prototype, and validation of the simulation was presented. A new algorithm to route autonomous vehicles, developed through the relationships encompassing IoT vehicle connectivity, simulation and theory was proposed.

6.3 Findings

The primary and secondary research questions posed in Chapter 1 are answered when the research objectives of the study have been met, as the research objectives and the research questions are aligned. The secondary research objectives, encapsulated by the primary research objective, have all been reached, and with that, the research outcomes have been validated. The outcomes (or findings) of each secondary research objective are summarised below.

6.3.1 Findings of Secondary Research Objective 1

SRO1: To determine the state-of-the-art technologies supporting autonomous automobiles

The routing of AVs was highlighted due to congruence with routing of packets in packet switching and a major solution to controlling AVs. As all high-tech features of vehicular autonomy converge into IoT vehicular networks ((Sherazi et al., 2019), the influence of ubiquitous connectivity benefits the routing attribute of AVs directly. The literature describes the latest technologies by drawing attention to some key findings relative to the research, which include:

- The proposed G5 ultra-high-speed mobile communication network currently being developed with security in mind to accommodate the IoT multi-service platform: Latencies with V2V communications will be swiftly dealt with and intermediary devices will become redundant, allowing for direct D2D communication (section 2.2.4.5). Increased reliability is expected to speed up the routing of vehicles, as a hop gain is acquired by bypassing any base station. Furthermore, 5G has also been proposed as a solution to platooning in order to improve present and unreliable ITS-G5 communication systems greatly, and to increase traffic flow (Rashdan, De Pont Müller & Sand, 2017) (section 2.6.1).
- VANETs utilising AI, ML and sensors are recognised by the IEEE as an inextricable component of autonomous vehicles (section 2.6.1). Routing techniques using VANETS proposed by Abbasi and Kahn (2018) (section 2.6.1) determined that position-based unicast routing protocol centred on location information would be the most appropriate approach in a city environment; however, linkage problems resulted

in packet loss due to high mobility and intermittent connectivity (Abbasi & Kahn, 2018) (sections 2.6.1 & 5.3.5). This pointed to a secure optimal path-based effective routing protocol that was needed to accommodate the problematic linkage issues.

- Switching, modes of connectivity and bandwidth availability through the IEEE 802.11p WAVE standard are limiting factors in VANET infrastructure with respect to data forwarding-based applications, with high mobility and frequently changing topology being inhibiting factors of the VANET environment. Such heterogeneous applications would require greater bandwidth and continuous network connectivity – which is as yet unavailable (Sherazi et al., 2019) (section 2.6.1).
- Short path algorithms were found to be ineffective in determining minimum short path on a dynamic stochastic network on conventional roads because of other determining factors such as distance, road works, driver's speed, congestion and other detractions (Alazab et al., 2011). This called for the optimal route to be policy-based and suggested modifying Dijkstra's algorithm by implementing an optimal routing algorithm incorporating traffic routing with real-time traffic flow information (section 2.6.2).
- Applying an extension to existing methods for network packet routing similar to reinforcement learning used by taxis, where each taxi's path history (including ETA to each respective node and destination) is stored in memory and used to update previous nodes. Q-system was considered unsuitable as other taxi locations and congestion factors were unknown and, in addition, the Q medium increased considerably with the number of nodes (Cox, Jennings & Krukowski, 2014) (section 2.6.2).
- Providing on-demand mobility was to ensure that by introducing a network flow model, a fleet of autonomous vehicles would be rebalanced on conventional roads, with capacity limitations regulated on nodes (intersections and locations) and road links. For network efficacy, the number of vehicles entering a node would need to equate to the exiting numbers (Zhang, Rossi & Pavone, 2016; Templeton, 2015; Barnard, 2016) (section 2.6.2).
- Autonomous vehicle in-built self-healing devices utilising a Software Defined In-Vehicle Networking (SDIVN) system could enable a standby link to automatically take over a failed primary link during emergency situations, like braking and lane centring (Halba, Mahmoudi & Griffor, 2018) (sections 2.6.2 & 5.3.3).
- Source routing, developed by IETF as an alternative mechanism to shortest-path first strategy, entails packets forwarded on routing information attached to the packets themselves, rather than stored at the provisional routers. This is expected to optimise traffic flow in large-scale networks (Reuter & Cordeiro, 2016) (section 2.6.2).

- The LiDAR system (section 2.3.2.1) incorporated into all conventional autonomous vehicles constructs a three-dimensional map of its environment by gathering information in every direction; however, during adverse weather conditions sensory ability is compromised.
- Malfunctions in GPS systems could be countered as the uncomplicated packet switching routing mechanism is applied instead. Problems confronted in the packet switching procedure, such as latency or congestion, are automatically resolved by the system itself (Hanley, 2016) (section 1.3 & 2.1).

6.3.2 Findings of Secondary Research Objective 2

SRO2: To determine the technologies involving packet switching

Analogies between the routing characteristics of packet switching and the routing of AVs were the focus of the research as the key factor for controlling the vehicles. This led to the investigation of packet switching technologies in the literature, with emphasis on various packet switching mechanisms used by routers as well as routing mechanisms and protocols. Further investigations regarding significantly fused autonomous vehicle technology were conducted, and findings concerning routing methodologies are described above in point 5.3.1. Deductions relevant to incorporating packet switching in routing autonomous vehicles were made as follows:

- A forwarding system of packets can be incorporated in on-board vehicle computer software to warn vehicles of braking ahead (Fawaz & Artail, 2010) (section 2.6)
- Load balancing distributes workload over numerous paths or processors to decrease congestion in a network. Should a primary path fail because of congestion, alternative paths are used (section 2.2.1.1)
- Instructions regarding the route, provided by the static procedure, would be remembered (learned) by the algorithm. This knowledge acquired by the algorithm through experience becomes stored (McCarthy & Feigenbaum, 1990; Turing, 1950) (section 2.3.2) and applied to route each subsequent vehicle as demonstrated in the experiment through replication of the model (sections 2.3.1 & 4.5) and eliminates issues should individual on-board vehicle sensors malfunction
- A router hosting a destination network, even from an alternative source as shown in the experiment, would disallow vehicles to traverse a network section once different data input is fed from a source to a non-existent destination. This would verify the reliability of the experiment and its safety effectiveness in application to the real-world scenario
- Vehicles prohibited from reaching their fictitious destination were routed back to their original address. In a real-world scenario, because of the time-wasting factor, a virtual

request can be sent to the destination address first, awaiting a reply before mobilising the vehicle. This can determine whether the model is capable of adapting to, and dealing accordingly with unexpected situations that could develop (Ackoff, 1981) (section 2.7.2.3)

- ICMP signals based on ARP information were forwarded to the specific routes in a manner analogous to the vehicle routed in the demonstration. This indicated that in reality, when several paths and links are involved, routing protocols need to have established routes in order to avoid congestion and maintain safety. As such, similar broadcast information should be conveyed across the network and back to the requesting autonomous vehicle, possibly via IoT VANET technology encompassing V2I to validate the journey to be undertaken beforehand
- Segment routing pertaining to source routing mechanisms is a procedure where the shortest-path-first scenario is avoided when a path is fraught with problems. This means that packets, instead of being stored at the in-between routers, are forwarded with routing information affixed to themselves in order to increase packet traffic flow. A segment is the instruction carried out by a node on an incoming packet (Reuter & Cordeiro, 2016) (sections 2.6.2 & 5.3.1)

6.3.3 Findings of Secondary Research Objective 3

SRO3: To identify the availability of security technologies to safeguard system security against attack

The research centred on the feasibility of implementing a vehicle routing system similar to the packet switching concept through IoT modelling. Though security was considered a vital safety constituent of the process, to visit this area in depth would have entailed a diversion from the purpose of this research. Nevertheless, the following security factors in respect of both failure of the system and cyber-attack were considered relevant:

Access to any Electrical Control Unit (ECU) of a vehicle would alter its behaviour; therefore, a Software Defined Standby (SDN) link programmed for possible ECU failure situations, with in-built repair mechanisms, can take over in an emergency to ensure timely message distribution. Furthermore, SDN program features permit manipulation of routing tables and routing engines and the execution of features to protect in-vehicle networks from failure (Halba, Mahmoudi & Griffor, 2018; Tuohy et al., 2015) (sections 2.6.2 & 5.3.1).

Manual configuration functional in static routing from source to destination restricts security intrusion in wireless transmission within the model (Baltatu et al., 2000), but redundancy was considered in the IoT real-world connectivity aspect with its myriad of links, should connection to a device be lost, and routing information, hence, updated (section 3.4).

Encryption of communication between V2V and V2D would be taken into account in a real-life scenario.

Cloud networks that store and process data that can be easily accessed are commonly used by organisations working with IoT (section 2.3.3.2). Multiple source data directed to a single destination requires IPv6 which supports routing protocols, both internal (RIP, EIGRP) and external (BGP), that are connected networks with enhanced security. Such devices operating in a stateless address configuration protocol can configure themselves automatically with an IPv6 address (Krishnan et al., 2012) (sections 2.2.4.4 & 2.3.3.2).

The introduction of G5 for mobile communication was considered, prioritising security to serve IoT effectively, offering ultra-low latencies and ultra-high reliability, while bypassing intermediary devices to allow direct D2D communication (sections 2.2.4.5 & 5.3.1). Meantime, the National Highway Traffic Safety Administration (NHTSA) is unremittingly developing cybersecurity guidelines and best practice databases in order to safeguard IoT automotive computer systems (section 2.4.31).

6.3.4 Findings of Secondary Research Objective 4

SRO4: To describe the environmental and societal benefits of developing the system

The societal benefits found in similar research areas are explained in the literature (section 2.4) and some of the main points with respect to improving quality of life by developing the system proposed in this research are reiterated herein, together with benefits as a direct result of this research:

- Reduction of total dependency on sensor systems and on GPS, since routing mechanisms that may be disrupted do not always provide the best route (section 1.12). Substantial V2I and V2V device investment can be reduced as not every traffic signal requires to be equipped with a radio in applying the packet switching concept
- Reduction of traffic congestion and saving on fuel consumption because a drop in stop-and-go wave actions (section 2.5)
- Air pollution and greenhouse gas emission drastically reduced, with a marked improvement in public health
- Economic viability to be taken into account alongside existing transport infrastructure in a common cost benefit framework with original equipment manufacturers entering the market as well AV dealers, chemical engineering, electric utilities, IT hardware and software and Telecommunications (Khayyam, 2020). A considerable number of lives saved annually and injury numbers significantly reduced, with favourable cost benefits to the economy. GDP will also improve as less time will be spent driving, and

work productivity will increase (Triplett et al., 2014) (section 1.3.2). In addition, the beverage and restaurant sector is expected to profit (Khayyam, 2020)

- New potential market opportunities will open as AVs are predicted to generate $7 trillion annual revenue by 2050 (Lanctot, 2017).
- Although car sales will drop, since fewer people will need to own vehicles, emerging technologies in computing platforms will enable business competitiveness the integration of ML and AI into the IoT design process. New players such as TaaS providers entering the autonomous vehicle market will offer employment opportunities and a better quality of life (Daugherty et al., 2015) (sections 2.5 & 2.6). New technologies influence the type of labour required by the economy, so opinions that AI will be instrumental in a jobless future are not supported by concrete evidence (Polson & Scott, 2018). In addition, experience in sectors such as aviation and rail indicate that the increased application of sophisticated technology will result in significant increases in maintenance costs, further securing employment prospects (Johnson, 2017)
- Mobility-on-demand will cut travel costs and reduce pollution and the need for parking (Zhang, Rossi & Pavone, 2016) (section 2.6.2)
- Improving quality of life for elderly and disabled persons by enabling them to achieve a level of independence (Giarratana, 2016; Jansen, Li & Lorenz, 1995) (section 1.3.2)

6.3.5 Findings of Secondary Research Objective 5

SRO5: To determine the challenges surrounding the technical implementation of the system

The simulation mimicked a phenomenon, showing how acting parts would react given a specific set of instructions and boundary conditions, described as the best method to predict the behaviour of a real-life complex system and substantiated through accuracy and consistency of measurements (Bedau, 2011) (section 4.7). The challenge remained to transform the simulation into an uncomplicated basic demonstration of routing a vehicle, given a set of instructions, in a manner analogous to the concept of packet switching, into the real-life situation.

The experiment did not make provision for routing autonomous vehicles among conventional vehicles on prevailing overly congested and unmodified roadways, on infrastructure that is not regularly maintained and with unexpected distractions. Herein lies the challenge of implementing the system on existing roads. It therefore becomes unquestionable to exclude the safety-related applications that VANETs provide from the IoT vehicular communication biosphere, even though they themselves are subject to several disconcerting issues. VANET problematic areas include connectivity issues, network congestion slowing down overall performance and latency in finding appropriate routes because of storage problems (Abbasi & Kahn, 2018) (section 2.6.1). Other issues presented include the complacency of drivers in a partially autonomous setting when the sudden need to alertness arises.

Because of the multi-faceted characteristics of vehicular networks that are integrated in IoT, it has become increasingly more challenging to manage increasing data rates, seamless connectivity and reliability, which also includes implementing a safe and secure routing protocol, since human lives depend on accuracy in functioning. Tesla's autopilot was recently tricked by Chinese hackers into switching lanes to avoid what the autopilot perceived to be a legitimate obstacle, by means of computing an 'adversarial example' (Huddleston, 2019). Adversarial example is merely an optical illusion created in computer vision that can deceive the system.

Introducing autonomous vehicles would disrupt present infrastructures. To overcome this challenge, Templeton (2015) and Barnard (2016) suggest that overall capacity of vehicle numbers be regulated by rebalancing the number of fully autonomous vehicles on the road once customers have reached their destinations (section 2.6.1). Capacity constraints should be implemented on nodes and road links, where vehicles travel at the speed limit of the link at which the link is able to contain congestion, with flow rate decreasing to avoid congestion (Zhang, Rossi & Pavone, 2016) (section 2.6.1).

Allowing autonomous vehicles on conventional roads using existing infrastructure presents challenges with legislation, which would have to be changed in order to accommodate vehicles running at different levels of automation. This could lead to ethical concerns in the event of an accident (section 1.11).

6.3.6 Findings of Secondary Research Objective 6

SRO6: To propose an IoT model to validate the operation of packet switching as a routing mechanism for autonomous vehicles in the South African context

- The alarming South African CO^2 emission statistics (section 1.3) and increasing congestion on our roads are motivating factors to consider, at very least, the implementation of a rudimentary system that runs fully autonomous vehicles on segregated roads such as at airports or at resorts, with infrastructure to facilitate further improvements. Each incremental improvement would be instrumental in further job creation (Daugherty et al., 2015) (section 2.4).
- A further option would be to consider Zhang, Rossi and Pavone's (2016) rebalanced algorithms to provide on-demand mobility for the taxi industry via a flow model applying intelligent routing in order to avoid areas of congestion (section 2.6.2), for

example, the central business district. Rebalancing would require that the number of autonomous vehicles exiting an area should be equal to those entering the area. Capacity constraints enforced through legislation would allow operators to operate in allocated localities.

- New technologies in IoT and maintenance of infrastructure would offer employment opportunities, with the placement of personnel in areas such as robotics, engineering, data science, software development, operations, testing, marketing and sales. Legislation as an employer would require additional workers to ensure the upholding of the law in the AV industry, while revision of our education system should be considered in order to accommodate the fourth industrial revolution and incorporate lifelong learning.

6.3.7 Findings of Primary Research Objective

PRO: To determine, by means of a proposed IoT solution, the viability of applying the virtual packet switching concept of traversing data as a routing mechanism to real-life autonomous vehicles

Addressing the findings of the secondary objectives of the research as mentioned above (section 5.3) contributed towards determining the viability of the primary objective. The suggestion of applying the concept of packet switching as routing mechanism for autonomous vehicles was effectively demonstrated by the model in a step-by-step configuration. Routing instructions eliminated any potential failure as a vehicle was seen to conform to what can be interpreted as an allotted source and destination address, in a manner analogous to a packet traversing a network. This would result in greater dependability in applying the demonstrated routing concept in the real world, with less reliance on GPS and costly sensor systems that could malfunction at times because of hostile weather conditions, as well as irregular connectivity often experienced with VANETS.

6.4 Contribution

This study provided insights into the functioning mechanisms of packet switching networks and the autonomous vehicle industry, while the empirical simulation model was observed to represent reality accurately. Validity and verification, strengthened through step-by-step specifications and comparisons in replication, confirmed the novel and effective implementation of routing of autonomous vehicles in a manner analogous to the concept of packet switching (Kampis, 2013; Sargent, 2005).

The probability of adopting packet switching as a router mechanism for autonomous vehicles, which can contribute further towards reliable, cost-effective and faster methods of human transportation, has been achieved in this study.

The study has also conclusively added to the body of knowledge that will keep growing with the advancement of vehicular IoT connectivity.

6.5 Recommendations for further research

Routing of autonomous vehicles along the lines of packet switching opens doors and further enhances the progression of IoT technologies such as all-optical satellite communications systems.

Further research could also be conducted on source routing mechanisms such as segment routing that avoid the shortest-path-first scenario when other factors determine the best route to a destination to be unsuitable. This means that packets are forwarded with routing information attached to themselves, instead of stored at intermediate routers. At the same time, possible attacks from within the routing domain of a segment need to be investigated.

Research to enhance the routing performance of VANETs further is needed, since Abbasi and Khan (2018) claim that protocols do not reflect environment characteristics accurately. VANET communication in a possible infrastructure-less environment also needs to be investigated in preparation for the integration of vehicular networks into G5 mobile technology (Cavalcanti et al., 2018) (section 2.6.1).

Investigation into the probability of incorporating VHF–VOR in combination with VANET technology while utilising the concept of packet switching as routing means could yield effective results.

An efficient operating system on which human life depends, calls for guidelines to help secure IoT automotive computer systems. Additional research in the area of in-built repair mechanisms utilising SDN, where standby links automatically take over once a primary link fails, needs to be undertaken.

6.6 Conclusion

A summary of the research process was presented in this chapter, together with findings from the literature, followed by contributions of this study to the discipline’s body of knowledge, and recommendations for further research. A concluding deduction by the researcher is summarised as follows:

AI is a proponent of information exchange in the all-encompassing VANET realm of IoT that has pervaded the automotive domain, engulfing the latest generation of vehicles in its machine-learning wake, and destined ultimately to oust humans from the driving seat. Infrastructure, nonetheless, has not been designed to accommodate autonomous vehicles and they encounter unforeseen obstacles and perpetual changes such as road works, new road signs and the erratic behaviour of conventional drivers and pedestrians, despite the fact that AI devices are used to assist in identifying these aspects.

Decisions regarding expenditure earmarked for changing the status quo direct that partially connected autonomous vehicles will continue to be incrementally enhanced (Litman, 2020) and will share the roadway with conventional vehicles in a hybrid environment for some time. Fresh ideas have surfaced, designed to manage both fully autonomous vehicles on segregated roadways and partially autonomous vehicles alongside conventional vehicles, and both scenarios have focused on routing.

A journey of any kind involves travelling a route from a source to a destination. Packet switching is a virtual communication system that allows data to travel from a source to a pre-determined destination by means of a series of coded instructions and events. The experiment that was undertaken in this study has shown that the packet switching network concept effectively correlates with directing the routing of real-life autonomous vehicles that too, depart from a source address to arrive at a destination.

REFERENCE LIST

AAAFTS *see* American Automobile Association Foundation for Traffic Safety.

CAR *see* Centre for Automotive Research.

CCNA see Cisco CCNA Academy.

CSIR *see* Council for Scientific & Industrial Research.

ETSC *see* European Transport Safety Council.

IEEE *see* Institute of Electrical and Electronics Engineers.

NHTSA *see* National Highway Traffic Safety Administration.

OCF *see* Open Computing Facility.

RTMC *see* Road Traffic Management Corporation.

SAEI *see* Society of Automotive Engineers International.

UNESCO *see* United Nations Educational and Scientific Council.

UNFCCC *see* United Nations Framework Convention on Climate Change.

WHO *see* World Health Organisation.

Abbasi, I. A. & Kahn, A. S. (2018). A review of vehicle to vehicle communication protocols for VANETs in the urban environment. *Future Internet,* 10(2). Open Access.

Abbasi, I. A., Nazir, B., Bilal, S. M. & Madani, S. A. (2014). A traffic flow oriented routing protocol for VANET. *EURASIP Journal of Wireless & Communication Networks,* Art. 121:121.

Abdalla, G. M. T, Aburgheff, M. A. & Senouci, S. M. (2008). Current trends in vehicular ad hoc networks. *Ubiquitous Computing & Communication Journal,* (6):1-13.

AccessScience Editors. (2017). *Potential impacts of self-driving cars*. McGraw-Hill Global Education. doi:https://doi.org/10.1036/1097-8542.BR0512171.

Ackerman, E. (2016). *Cheap LiDAR: The key to making self-driving vehicles affordable*. IEEE Spectrum, 22 September 2016. [Online]. Available from: http://spectrum.ieee.org/transportation/advanced-cars/cheap-lidar-the-key-to-making-selfdriving-cars-affordable [Accessed: 3 October 2017]

Ackoff, R. L. (1981). *Creating the corporate future.* New York: John Wiley & Sons.

Alippi, C. (2014). *Intelligence for embedded systems.* Springer Verlag. E-book: ISBN: 978-3-319-05278-6. [Online]. Available from: http://www.springer.com/gp/book/9783319052779 [Accessed: 16 November 2017]

Alazab, A., Venkatraman, S., Abawajy, J. & Alazab, M. (2011). An optimal transportation routing approach using GIS-based dynamic traffic flows. *Proceedings.* 3rd International Conference on Information and Financial Engineering (*IPEDR)*, Singapore, 22 Feb. p. 12. ACSIT Press.

American Automobile Association Foundation for Traffic Safety. (2018). *Annual Report.* [Online]. Available from: https://www.itf-oecd.org/sites/default/files/docs/irtad-road-safety-annual-report-2018_0.pdf [Accessed: 2 October 2018]

Andrews, J. G., Stefano, B., Wan Choi, S, V., Lozano, H. A., Soong, A. C. K & Zhang, J. C. (2014). What will 5G be? Selected Areas in Communications. *IEEE Journal,* (32)6:1065-1082.

Anu, M., Andradóttir, S., Healy, K. J., Withers, D. H. & Nelson, B. L. (eds). (1997). White Paper introduction to modelling and simulation. *Proceedings*. The 1997 Winter Simulation Conference Department of Systems, Science and Industrial Engineering, State Universities of New York at Binghampton, NY, December. pp. 7-13.

Arbib, J. & Seba, T. (2017a*). Disruption, implications and choices. Rethinking transportation 2020-2030*. A RethinkX Sector Disruption Report. USA.

Arbib, J. & Seba, T. (2017b). *Rethinking transportation 2020-2030. The disruption of transportation and the collapse of the internal-combustion vehicle and oil industries*. [Online]. Available from: https://www.wsdot.wa.gov/publications/fulltext/ProjectDev/PSEProgram/Disruption-of-Transportation.pdf [Accessed: 13 December 2017]

Arpan, P. (2015). Internet of Things: Making the hype a reality. *IT Professional,* 17(3):2-4. IEEE. doi:10.1109/MITP.2015.36.

Ashton, K. (2009). *That 'Internet of Things' thing.* [Online]. Available from: http://www.rfidjournal.com/articles/view?4986 [Accessed: 11 November 2017]

Australian Civil Aviation Authority*.* (n.d.). *Operational notes on VHF Omni Range (VOR).* Contents Booklet published by the Civil Aviation Authority. [Online]. Available from: https://www.casa.gov.au/files/vhf-omni-range-vor-0 [Accessed: 23 June 2018]

Automobile Association. (2018). *SA road deaths a national crisis: 134 000 + killed over 10 years – AA*. Wheels 24. [Online]. Available from: https://www.wheels24.co.za/News/Guides_and_Lists/sa-road-deaths-a-national-crisis-134-000-killed-over-10-years-aa-20180424 [Accessed: 2 October 2018]

Awad, E., Dsouza, S., Kim, R., Schulz, J., Henrich, J., Shariff, A., Bonnefon, J.-F. & Rahwan, I. (2018). The Moral Machine experiment. *Nature*, (563):59-64.

Bacon, F. (1620). *Novum Organum.* England.

Baltatu, M., Lioy, A., Maino, F. & Mazzocchi, D. (2000). Security issues in control, management and routing protocols. *Proceedings*. Terena Networking Conference, Lisbon, Portugal, 22-25 May. pp. 1-12.

Banks, J. & Carson, J. S. (1990). *Discrete-Event system simulation.* NJ. Prentice-Hall.

Baomar, H. & Bentley, P. J. (2016). An intelligent autopilot system that learns piloting skills from human pilots by imitation. *Proceedings.* International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA, 7-10 June. doi:10.1109/ICUAS.2016.7502578.

Barberousse, A., Franceschelli, S. & Imbert, C. (2009). Computer simulations as experiments. *Synthese,* 169(3):557-574.

Barnard, M. (2016). *Autonomous cars likely to increase congestion?* [Online]. Available from: https://cleantechnica.com/2016/01/17/autonomous-cars-likely-increase-congestion/ [Accessed: 23 January 2017]

Bedau, M. A. (2011). Weak emergence and computer simulation. *In* Humphreys, P. & Imbert, C. (ed), *Models, simulations, and representations*. New York: Routledge. pp. 91-114.

Belyaev, M. & Gaivoronski, S. (2014). Towards load balancing in SDN-networks during DDoS attacks. *Proceedings.* First International Science and Technology Conference (Modern Networking Technologies) (MoNeTeC), Moscow, 28-29 October. pp. 1-6. IEEE.

Berg, C. (2018). *Currently what's the biggest obstacle in self-driving?* Aerospace Engineer at Rockwell Collins. [Online]. Available from: https://www.quora.com/Currently-whats-the-biggest-obstacle-in-self-driving [Accessed: 16 November 2018]

Bertnaga, P. (2010). *How does a GPS tracking system work?* [Online]. Available from: https://www.eetimes.com/how-does-a-gps-tracking-system-work/# [Accessed: 12 February 2020]

Bhattacherjee, A. (2012). *Social science research: Principles, methods and practices.* Textbooks Collection, Book 3. University of South Florida: USF Tampa Library Open Access Collections.

Bianco, A., Birke, R., Giraudo, L. & Palacin, M. (2010). Openflow switching: Data plane performance. *Proceedings.* International Conference on Communications (ICC), Cape Town, RSA, 23-27 May. pp. 1-5. IEEE.

Bilal, S. M., Khan, A. R. & Ali, S. (2017). Review and performance analysis of position based routing in VANETS. *Wireless Personal Communications,* (94)3:559-578. doi:10.3390/fi10020014.

Bilal, S. M., Madani, S. A. & Khan, I.A. (2011). Enhanced junction selection mechanism for routing protocol in VANETs. *International Arabic Journal of Information Technology,* 8(4):422-429.

Bishop, C. M. (2013). 2013 Model-based machine learning. *Philosophical transactions of the Royal Society of London* A, 371. http://dx.doi.org/10.1098/rsta.2012.0222.

Bode, H., Mosteller, F., Tukey, F. & Winsor, C. (1949). The education of a scientific generalist. *Science,* 9(2480):553-558. doi:10.1126/science.109.2840.553.

Botswick, G. J. & Kyte, N. S. (1981). Measurement. *In* Grinnell, R. M. (ed), *Social research and evaluation*. Itasca, IL: Peacock.

Brown, E. (2016). *Who needs the Internet of Things?* [Online]. Available from: http://www.linux.com/news/who-needs-internet-things [Accessed: 12 November 2017]

Campbell, D. T. & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Chicago: Rand McNally.

Carnap, R. 1934. *The Unity of Science*. London.

Cavalcanti, E. R., De Sousa, J. A. R., Spohn, M. A., Gomes, R. C. & Da Costa, F. B. F. (2018). VANETs research over the past decade: Overview, credibility and trends. *Computer Communication Review,* 48(2):31-39.

Centre for Automotive Research. (2014). *Self-driving cars: The next revolution.* KPMG.

Centre of Technological Excellence. (2020). *Technology is advancing at a phenomenal pace...* Facebook comment. [Online]. Available from: https://www.facebook.com/CTEions/posts/101430318114488 [Accessed 13 February 2020].

Chambers, J. (2011). *Cisco and the World IPv6 Internet launch.* [Online]. Available from: http://www.cisco.com/go/ipv6 [Accessed: 13 September 2019]

Chenxi, L., Van Eenennaam, E. M., Wolterink, W. K., Ploeg, J., Karagiannis, G. & Heijenk, G. (2012). Evaluation of CACC string stability using SUMO, Simulink, and OMNeT++. *EURASIP Journal on Wireless Communications and Networking,* Art. 1:1-12.

Chou, D. C. & Chou, A. Y. (2012). Awareness of Green IT and its value model. *Computer Standards and Interfaces,* 34(5):447-451.

Churchman, M. (2017). *Simulated vs. real-device testing.* Sauce Labs. [Online]. Available from: https://saucelabs.com/blog/simulated-vs-real-device-testing [Accessed: 3 October 2017]

Cioara, J., Minutella, D. & Stevenson, H. (2008). *Exam prep CCNA: Your complete certification solution*. 2nd ed. U.S.A.: Library of Congress.

Cisco. (2011). *10S IP Configuration Guide Release 12.2.* San Jose, CA: Cisco.

Cisco CCNA Academy. (2018). *Routing Information Protocol (RIP).* [Online]. Available from: https://www.ccnaacademy.com/2018/10/routing-information-protocol-rip.html [Accessed: 20 October 2018]

Clough, P. & Nutbrown, C. (2012). *A student’s guide to methodology.* 3rd ed. New York: Sage.

Codd, E. F. (1970). A relational model for data for large shared data banks. *Communications of the Association for Computing Machinery (ACM*), 13(6):377-387.

Coleman, G. (n.d.). *9 quotes that sum up the Fourth Industrial Revolution*. World Economic Forum. [Online]. Available from: https://www.weforum.org/agenda/2016/01/9-quotes-that-sum-up-the-fourth-industrial-revolution/ [Accessed: 21 May 2020]

Comte, A. (1830-1842). *The positive philosophy of Auguste Comte*. Course of positive philosophy.

Council for Scientific & Industrial Research. (2016). *A South African perspective on the 2016 Global Carbon Budget*. [Online]. Available from: https://www.csir.co.za/south-african-perspective-2016-global-carbon-budget [Accessed: 3 June 2017]

Cox, J., Jennings, B. & Krukowski, S. (2014). *Reinforcement learning for adaptive routing of autonomous vehicles in congested networks*. [Online]. Available from: http://cs229.stanford.edu/proj2013/CoxJenningsKrukowski [Accessed: 15 February 2017]

CreateASoft Inc. (2018). *An analysis of simulation software: Static simulation tools vs. dynamic simulation tools.* CreateASoft Inc. Naperville, Illinois. [Online]. Available from: www.tableau.com/Data-Analysis [Accessed: 18 June 2018]

Creswell, J. W. (2009). *Research design: Qualitative, quantitative, and mixed methods approaches*. 3rd ed. Los Angeles: Sage.

Curtis, N. & Taylor, P. J. (2004). *Network and certification.* 4th ed. NY: Element K.

Daugherty, P., Negm, W., Banerjee, P. & Alter, A. E. (2015). *Driving unconventional growth through the Industrial Internet of Things.* Accenture Technology.

David, F. (2010). *3 questions to Philippe Gautier.* [Online]. Available from: http://www.i-o-t.org/post/3questionstoPhilippeGAUTIERbyDavidFayon [Accessed: 2 June 2018]

Davis, J. P., Eisenhardt, K. M. & Bingham, C. B. (2007). Developing theory through simulation methods. *Academy of Management Review,* (32)2:480-499.

Deering, S. & Hinden, R. (1998). *Internet Protocol, Version 6 (IPv6) Specification*. *RFC 2460*. [Online]. doi:https://doi.org/10.17487/RFC2460.

Delling, D. & Wagner, D. (2009). Time-dependent route planning. *In Robust and online large-scale optimisation*. pp. 207-230. https://doi.org/10.1007/978-3-642-05465-5_8.

Delport, C. S. L. (2007). *Quantitative data collection methods. In* De Vos, A. S., Strydom, A., Fouché, C. B. & Delport, C. S. L. (eds), *Research at grass roots: For the social sciences and human service professions.* 3rd ed. Pretoria: Van Schaik.

Devikar, R. N., Patil, D. V. & Chandraprakash, V. (2016). Issues in routing mechanism for packet forwarding: A survey. *International Journal of Electrical & Computer Engineering (IJECE),* 6(1):421-430.

Dias, D. (ed). (2015). *Packet switching technologies Part 1: Adapted from notes.* Department of Electronic Telecommunication Engineering, University of Moratuwa. [Online]. Available from: https://www.cisco.com/c/en/us/solutions/index.html [Accessed: 18 August 2017]

Eckelmann, S., Trautmann, T., Uβler, H., Reichelt, B. & Michler, O. (2017). V2V-communication, LiDAR system and positioning sensors for future fusion algorithms in connected vehicles. 20th Euro Working Group on Transportation Meeting (EWGT 2017), Budapest Hungary, 4-6 Sept. *Transportation Research Procedia,* 27:69-76.

Edgeworth, B., Foss, A. & Rios, R. G. (2015*). IP routing on Cisco IOS, IOS XE, and IOS XR: An essential guide to understanding and implementing IP routing protocols.* Part of the Networking Technology series. Cisco Press.

Elliot. I. (2018). *Ethics review boards and AI self-driving cars*. [Online]. Available from: https://www.aitrends.com/ai-insider/ethics-review-boards-and-ai-self-driving-cars/ [Accessed: 13 February 2020]

European Transport Safety Council. (2016). *Prioritising the safety potential of automated driving in Europe*. [Online]. Available from: https://etsc.eu/automated-driving-report/ [Accessed: 18 December 2018]

Fawaz, A. & Artail, H. (2010). Enhanced cooperative collision avoidance in sudden vehicle braking scenarios. *Proceedings*. The 2010 17th International Conference on Telecommunications, Doha, 4-7 April. pp. 806-813.

Fernandes. P. & Urbano. N. (2012). Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow. *Intelligent Transportation Systems,* 13(1):91-106. IEEE.

Fiore, M., Härri, J., Filali, F. & Bonnet, C. (2007). Vehicular mobility simulation for VANETs. *Proceedings.* 40th Annual Simulation Symposium (ANSS'07), Norfolk, VA, USA, 26-28 March. IEEE. doi:10.1109/ANSS.2007.44.

Fiske, O. J. (2002). *The Magtube low cost Maglev transportation system.* LaunchPoint Technologies, Inc., Goleta, California, USA. US Patent No. 6374746, Washington, DC, US Patent and Trademark Office.

Flood, R. & Wilson, R. (2012). *The great mathematicians.* London: Arcturus.

Fouché, C. & Schurink, W. (2011). *Qualitative research designs. In* De Vos, A. S., Strydom, H., Fouché, C. B. & Delport, C. S. L. (eds), *Research at grass roots, for the social sciences and human service professions*. Pretoria: Van Schaik. pp. 307-327.

Garmin. (n.d.). *What is GPS?* [Online]. Available from: https://www.garmin.com/en-US/aboutGPS/ [Accessed: 13 May 2020]

Geeks for Geeks. (2018). *Distance vector routing v/s Link state routing.* [Online]. Available from: https://www.geeksforgeeks.org/computer-network-distance-vector-routing-vs-link-state-routing/ [Accessed: 26 February 2018]

Giarratana, C. (2016). *Driving vehicle automation forward.* Safety Resource Centre. [Online]. Available from: https://www.trafficsafetystore.com/blog/autonomous-cars-environmental-impact/ [Accessed: 3 October 2017]

GitHub Inc. (2008). *Psimulator2 system*. [Online]. Available from: https://github.com/rkuebert/psimulator [Accessed: 13 May 2020]

Given, L. M. (2008). *The Sage Encyclopaedia of qualitative research methods*. Los Angeles, Calif.: Sage.

Global Carbon Project. (2016). *Homepage*. [Online]. Available from: http://www.globalcarbonproject.org/ [Accessed: 30 May 2017]

Goldcorp. (2018). *Our purpose is to create value and improve lives through sustainable and responsible mining*. [Online]. Available from: https://www.newmont.com/about-us/strategy/default.aspx [Accessed: 28 December 2018]

Google. (2014). *Just press go: Designing a self-driving vehicle*. [Online]. Available from: http://googleblog.blogspot.com/2014/05/just-press-go-designing-self-driving.html [Accessed: 17 December 2016]

Google News Archive. (1926). *Phantom Auto will tour city*. The Milwaukee Sentinel.

Greenemeier, L. (2016). *Deadly Tesla crash exposes confusion over automated driving: Amid a federal investigation, ignorance of the technology's limitations comes into focus*. [Online]. Available from: https://www.scientificamerican.com/article/deadly-tesla-crash-exposes-confusion-over-automated-driving/ [Accessed: 20 July 2016]

Greenough, J. (2015). *The Enterprise Internet of Things market. The corporate 'Internet of Things' will encompass more devices than the smartphone and tablet markets combined.* Business Insider. [Online]. Available from: http//www.businessinsider.com/the-enterprise-internet-of-things-market-2014-12 [Accessed: 11 November 2017]

Gregor, S. (2006). The nature and theory of information systems. *MIS Quarterly*, 3(30):611-642.

Gregor, S. & Jones, D. (2007). The anatomy of a design theory. *Journal of the Association for Information Systems,* 8(5):312-335.

Grove, S. K., Burns, N. & Gray, J. R. (2013). *The practice of nursing research appraisal, synthesis and generation of evidence*. St Louis, MO: Elsevier Saunders.

Guergiuev, J., Lillicrap, T. P. & Richards, B. A. (2017). *Towards deep learning with segregated dendrites*. Link Life. [Online]. Available from: http://linclab.org/deep-learning-with-segregated-dendrites/ [Accessed: 13 May 2017]

Guillemin, P. & Friess, P. (2011). Internet of Things strategic research roadmap. In *Internet of Things – Global technological and societal trends from smart environments and spaces to green ICT.* Gistrup, Denmark: River.

Gulyȧs, L. & Kampis, G. (2015). Models, representations and comparisons in computer simulations. *Procedia Computer Science,* 66:5-12. https://doi.org/10.1016/j.procs.2015.11.003.

Gupta, N., Prakash, A. & Tripathi, R. (2017). Adaptive beaconing in mobility aware clustering based MAC protocol for safety message dissemination in VANET. *Wireless Communications & Mobile Computing,* Art. ID 1246172. Open Access. https://doi.org/10.1155/2017/1246172.

Guseva, I. (2009). *Bad economy is good for open source.* [Online]. Available from: http://www.cmswire.com/cms/web-cms/bad-economy-is-good-for-open-source-004187.php [Accessed: 25 August 2018]

Halba, K., Mahmoudi, C. & Griffor, E. (2018). Robust safety for autonomous vehicles through reconfigurable networking. *Electronic Proceedings in Theoretical Computer Science (EPTCS),* (269):48-58. doi:10.4204/EPTCS.269.5.

Hamid, R. A. (2017). MIST: Fog-based data analytics scheme with cost-efficient resource provisioning for IoT crowd sensing applications. *Journal of Network and Computer Applications.* 82(15):152-165.

Hanseth, O. & Lyytinen, K. (2010). Design theory for dynamic complexity in information infrastructures: The case of building internet. *Journal of Information Technology*, 25:1-19.

Hanley, S. (2016). *Volvo autonomous car engineer calls Tesla's Autopilot a "wannabe".* TESLARATI. [Online]. Available from: https://www.teslarati.com/volvo-engineer-calls-tesla-autopilot-wannabe/ [Accessed: 22 May 2019]

Hart, J. K. & Martinez, K. (2015). Toward an environmental Internet of Things. *Earth & Space Science,* (2):194-200.

Hevner, A. R. & Gregor, S. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly,* 37(2):337-355.

Hevner, A. R., March, S. T., Park, J. & Ram, S. (2004). Design science in information systems research. *MIS Quarterly,* 28(1):75-105.

Hofstee, E. (2009). *Constructing a good dissertation: A practical guide to finishing a Master's MBA or PhD on schedule.* Sandton: EPE.

Horberry, T., Anderson, J., Regan, M. A., Triggs, T. J. & Brown, J. (2006). Driver distraction: The effects of concurrent in-vehicle tasks, road environment and complexity and age on driving performance. *Accident Analysis and Prevention,* 38(1):185-191.

Hornick, M., Marcadė, E. & Venkayala, S. (2006). *Java data mining: Strategy standard and practice: A practical guide for architecture, design and implementation* 1st ed. U.S.A.: Morgan Kaufman (Imprint).

Howard, C. E. (2011). *Optical technology at the speed of light*. Military & Aerospace. [Online]. Available from: http://www.militaryaerospace.com/articles/print/volume-22/issue-4/technology-focus/optical-technology-at-the-speed-of-light.html [Accessed: 2 October 2017]

Hsu, T. (2016). *75% of U.S. drivers fear self-driving cars, but it's an easy fear to get over*. IIEEE Spectrum. [Online]. Available from: http://spectrum.ieee.org/cars-that-think/transportaion/self-driving/driverless-cars-inspire-both-fear-and-hope http://ieeexplore.ieee.org/Xplore/hoLibrary [Accessed: 23 May 2017]

Huang, D., Su, C. & Yan, Y. (2017). Traffic-aware opportunistic data delivery strategy for urban vehicular Ad Hoc networks. *International Journal of Performability Engineering*, 13(4):437-445.

Huddleston Jr., T. (2019). *These Chinese hackers tricked Tesla's Autopilot into suddenly switching lanes*. [Online]. Available from: https://www.cnbc.com/2019/04/03/chinese-hackers-tricked-teslas-autopilot-into-switching-lanes.html [Accessed: 7 May 2019]

Hurwitz, J. & Kirsch, D. (2018). *Machine learning for dummies: IBM limited edition.* NJ: Wiley & Sons.

Hush, D. & Horne, B. (1993). Progress in supervised neural networks. *Signal Processing Magazine,* 10(1):8-39. IEEE.

Institute of Electrical and Electronics Engineers. (2017). *IEEE search engine*. [Online]. Available from: http://ieeexplore.ieee.org/Xplore/hoLibrary [Accessed: 21 May 2017]

Institute of Electrical and Electronics Engineers. (2018). IEEE 802.11p contention for overhead-free spectrum sensing in CR-VANETs. *Proceedings*. 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, 12-15 Jan. doi:10.1109/CCNC.2018.8319225.

International Transport Forum. (2017). *Road Safety Annual Report 2017* [Online]. Available from: https://www.oecd-ilibrary.org/transport/road-safety-annual-report-2017_irtad-2017-en [Accessed: 13 May 2020]

ITU. (2020). *Internet of Things global standards initiative.* [Online]. Available from: https://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx [Accessed: 26 June 2018]

Isermann, R. & Balla, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, 5(5):709-719.

Jansen, J. D. (2012). The quality of doctoral education in South Africa: A question of significance. *In* Maree, K. (ed.), *Complete your thesis or dissertation successfully: Practical guidelines*. Landsdowne: Juta. pp. 1-11.

Jansen, P. L., Li, L. & Lorenz, R. D. (1995). Analysis of competing topologies of linear induction machines for high-speed material transport systems*. IEEE Transactions on Industry Applications,* 31(4):925. Jul/Aug.

Jayavardhana, G., Rajkumar, B., Slaven, M. & Marimuthu, P. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems,* 29(7):16451660.

Joy, M. (2014). *What's it like in a driverless car?* [Online]. Available from: recombu.com/cars/article/what-s-it-like-in-a-driverless-car/ [Accessed: 16 May 2017]

Kampis, G. (2013). *Self-modifying systems in biology and cognitive science: A new framework for dynamics, information and complexity.* Volume 2. Amsterdam: Elsevier Science.

Kasim, R., Alexander, K. & Hudson, J. (2010). *A choice of research strategy for identifying community-based action skill requirements in the process of delivering housing market renewal*. Research institute for the Built and Human Environment, University of Salford, UK.

Kenny, J. (2018). *Human brain' supercomputer with 1 million processors switched on for first time.* University of Manchester. Tech Xplore. [Online]. Available from: https://techxplore.com/news/2018-11-human-brain-supercomputer-million-processors.html [Accessed: 4 March 2019]

Khayyam, H., Javadi, B., Jalili, M & Jazar, R. (2020). Artificial intelligence and Internet of Things for autonomous vehicles. *In*: *Nonlinear approaches in engineering applications*. Springer International. pp. 39-68.

Kimmelman, M. (2012). *Paved, but still alive.* The New York Times. [Online]. Available from: http://www.nytimes.com/2012/01/08/arts/design/taking-parking-lots-seriously-as-public-spaces.html [Accessed: 30 May 2017]

Kohavin R. & Provost, F. (eds). (1998). Glossary of terms. *Machine Learning,* (30):271-274.

Kothari, C. R. (2004). *Research methodology: Methods and techniques.* 2nd revised ed. New Delhi: New Age International.

Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J. & Bhatia, M. (2012). *A uniform format for IPv6 extension headers. RFC 6564 (proposed standard).* [Online]. Available from: ftp://ftp.dc.aleron.net/pub/rfc/pdf/rfc6564.txt.pdf [Accessed: 15 June 2017]

Kuhn, T. S. (1970). *The structure of scientific revolutions.* 2nd ed. Chicago: University of Chicago Press.

Kulbytē, T. (2020). *37 customer experience statistics you need to know for 2020.* [Online]. Available from: https://www.superoffice.com/blog/customer-experience-statistics/ [Accessed: 27 June 2020]

Lanctot, R. (2017). *Accelerating the future: The economic impact of the emerging passenger economy.* [Online]. Available from: https://newsroom.intel.com/newsroom/wp-content/uploads/sites/11/2017/05/passenger-economy.pdf [Accessed: 22 January 2021]

Laszlo, A. & Krippner, S. (1998). Systems theories: Their origins, foundations and development. *In* Jordan, J. S. (ed.), *Systems theories and a priori aspects of perception*. Amsterdam: Elsevier Science. pp. 47-74.

Law, A. M. & Kelton, D. (1991). *Simulation modelling and analysis.* 2nd ed. New York, NY: McGraw-Hill.

Lee, J., Bagheri, B. & Hung-An, K. (2015). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, (3):18-23. https://doi.org/10.1016/j.mfglet.2014.12.001.

Le Quéré, C., Andrew, R. M., Canadell, J. G., Sitch, S., Korsbakken, J. I., Peters, G. P., Manning, A. C., Boden, T. A. et al. (2016). Global Carbon Budget 2016. *Earth System Science Data*, 8:605-649. https://doi.org/10.5194/essd-8-605-2016.

Lexico.com. (2020). *Prepend*. [Online]. Available from: https://www.lexico.com/en/definition/prepend [Accessed: 12 May 2020]

Liang, L., Peng, H., Ye Li, G. & Shen, X. (2017). vehicular communications: a physical layer perspective. *IEEE Transactions on Vehicular Technology*, 66(12):10647-10659. doi:10.1109/TVT.2017.2750903.

Lilienfeld, R. (1978). *The rise of systems theory: An ideological analysis.* New York: Wiley.

Lindner, T. (2015). *The supply chain: Changing at the speed of technology.* Connected World*.* [Online]. Available from: http://connectedworld.com/the-supply-chain-changing-at-the-speed-of-technology/ [Accessed: 11 November 2017]

Linkletter, B. (2016). *Psimulator2 network project.* Czech Technical University, Prague, Faculty of Information Technology. [Online]. Available from: https://github.com/rkuebert/psimulator [Accessed: 13 May 2020]

Lissy, K. S., Cohen, J. T., Park, M. Y. & Graham, J. D. (2000). *Cellular phone use while driving: Risks and benefits: Phase 1 report.* Harvard Centre for Risk Analysis at Harvard School of Public Health, Boston, Massachusetts.

Litman, T. A. (2020). *Autonomous vehicle implementation predictions: Implications for transport planning.* Victoria Transport Policy Institute.

Liu, J., Wan, J. & Wang, Q. (2016). A survey on position based routing for vehicular ad hoc networks. *Journal of Telecommunication System Architecture,* (62):15-30.

March, S. T. & Smith, G. (1995). Design and natural science research on information technology. *Decision Support Systems,* 15(4):251-266.

Marinik, A., Bishop, R., Fitchett, V., Morgan, J. F., Trimble, T. E. & Blanco, M. (2014). *Human factors evaluation of Level 2 and Level 3 automated driving concepts: Concepts of operation*. (Report No. DOT HS 812 044). Washington, DC: National Highway Traffic Safety Administration.

Mathews, S. A. (2020). When rubber meets the road: Balancing innovation and public safety in the regulation of self-driving cars. *Boston College Law Review*, 61(1):295-338.

McCarthy, J. & Feigenbaum, E. (1990). In memoriam: Arthur Samuel: Pioneer in machine Learning. *AI Magazine,* 11(3):10. doi:https://doi.org/10.1609/aimag.v11i3.840.

McClelland, C. (2020). *The impact of artificial intelligence – widespread job losses*. [Online]. Available from: https://www.iotforall.com/impact-of-artificial-intelligence-job-losses/ [Accessed: 12 May 2020]

McClelland, C. & Feigenbaum, E. (2017). *The difference between artificial intelligence, machine learning and deep learning.* [Online]. Available from: https://medium.com/iotforall/the-difference-between-artificial-intelligence-machine-learning-and-deep-learning-3aa67bff5991 [Accessed: 24 December 2017]

Mehdipour, F., Javadi, B., Mahanti, A. & Ramirez-Prado, G. (2019). Fog computing realisation for big data analytics. *In: Fog and edge computing: Principles and paradigms*. New York: Wiley. pp. 259-290.

Merriam-Webster. (2020). *Algorithm*. [Online]. Available from: https://www.merriam-webster.com/dictionary/algorithm [Accessed: 13 May 2020]

Mervis, J. (2017). *Are we going too fast on driverless cars?* Science Magazine. [Online]. Available from: www.sciencemag.org/news/2017/12/are-we-going-too-fast-driverless-cars [Accessed: 24 September 2018]

Metz, D. (2018). Developing policy for urban autonomous vehicles: Impact on congestion. *Urban Science,* 2(2):33.

Mois, G., Folea, S. & Sanislav, T. (2017). Analysis of three IoT-based wireless sensors for environmental monitoring. *IEEE Transactions on Instrumentation & Measurement,* (66):2056-2064.

Monette, D. R., Sullivan, T. J. & De Jong, C. R. (2002). *Applied social research: Tool for the human services.* 5th ed. Toronto: Wadsworth Thomson Learning.

Morgan Stanley Research. (2018). *Techs next big wave: Manufacturing.* [Online]. Available from: https://www.morganstanley.com/ideas/manufacturing-tech-wave [Accessed: 6 February 2020]

Mouton, J. (1996). *Understanding social research.* Pretoria: Van Schaik.

Mulvenna, M., Hutton, A., Martin, S., Todd, S., Bond, R. & Moorhead, A. (2017). Views of caregivers on the ethics of assistive technology used for home surveillance of people living with dementia. *Neuroethics,* 10(2):255-266.

Muoio, D. (2016). *Self-driving cars are prone to hacks – and automakers are barely talking about it.* [Online]. Available from: http://www.businessinsider.com/driverless-cars-hacking-ricks-2016-12 [Accessed: 2 October 2017]

Myers, M. D. (2009). *Qualitative research in business & management*. London: Sage.

National Highway Traffic Safety Administration. (2013). *Vehicle Policy 4. Preliminary statement of policy regarding automated vehicles*. Washington DC.

National Highway Traffic Safety Administration. (2014). *Traffic safety facts research note: Distracted driving.* Washington DC.

Nel, F. (2019). *South Africa's road death statistics are appalling: Here's a way to bring them down*. Daily Maverick. [Online]. Available from: https://www.dailymaverick.co.za/opinionista/2019-12-11-south-africas-road-death-statistics-are-appalling-heres-a-way-to-bring-them-down/ [Accessed: 14 January 2020]

Nelson, B. L. (1995). *Stochastic modelling, analysis and simulation*. McGraw-Hill.

Neser, M. (2006). *Neuro-fuzzy techniques for intelligent control*. Master's Thesis, North-West University, South Africa.

Neuman, W. L. (2000). *Social research methods: Qualitative and quantitative approaches.* 4th ed. Boston: Allyn & Bacon.

Ngquakamba, S. (2019). *Drunk driving main cause of road accidents this festive season.* [Online]. Available from: https://www.news24.com/SouthAfrica/News/drunk-driving-the-main-cause-of-road-accidents-this-festive-season-20190108 [Accessed: 16 June 2019]

Norton, S. & Suppe, F. (2001). Why atmospheric modeling is good science*. In* Mille, C. & Edwards, P. (eds.), *Changing the atmosphere: Expert knowledge and environmental governance.* Cambridge, MA: MIT Press. pp. 88-133.

Nunamaker, J., Chen, M. & Purdin, T. D. M. (1991). Systems development in information systems research. *Journal of Management Information Systems,* 7(3):89-106.

Nutaro, J. J. (2010). *Building software for simulation: Theory and algorithms, with applications in C++.* Hoboken: John Wiley & Sons.

Oberkampf, W. L. & Roy, C. J. (2010). *Verification and validation in scientific computing*. Cambridge University Press. https://doi.org/10.1017/CBO9780511760396.

O'Brien, J.A. & Marakas, G. M. (2011). *Management information systems*. New York: McGraw-Hill/Irwin.

Olivier, M. S. (2011). *Information technology research: A practical guide for computer science and informatics*. 3rd ed. Pretoria: Van Schaik.

Open Computing Facility. (2021). *Key flaw of the Turing Test.* [Online]. Available from: https://www.ocf.berkeley.edu/~arihuang/academic/research/turingflaw.html [Accessed: 22 January 2021]

Osterloh, H. (2001). *IP Routing Primer Plus.* Indianapolis, Indiana: Sams.

Östman, L. E. (2005). *Design theory is a philosophical discipline – reframing the epistemological issues in design theory*. PhD Dissertation, Royal Institute of Technology, Stockholm.

O'Toole, R. (2009). *Gridlock: Why we're stuck in traffic and what to do about it.* Washington D.C.: Cato Institute. pp. 189-192.

Padayachi, Y. R. & Thambiran, T. (2012). Geared for change? CO^2 emissions from South Africa's road transport sector. *Proceedings.* The Annual Conference on National Association for Clean Air, Rustenburg, 31 October-2 November. pp. 1-4.

Panetta, K. (2018). *Widespread artificial intelligence, biohacking, new platforms and immersive experiences dominate this year's Gartner Hype Cycle.* [Online]. Available from: https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/ [Accessed: 21 November 2018]

Parker, W. (2015). Computer simulation*. In* Psillos, S. & Curd, M. (eds*), The Routledge Companion to Philosophy of Science.* 2nd ed. London: Routledge.

Pawsey, C. (2017). *Autonomous driving: Beacon projects in Europe.* [Online]. Available from: https://www.automotive-iq.com/chassis-systems/articles/autonomous-driving-beacon-projects-europe [Accessed: 13 May 2020]

Peffers, K., Tuunanen, T., Rothenberger, M. A. & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems,* 24(3):45-77.

Peissener, M., Doebler, V. & Metze, F. (2011). *Can voice interaction help reducing the level of distraction and prevent accidents?* Meta study on driver distraction and voice interaction. White Paper. Germany: Nuance Communications.

Pettey, C. (2017). *CIOs and IT leaders will drive significant competitive advantages with this integration.* [Online]. Available from: https://www.gartner.com/smarterwithgartner/when-it-and-operational-technology-converge/ [Accessed: 13 December 2018]

Pidd, M. (1998). *Computer simulation in management science.* 4th ed*.* New York: Wiley.

Polson, N. & Scott, J. (2018). *AIQ: How artificial intelligence works and how we can harness its power for a better world.* London: Bantam Press.

Porup, J.M. (2016). *Internet of Things security is hilariously broken and getting worse.* Ars Technica. [Online]. Available from: https://arstechnica.com/information-technology/2016/01/how-to-search-the-internet-of-things-for-photos-of-sleeping-babies/ [Accessed: 21 November 2017]

Pyper, J. (2014). *Self-driving cars could cut greenhouse gas pollution.* [Online]. Available from: https://www.scientificamerican.com/article/self-driving-cars-could-cut-greenhouse-gas-pollution/ [Accessed: 5 June 2017]

Rablau, C. (2019). LiDAR: A new self-driving vehicle for introducing optics to broader engineering and non-engineering audiences. *Proceedings.* The Fifteenth Conference on Education and Training in Optics and Photonics (ETOP 2019), 2 July. doi:10.1117/12.2523863.

Raji, R. S. (1994). Smart networks for control. *IEEE Spectrum,* 31(6):49-55.

Rashdan, I., De Pont Müller, F. & Sand, S. (2017). *ITS-G5 challenges and 5G solutions for vehicular platooning.* DLR project Next Generation Car, Fahrzeugintelligenz und Mechatronisches Fahrwerk (FiF). German Aerospace Centre (DLR), Institute of Communications & Navigation, Germany.

Regan, P. (2013). *Administering Windows Server 2012*. Hoboken: John Wiley & Sons.

Repenning, N. (2002). A simulation-based approach to understanding the dynamics of innovation implementation. *Organisation Science,* 13(2):109-127.

Reuter, A. & Cordeiro, E. (2016). Source Packet Routing in Networking (SPRING). *Seminar*. Future Internet WS 2016, University of Munich, 1 May. pp. 31-37. [Online]. Available from: https://pdfs.semanticscholar.org/d42a/6b5961b0078a0cd6195496262943f932fab6.pdf [Accessed: 23 April 2017]

Road Traffic Management Corporation. (2020). Homepage. [Online]. Available from: https://www.rtmc.co.za/index.php/publications/press-releases/2020 [Accessed: 13 May 2020]

Robinson, S. (2004). *Simulation – the practice of model development and use.* New York: Wiley.

Roedolf, G. & Voelzke, U. (2017). *Three sensor types drive autonomous vehicles.* Automotive & Discretes Group (ADAS). FierceElectronics. [Online]. Available from: https://www.fierceelectronics.com/components/three-sensor-types-drive-autonomous-vehicles [Accessed: 2 January 2018]

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organisation in the brain. *Psychological Review*, 65(6):386-408.

Roy, S. (2005). Recent advances in numerical methods for fluid dynamics and heat transfer. *Journal of Fluid Engineering,* 127(4):629-630.

Sanders, C. (2007). *Practical packet analysis: Using Wireshark to solve real-world network problems*. California: No Starch Press.

Santucci, G. (2011). *The Internet of Things between the revolution of the Internet and the metamorphosis of objects.* European Commission Community Research and Development Information Service.

Sargent, R. G. (2005). Verification and validation of simulation models. *Proceedings.* 37th Conference on Winter Simulation, Orlando, FL, USA, 4-7 Dec. pp. 130-143.

Sastry, M. A. (1997). Problems and paradoxes in a model of punctuated organisational change. *Administrative Science Quarterly,* 42(2):237-275.

Satyanarayanan, M. (2017). The emergence of edge computing. *Computer,* (50):30-39.

Saunders, M. N. K., Thornhill, A. & Lewis, P. (2009). *Research methods for business students.* 5th ed. UK: Pearson.

Schippers, R. & Socash, J. (2017). *Pedestrian and non-motor detection systems with LIDAR for autonomous vehicles*. Paper presented at University of Pittsburgh Swanson School of Engineering. Semantic Scholar, Corpus ID: 45277423.

Sheldon, T. (2001). *McGraw-Hill's Encyclopedia of Networking and Telecommunication (Network Professional's Library)*. McGraw-Hill.

Sherazi, H. H. R., Kahn, Z. A., Iqbal, R., Rizwan, S., Imran, M. A. & Awan, K. (2019). A heterogeneous IoV architecture for data forwarding in vehicle to infrastructure communication. *Mobile Information Systems,* Article ID 3101276. https://doi.org/10.1155/2019/3101276.

Sherif, A. B. T. (2017). Privacy-preserving ride sharing scheme for autonomous vehicles in big data era. *IEEE Internet of Things Journal*, (4):611-618.

Simon, H. (1996). *The Sciences of the artificial.* 3rd ed. Cambridge, MA: MIT Press.

Simonite, T. (2016). *Tesla's first autonomous vehicle may arrive before the Model 3.* [Online]. Available from: https://www.technologyreview.com/2016/05/26/159920/teslas-first-autonomous-vehicle-may-arrive-before-the-model-3/ [Accessed: 13 May 2020]

Soare, R.I. (2009). *Turing oracle machines, online computing, and three displacements in Computability Theory.* Department of Mathematics, University of Chicago, Chicago, Illinois.

Society of Automotive Engineers International. (2018). *Levels of automation.* [Online]. Available from: www.sae.org/news/press-room/2018/12/sae-international-releases-updated-visual-chart-for-its-“levels-of-driving-automation”-standard-for-self-driving-vehicles. [Accessed: 11 May 2018]

South African Department of Environmental Affairs. (2013). *2013/2014 Annual Report*. [Online]. Available from: https://www.gov.za/documents/department-environmental-affairs-annual-report-20132014 [Accessed: 23 July 2017]

South African Road Safety Audit. (2012). *New roads and schemes: Road safety audit manual.* pp. III-IV.

St. Laurent, A. M. (2008). *Understanding open source and free software licensing*. O'Reilly Media.

Standage, T. (2018). *What's it like to ride in a self-driving car*? [Online]. Available from: https://medium.economist.com/whats-it-like-to-ride-in-a-self-driving-car-24015c69cc48 [Accessed: 12 May 2020]

Stokes, P. (2012). *Philosophy 100 essential thinkers: The ideas that have shaped our world*. London: Arcturus.

Stringfield, N., White, R & McKee, S. (2007). *Cisco express forwarding: Understanding and troubleshooting CEF in Cisco routers and switches*. US: Cisco Press.

Strogatz, S. (2007). The end of insight. *In* Brockman, J. (ed.), *What is your dangerous idea?* HarperCollins. ISBN: 9780061214950.

Taylor, D. (2016). *South African road accidents on rise*. [Online]. Available from: https://www.voanews.com/a/south-african-road-accidents-on-rise/3305148.html [Accessed: 2 June 2017]

Templeton, B. (2015). *Traffic congestion & capacity.* [Online]. Available from: https://www.templetons.com/brad/robocars/congestion.html [Accessed: 13 May 2020]

Thompson, K. & Mattalo, B. (2015). *The internet of Things: Guidance, regulation and the Canadian approach.* CyberLex. [Online]. Available from: https://www.mccarthy.ca/en/insights/blogs/snipits/internet-things-guidance-regulation-and-canadian-approach [Accessed: 13 May 2020]

Thrun, S. (2011). *Google's driverless car*. TED talk. [Online]. Available at: https://www.ted.com/talks/sebastian_thrun_google_s_driverless_car/transcript?language=en [Accessed: 13 May 2020]

Tomovic, S., Prasad, N. & Radusinovic, I. (2014). SDN control framework for QoS provisioning. *In* 22nd *Telecommunications Forum* (TELFOR), Belgrade. pp. 111–114. doi:10.1109/TELFOR.2014.7034369.

Triplett, T., Santos, R., Rosenbloom, S. & Tefft, B. (2014). *American driving survey 2014/2015.* AAA Foundation for Traffic Safety.

TRL. (2016). *Heavy Vehicle Platoons on UK Roads: Case study.* [Online]. Available from: https://trl.co.uk/heavy-vehicle-platoons-uk-roads [Accessed: 12 May 2020]

Trochim, W. M. K. & Donnely, J. P. (2008). *The research methods knowledge base*. 3rd ed. United States: Cengage Learning.

Tsiu, J. (2020). *How will self-driving vehicles impact the environment?* [Online]. Available from: https://www.greenjournal.co.uk/2020/01/how-will-self-driving-vehicles-impact-the-environment/ [Accessed: 12 May 2020]

Tuohy, S., Glavin, M., Hughes, C., Jones, E., Trivedi, M. & Kilmartin, L. (2015). Intravehicle networks: A review. *IEEE Transactions on Intelligent Transportation Systems,* 16(2):534-545.

Turing, A. M. (1939). *Systems of logic based on ordinals*. PhD Thesis, Princeton University.

Turing, A. M. (1950). Computing machinery and intelligence: A quarterly review of psychology and philosophy*. Mind*, 59(236):433-460.

UD Trucks. (2013). *Rail expected to have increasing impact on road freight industry*. [Online]. Available from: https://www.forestry.co.za/rail-expected-to-have-increasing-impact-on-road-freight-industry/ [Accessed: 2 Jul 2017]

Ulgen, O. M., Black, J. J., Johnsonbaugh, B. & Klungle, R. (2014). *Simulation methodology – A practitioner's perspective*. Paper presented by the Michigan Simulation User Group Technical Committee on Simulation Methodology. University of Michigan-Dearborn and Production Modeling Corporation, Dearborn, MI.

United Nations Educational and Scientific Council. ((2010). *Report of the International Bioethics Committee (IBC) on social responsibility and health.* UNESCO. [Online]. Available from: https://books.google.co.za/books?id=a_nFBIF_a4EC&printsec=frontcover&dq=inauthor:%22International+Bioethics+Committee%22&hl=en&sa=X&ved=2ahUKEwiTzoTLjNLqAhX9ThUIHXl1B60Q6AEwAHoECAEQAg#v=onepage&q&f=false [Accessed: 2 August 2017]

United Nations Framework Convention on Climate Change. (2010). *Paris Agreement – Status of ratification.* [Online]. Available from: http://unfccc.int/2860.php [Accessed: 26 June 2017]

USA Department of Defence. (2008). *Global positioning system standard positioning service performance standard.* 4th ed. GPS Navstar.

Volvo Cars. (2016). *Drive Me, the world's most ambitious and advanced public autonomous driving experiment, starts today*. [Online]. Available from: https://www.media.volvocars.com/global/en-gb/media/pressreleases/196240/drive-me-the-worlds-most-ambitious-and-advanced-public-autonomous-driving-experiment-starts-today [Accessed: 13 May 2020]

Von Bertalanffy, L. (1973), *General systems theory.* Revised ed. New York: GeorgeBraziller.

Wahl, R. L. (2013). *The study of packet switched technology incorporated into large-scale magnetic levitating vehicles.* MTech Dissertation, Vaal University of Technology, South Africa.

Walker. (2013). *Customers 2020: A progress report: More insight for a new decade*. [Online]. Available from: https://www.walkerinfo.com/knowledge-center/featured-research-reports/customers-2020-a-progress-report [Accessed: 30 August 2018]

Wallace, M. (2016). *Fragmentation is the enemy of the Internet of Things.* [Online]. Available from: https://www.qualcomm.com/news/onq/2016/02/19/fragmentation-enemy-internet-things [Accessed: 3 January 2017]

Walls, J. G., Widemeyer, G. R. & El Sawy, O. A. (1992). Building an Information System Design theory for Vigilant EIS. *Information Systems Research,* (3)1:36-59.

Welman, J. C., Kruger, S. J. & Mitchell, B. (2005). *Research methodology*. 3rd ed. Cape Town: Oxford University Press.

Wichmann, B. A., Canning, A. A., Clutterbuck, D. L., Winsbarrow, L. A., Ward, N. J. & Marsh, D. W. R. (1995). Industrial perspective on static analysis*. Software Engineering Journal,* 10(2):69-75. Archived from the original (PDF) on 2011-09-27.

Wiener, N. (1988). *The human use of human beings: Cybernetics and society*. Houghton Mifflin, Boston, 1950/1954; 2nd ed. New York: Da Capo.

Wigan, M. R. & Drain, P. (2002). *Open transport modelling networking for transportation applications*. (No. TLUMIP-WP-02-7). Portland USA: PB Consult for Oregon Department of Transport.

Winsberg, E. (2010). *Science in the age of computer simulation*, Chicago: The University of Chicago Press. ISBN-13: 978-0226902043. ISBN-10: 0226902048.

Wiysonge, C. S. (2020). *South Africa's War on covid-19*. [Online]. Available from: https://www.thinkglobalhealth.org/article/south-africas-war-covid-19 [Accessed: 23 January 2021]

World Health Organisation. (2013). *Global Status Report on Road Safety 2013*. [Online]. Available from: https://www.who.int/violence_injury_prevention/road_safety_status/2013/en/ [Accessed: 6 February 2020]

World Health Organisation. (2015). *Global Status Report on Road Safety 2015*. [Online]. Available from: https://www.who.int/violence_injury_prevention/road_safety_status/2015/en/ [Accessed: 6 February 2020]

World Health Organisation. (2018). Global Status Report on Road Safety 2018. [Online]. Available from: https://www.who.int/publications/i/item/9789241565684 [Accessed: 23 January 2021]

Xu, W. (2018). Internet of vehicles in big data era. *IEEE/CAA Journal of Automatica Sinica,* (5):19-35.

Yass, J. (2017). *Satellites to provide vital link to connected cars*. [Online]. Available from: http://www.intelsat.com/news/blog/satellites-to-provide-vital-link-to-connected-cars/ [Accessed: 16 September 2017]

Yu, Z. & Tsai, J. J. P. (2011). *Intrusion detection: A machine learning approach*. London: Imperial College Press.

Zacker, C. (2014). *Installing and configuring Windows Sever 2012 R2 Exam 70-410.* Hoboken: John Wiley & Sons.

Zhang, B. J. & Voas, J.(2013). Cloud Computing, *IT Professional*, 15(2):12-14.

Zhang, R., Rossi, F. & Pavone, M. (2016). *Routing autonomous vehicles in congested transportation networks: Structural properties and coordination algorithms*. arXiv:1603.00939v2. Stanford: Stanford University.

Zhuge, H. (2004). Future interconnection environment – dream, principle, challenge and practice. *Proceedings*. The 5th International Conference on Web-Age Information Management (WAIM 2004), Dalian, China, 15-17 July. *Advances in Web-Age Information Management*, pp.13-22. Part of the Lecture Notes in Computer Science book series (LNCS, volume 3129).

Zhuge, H. (2005). The future interconnection environment. *IEEE Computer,* 38(4):27-33.

Zikmund, W. G., Babin, B. J. Carr, J. C. & Griffin, M. (2010). *Business research methods.* 8th ed. South-Western: Cengage Learning.

Zott, C. (2003). Dynamic capabilities and the emergence of intra-industry differential firm performance: Insights from a simulation study. *Strategic Management Journal,* 24(2):97-125.

ANNEXURE A: CONSOLE LOGS OF ROUTER EVENTS

Route Info	Packet Info	Details
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=1	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=1	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=2	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=2	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=2	IP packet received from interface: FastEthernet0/3.

Route Info	Packet Info	Details
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=2	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=2	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=3	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=3	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=3	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=3	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=3	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=3	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=3	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=3	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=3	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=3	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=3	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=3	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=4	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=4	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=4	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=4	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=4	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1025 seq=4	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=4	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=4	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=4	Sending packet.

Route Info	Packet Info	Details
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=4	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=4	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1025 seq=4	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=1	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=1	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=1	ARP reply received, sending packet to nextHop.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=1	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=1	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=1	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=1	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=1	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=1	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=1	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=2	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=2	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=2	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=2	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=2	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=2	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=2	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=2	IP packet received from interface: FastEthernet0/0.

Route Info	Packet Info	Details
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=2	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=2	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=2	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=2	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=3	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=3	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=3	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=3	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=3	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=3	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=3	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=3	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=3	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=3	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=3	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=3	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=4	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=4	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=4	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=4	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=4	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=4	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=4	Sending packet.

Route Info	Packet Info	Details
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=4	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=4	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=4	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=4	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=4	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=5	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=5	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=5	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=5	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=5	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=5	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=5	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=5	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=5	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=5	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=5	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=5	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=6	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=6	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=6	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=6	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=6	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=6	Received IP packet destined to be mine.

Route Info	Packet Info	Details
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=6	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=6	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=6	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=6	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=6	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=6	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=7	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=7	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=7	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=7	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=7	Sending packet.
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=7	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=7	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=7	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=7	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=7	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=7	Sending packet.
[INFO] NET: home8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=7	Received IP packet destined to be mine.
[INFO] NET: home8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=8	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=8	IP packet received from interface: FastEthernet0/2.
[INFO] NET: Router5 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=8	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=8	IP packet received from interface: FastEthernet0/1.
[INFO] NET: Router8 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=8	Sending packet.

Route Info	Packet Info	Details
[INFO] NET: home1 IPLayer: IpPacket: src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 IPv4	IcmpPacket: REQUEST ZERO id: 1026 seq=8	Received IP packet destined to be mine.
[INFO] NET: home1 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=8	Sending packet.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 64 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=8	IP packet received from interface: FastEthernet0/0.
[INFO] NET: Router8 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=8	Sending packet.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 63 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=8	IP packet received from interface: FastEthernet0/3.
[INFO] NET: Router5 IPLayer: IpPacket: src: 7.0.0.2 dst: 4.0.0.4 ttl: 62 IPv4	IcmpPacket: REPLY ZERO id: 1026 seq=8	Sending packet.

ANNEXURE B: COMPREHENSIVE LOGS OF CAPTURED EVENTS

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
133	95	=== Ethernet === size: 52 src: f9:c8:d5:68:47:e5 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: f9:c8:d5 68:47:e5 4.0.0.4 C4target: 00:00:00:00:00:00 4.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	74	652
131	72	=== Ethernet === size: 52 src: f9:c8:d5:68:47:e5 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: f9:c8:d5:68:47:e5 4.0.0.4 target: 00:00:00:00:00:00 4.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	95	655
106	80	=== Ethernet === size: 52 src: f9:c8:d5:68:47:e5 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: f9:c8:d5:68:47:e5 4.0.0.4 C5target: 00:00:00:00:00:00 4.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	95	655
132	76	=== Ethernet === size: 52 src: f9:c8:d5:68:47:e5 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: f9:c8:d5:68:47:e5 4.0.0.4 target: 00:00:00:00:00:00 4.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	95	656
106	95	=== Ethernet === size: 52 src: 44:0e:7c:1c:fa:6d dst: f9:c8:d5:68:47:e5 === ARP ARP_REPLY === sender: 44:0e:7c:1c:fa:6d 4.0.0.1 target: f9:c8:d5:68:47:e5 4.0.0.4	SUCCESSFULLY_ TRANSMITTED	ARP	80	658
133	74	=== Ethernet === size: 52 src: 44:0e:7c:1c:fa:6d dst: f9:c8:d5:68:47:e5 === ARP ARP_REPLY === sender: 44:0e:7c:1c:fa:6d 4.0.0.1 target: f9:c8:d5:68:47:e5 4.0.0.4	SUCCESSFULLY_ TRANSMITTED	ARP	95	659
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	74	666
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	95	668

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
125	118	=== Ethernet === size: 52 src: 53:f4:72:6a:4d:9e dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 53:f4:72:6a:4d:9e 5.0.0.1 target: 00:00:00:00:00:00 5.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	80	678
125	80	=== Ethernet === size: 52 src: fd:9b:74:b9:5f:d6 dst: 53:f4:72:6a:4d:9e === ARP ARP_REPLY === sender: fd:9b:74:b9:5f:d6 5.0.0.2 target: 53:f4:72:6a:4d:9e 5.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	118	692
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	698
128	108	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	118	702
134	28	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	108	704
135	30	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	108	705
136	34	=== Ethernet === size: 52 src: a3:db:32:49:14:19 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: a3:db:32:49:14:19 7.0.0.1 target: 00:00:00:00:00:00 7.0.0.2	SUCCESSFULLY_ TRANSMITTED	ARP	108	705
134	108	=== Ethernet === size: 52 src: 49:07:cb:76:58:bb dst: a3:db:32:49:14:19 === ARP ARP_REPLY === sender: 49:07:cb:76:58:bb 7.0.0.2 target: a3:db:32:49:14:19 7.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	28	707

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
128	118	=== Ethernet === size: 52 src: 49:07:cb:76:58:bb dst: a3:db:32:49:14:19 === ARP ARP_REPLY === sender: 49:07:cb:76:58:bb 7.0.0.2 target: a3:db:32:49:14:19 7.0.0.1	SUCCESSFULLY_ TRANSMITTED	ARP	108	708
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	118	711
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 1 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	108	712
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	74	1660
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	95	1663
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	1675
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	118	1682
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 2 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	108	1684

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 3 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	74	2661
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 3 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	95	2663
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 3 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	80	2675
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 3 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	118	2681
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 3 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	108	2682
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 4 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	74	3656
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 4 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	95	3658
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 4 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	80	3662

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 4 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	118	3668
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 4 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	108	3669
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 5 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	74	4662
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 5 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	95	4664
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 5 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	80	4674
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 5 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	118	4677
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO B41id: 1026 seq: 5 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	108	4678
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 6 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	74	5663

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 6 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	95	5664
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 6 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	5672
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 6 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	118	5675
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO C46id: 1026 seq: 6 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	108	5676
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 7 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	74	6663
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 7 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	95	6664
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 7 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	80	6672
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 7 payloadSize: 56	SUCCESSFULLY_ TRANSMITTED	ICMP	118	6680

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 7 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	108	6682
133	95	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 8 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	74	7664
106	80	=== Ethernet === size: 108 src: f9:c8:d5:68:47:e5 dst: 44:0e:7c:1c:fa:6d === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 8 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	95	7666
125	118	=== Ethernet === size: 108 src: 53:f4:72:6a:4d:9e dst: fd:9b:74:b9:5f:d6 === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 8 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	80	7678
128	108	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 8 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	118	7683
134	28	=== Ethernet === size: 108 src: a3:db:32:49:14:19 dst: 49:07:cb:76:58:bb === IP === src: 4.0.0.4 dst: 7.0.0.2 ttl: 62 size: 84 === ICMP === type: REQUEST code: ZERO id: 1026 seq: 8 payloadSize: 56	SUCCESSFULLY_TRANSMITTED	ICMP	108	7684

ANNEXURE C: COMPREHENSIVE LOGS OF REPLICATION TESTING

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
123	113	=== Ethernet === size: 52 src: 69:0e:ab:ef:ae:78 dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 69:0e:ab:ef:ae:78 1.0.0.2 target: 00:00:00:00:00:00 1.0.0.1	SUCCESSFULLY _TRANSMITTED	ARP	45	139
123	45	=== Ethernet === size: 52 src: 1c:30:22:c1:ce:69 dst: 69:0e:ab:ef:ae:78 === ARP ARP_REPLY === sender: 1c:30:22:c1:ce:69 1.0.0.1 target: 69:0e:ab:ef:ae:78 1.0.0.2	SUCCESSFULLY _TRANSMITTED	ARP	113	139
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 1 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	140
140	138	=== Ethernet === size: 52 src: 39:18:99:3c:2b:8f dst: ff:ff:ff:ff:ff:ff === ARP ARP_REQUEST === sender: 39:18:99:3c:2b:8f 8.0.0.1 target: 00:00:00:00:00:00 8.0.0.2	SUCCESSFULLY _TRANSMITTED	ARP	113	141
140	113	=== Ethernet === size: 52 src: 5b:40:f1:b9:1b:51 dst: 39:18:99:3c:2b:8f === ARP ARP_REPLY === sender: 5b:40:f1:b9:1b:51 8.0.0.2 target: 39:18:99:3c:2b:8f 8.0.0.1	SUCCESSFULLY _TRANSMITTED	ARP	138	142
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 1 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	143
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 2 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	1139

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 2 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	1142
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 3 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	2135
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 3 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	2138
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 4 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	3134
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 4 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	3136
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 5 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	4136

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 5 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	4139
123	113	=== Ethernet === size: 108src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZEROid: 1025 seq: 6 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	5135
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 6 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	5137
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 7 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	6140
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 7 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	6144
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 8 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	7140

Cable ID	Dest ID	Details Text	Event Status	Packet Type	Source ID	Time-stamp
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 8 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	7145
123	113	=== Ethernet === size: 108 src: 69:0e:ab:ef:ae:78 dst: 1c:30:22:c1:ce:69 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 64 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 9 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	45	8136
140	138	=== Ethernet === size: 108 src: 39:18:99:3c:2b:8f dst: 5b:40:f1:b9:1b:51 === IP === src: 1.0.0.2 dst: 8.0.0.2 ttl: 63 size: 84 === ICMP === type: REQUEST code: ZERO id: 1025 seq: 9 payloadSize: 56	SUCCESSFULLY _TRANSMITTED	ICMP	113	8137

ANNEXURE D: CONFERENCE PROCEEDINGS

Wahl, R., Jordaan, A. & Joubert, A. (2020). Digitalisation for self-regulated vehicles via the Internet of Things. *Proceedings***.** NEMISA Digital Skills Summit and Research Colloquium, 11-13 March 2020, Birchwoord, Johannesburg**.**

ABSTRACT

Aim/Purpose

The persistent increase of vehicles on our road networks, alongside the human factor, necessitates reliable communication technology incorporated within Intelligent Transport Systems (ITS) to conform within the autonomous vehicle arena.

The aim of the research is to explore the realm of IoT and how it is integrated into self-regulated vehicles and other applications from a pragmatic standpoint. The research demonstrates the application of contemporary data network facilities together with detection and communication techniques, such as machine learning, to develop an effective IoT solution for directing self-regulated vehicles. An IoT model was constructed encompassing vehicular network technologies and based on empirical simulation to explain and substantiate the operation of such a system.

Background

IoT, by definition, deals with the transmission of immense volumes of data and therefore requires a new generation of data storage facilities, infrastructure and warehouse design geared for extremely fast data retrieval. The applications are boundless and can be used to aid in mitigating a crisis, for example, by way of measuring specific environmental conditions which could in turn warn the public of an impending earthquake, or the monitoring of farming equipment and subsequently reducing water usage in drought-stricken areas.

Any network of objects, such as vehicles, combined with electronic devices such as sensors and software connected via a communication system is encompassed within IoT (Ashton, 2009). This allows a more direct integration of the physical world with data systems, thereby reducing human interference with ensuing efficiency, accuracy and economic viability (Santucci, 2011; Lindner, 2015).

Vehicle ad-hoc systems (VANETS), through artificial intelligence (AI), have improved the efficiency of cross-vehicular communication and can thus perform best-route calculations on the fly based on real-time events that transpire while in transit to a destination. Literature has shown that with an increasing array of devices, stringent security standards are constantly created by organisations (Porup, 2016).

Methodology

A literature review was conducted on integrated routing mechanisms for self-regulated vehicles by way of IoT technology. Furthermore, the feasibility of adopting contemporary and fundamental packet routing networks as measure of routing self-regulated vehicles was explored.

Contribution

By utilising IoT technology, a conceptual model for routing self-regulated vehicles was identified and proved plausible through simulation.

Findings

The following findings can be deduced from literature:

- State-of-the-art wireless technologies are used in conjunction with self-regulated vehicles
- Packet routing technologies can be used as a concept for routing self-regulated vehicles

- Security continues to remain a concern in wireless technologies
- Self-regulated vehicles can have a positive environmental impact
- An IoT model can be used to validate the operation of routing system in the South African context
- Challenges exist surrounding the technical implementation of a routing system for self-regulated vehicles

Recommendations for Practitioners Present infrastructure and expenditure dictate that autonomously enhanced vehicles will have to co-exist alongside conventional vehicles (Litman, 2017). Practitioners can put laws into practice that prioritise safety to accommodate a collaborative automotive environment. Revise our education system to include digital literacy skills in preparation for 4IR with its new technologies in IoT and employment opportunities such as robotics, engineering, data science, software development, operations and testing. In addition, IoT security environments need to be constantly monitored to prevent interception.

Recommendation for Researchers IoT has become increasingly more challenging to manage due to its multi-facet characteristics. The ecosystem of collaborative information needs to be stored, analysed, and accessible at all times, and research can contribute towards enhancement in these areas, on which human life depends.

Impact on Society IoT requires players to create/develop and sell products, software developers, hardware designers, data scientists, and designers. AI and vehicle automation will need upkeep and upgrading (Daugherty et al., 2015), while new entrants will emerge in the market, such as sectors in vehicle operating systems. Among other favourable factors, regulating the routing of self-regulated vehicles will ensure a drastic reduction in road fatalities as well as a reduction in air pollution and greenhouse gases.

Future Research Research into the integration of IoT vehicular networks with G5 mobile technology (Cavalcanti et al., 2018) is imperative. Investigations should be carried out on alternative routing methods for self-regulated vehicles and on how to prevent probable attack from within the routing domain. Explore packet switching as a routing means by considering built-in repair mechanisms such as standby links automatically taking over should a primary link fail (Halba et al., 2018).

Keywords Internet of Things (IoT), self-regulated vehicle, innovation, digital skills, wireless technologies, network technologies

REFERENCES

Ashton, K. (2009). *That 'Internet of Things' thing.* Retrieved November 11, 2017 from http://www.rfidjournal.com/articles/view?4986

Cavalcanti, E.R., De Sousa, J.A.R., Spohn, M.A, Gomes, R.C. & Da Costa, F.B.F. (2018). VANETs research over the past decade: overview, credibility and trends. *Computer Communication Review,* 48(2), 31-39.

Daugherty, P., Negm, W., Banerjee, P. & Alter, A. E. (2015). *Driving unconventional growth through the Industrial Internet of Things.* Accenture.

Halba, K., Mahmoudi, C. & Griffor, E. (2018). Robust safety for autonomous vehicles through reconfigurable networking. *Electronic Proceedings in Theoretical Computer Science (EPTCS),* (269), 48-58.

Lindner, T. (2015). *The supply chain: changing at the speed of technology.* Retrieved November 11, 2017 from https://connectedworld.com/the-supply-chain-changing-at-the-speed-of-technology/

Litman, T. A. (2017). *Autonomous vehicle implementation predictions implications for transport planning.* Victoria Transport Policy Institute. Retrieved February 13, 2017 from: http://www.vtpi.org/avip.pdf

Porup, J.M. (2016). *Internet of Things' security is hilariously broken and getting worse.* Retrieved November 21, 2017 from https://arstechnica.com/information-technology/2016/01/how-to-search-the-internet-of- things-for-photos-of-sleeping-babies/

Santucci, G. (2011). *The Internet of Things between the revolution of the Internet and the metamorphosis of objects.* European Commission Community Research and Development Information Service.

ANNEXURE E: CONFERENCE POSTER

Wahl, R., Jordaan, A. & Joubert, A. (2020). Digitalisation for self-regulated vehicles via the Internet of Things. *Poster presentation*. NEMISA Digital Skills Summit and Research Colloquium, 11-13 March 2020, Birchwood, Johannesburg.

VUT
Vaal University of Technology

Digitalisation for Self-regulated Vehicles via the Internet of Things

Roger Wahl, Annelie Jordaan, Andre Joubert

Vaal University of Technology

NEMISA NPC
National Electronic Media Institute of South Africa

Introduction

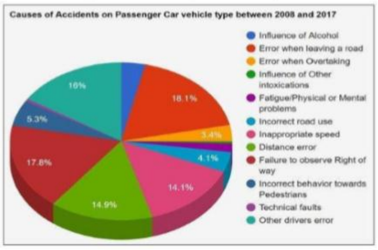
Continual increase of vehicles on road networks, alongside the human factor, necessitated the integration of the Internet of Things (IoT) driven by reliable communication technology incorporated within Intelligent Transport Systems (ITS) to conform within the autonomous vehicle arena.

This research aims to explore the realm of IoT and how it is integrated into self-regulated vehicles and other applications from a pragmatic standpoint. Application of contemporary data network facilities are demonstrated with detection and communication techniques such as machine learning to develop an effective IoT solution. An IoT model was constructed encompassing vehicular network technologies, based on empirical simulation to explain and substantiate the operation of system.

Background

Any network of objects such as vehicles, combined with electronic devices such as sensors and software connected via a communication system, are encompassed within IoT (Ashton, 2009), allowing a more direct integration of the physical world with information systems, thereby reducing human interference with ensuing efficiency, accuracy and economic viability.

Figure 1: Causes of accidents between 2008 and 2017 (Source: Khaliq et al., 2019)



Vehicle ad-hoc systems (VANETS), through Artificial Intelligence, have improved efficiency of cross-vehicular communication and can thus perform best-route calculations on the fly based on real-time events that transpire while in transit to a destination.

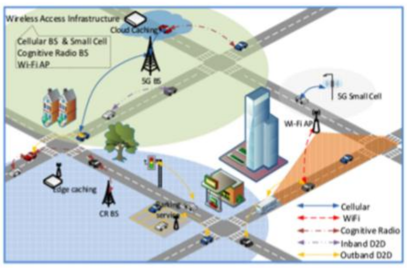
Routing

Reviewing V2V communication using VANETs, Abbasi and Kahn (2018) considered position-based protocols as opposed to topology-based protocols in urban settings, due to frequency of environmental topology changes. Insight is drawn to routing in forwarding techniques, means of junction selection, and methods of dealing with local peak traffic situations. Simulation of dynamic junction selection and static junction selection-based routing protocols was demonstrated. Researchers verified that for effective routing protocol, certain characteristics of VANETs resulted in unreliable communication. They claimed that high mobility and intermittent connectivity affected linkage and subsequent loss of packets within a network that fluctuates from congested to sparse. Researchers also felt that these attributes made security challenging, and established that the existing urban real-life environment is not suitably reflected in present routing VANET protocols and position-based protocols.

Zhang *et al*. (2016) propounded a congestion-aware routing algorithm to route a fleet of autonomous vehicles in a coordinated manner that does not increase congestion. Simulation demonstrated that routing and rebalancing (empty vehicle trips between drop-off and next collection) of self-regulated vehicles in a coordinated manner in Manhattan that provided mobility-on- demand service to customers could reduce pollution, demand for parking, and cost of travel.

Once a customer is serviced, the vehicle drives itself to the next customer and the network ensures that the proportion entering each node is equivalent to the proportion exiting node. Rebalancing therefore did not increase the total number of self-regulated vehicles on the road, as they were intelligently routed to avoid increasing congestion by optimising routes of both passenger-carrying and empty vehicles.

Figure 2: Wireless access infrastructure (Source: Cheng & Shen, 2016)



Other researchers also postulated that in managing a fleet of self-regulated vehicles, the mobility-on-demand system could help reduce capacity if multiple rides are serviced with a single trip. A New York study by Santi *et al.* (2014) showed that a taxi trip shared by two riders in Manhattan increased travel time in 80% of cases by only a few minutes, which was later verified by Alonso-Mora *et al.* (2017).

Method

By employing IoT vehicular technology, a computer model was designed and built representing the real-life environment to demonstrate through algorithms how the concept would function. Simulation models can be used to verify theory (Olivier, 2011). While computer security was addressed in line with Construct Theory and theoretical predictions, a simulation was performed and the results were analysed. As a requirement in computer simulations (Gulyás & Kampis, 2015), replication testing was performed using different data input.

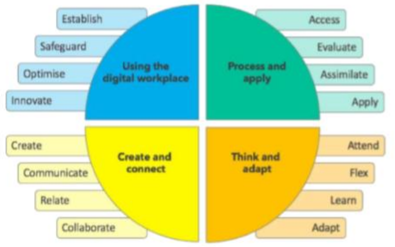
Digitalisation

"Digitalisation can inter alia be defined as the adoption of digital technologies to modify a business model. The aim is to create a value from the use of new, advanced technologies by exploiting digital network dynamics and the giant digital flow of information" (IGI Global, 2020).

Digitalisation beyond the discipline of the Internet of Things primarily refers to human-generated (digital) information and centralised autonomous systems. It creates digitisation of the physical world through dynamic and disseminated computation and algorithms into devices which humans are becoming more naturally dependant on.

Concerns are raised about Artificial Intelligence and human activity, cybersecurity and privacy, justifying an appetite for new skills and development. Digitalisation is knowledge gain; it encompasses understanding and application. As a foundation for *Design Thinking,* digital literacy holds responsible those who benefit from using the technology (Knitl & Erdebil, 2019).

Figure 3: Workplace Digitalisation (Source: Marsh, 2018)



Results

The experiment conducted in this research conformed to expected behaviour with similar outcomes in the analysis of repeat tests, thereby verifying initial results. This equated to the consolidation of the virtual world and the real world through computation and theoretical reasoning (Sastry, 1997).

Future Research

Further research is needed in the following areas:

- VANET routing performance, as environmental characteristics are not accurately reflected (Abbasi & Khan, 2018)
- Security research in multi-faceted IoT automotive computer systems with possible in-built repair systems
- VANET communication in an infrastructure-less environment to prepare assimilation of vehicular networks with 5G mobile technology

Bibliography

Abbasi, I. A. & Kahn, A. S. (2018). *A review of vehicle to vehicle communication protocols for VANETs in the urban environment*. Future in Internet. http://www.mdpi.com/journal/futureinternet [3/12/2018].

Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E. & Rus, D. (2017). On-demand high-capacity ridesharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences (PNAS),* (114):462–67.

Ashton, K. (2009). *That 'Internet of Things' thing.* http://www.rfidjournal.com/articles/view?4986 [11/11/2017].

Cheng, N. & Shen, X. (2016). *Introduction in: Opportunistic Spectrum Utilization in Vehicular Communication Networks*. https://link.springer.com/chapter/10.1007/978-3-319-20445-1_1) [03/01/2020].

Gulyás, S. L. & Kampis, G. (2015). Models, representations and comparisons in computer simulations. *Proceedings*. 4th International Young Scientists Conference on Computational Science. *Procedia Computer Science,* (66):5–12.

IGI Global. (2020). *What is digitalisation*. https://www.igi-global.com/dictionary/it-strategy-follows-digitalization/7748 [19/02/2020].

Khaliq, K. A., Chughtai, O., Shahwani, A., Qayyum, A. & Pannek, J. (2019). Road accidents detection, data collection and data analysis using V2X Communication and Edge/Cloud Computing. *Electronics,* 8:896.

Knitl, W. & Erdebil, N. (2019). *Spring2 Innovation: Internet of Things and Design Thinking pave the way to digital literacy.* http://www.spring2innovation.com/2019/06/internet-of-things-design-thinking-pave-the-way-to-digital-literacy/ [15/01/2020].

Marsh, E. (2018). *The Digital Workplace Skills Framework ensuring the workforce is ready to work digitally*. Digital Work Research. https://digitalworkresearch.com/wp-content/uploads/2018/08/The-Digital-%20%20Workplace-Skills-Framework-final.pdf/ [18/01/2020].

Olivier, M. S. (2011). *Information technology research: A practical guide for computer science and informatics*. 3rd ed. Pretoria: Van Schaik.

Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S. H. & Ratti, C. (2014). Quantifying the benefits of vehicle pooling with shareability networks. *Proceedings of the National Academy of Sciences (PNAS), (111):13290–94.*

Sastry, M. A. (1997). Problems and paradoxes in a model of punctuated organisational change. *Administrative Science Quarterly,* 42(2):237–275.

Zhang, R., Rossi, F. & Pavoni, M. (2016). *Routing autonomous vehicles in congested transportation networks: Structural properties and coordination algorithms*. arXiv:1603.00939v2. Stanford University.