

TRFS

DESENVOLVIMENTO DE UM PROTÓTIPO PARA SIMULAÇÃO MICROSCÓPICA DE TRÁFEGO

JOÃO PEDRO SOUSA GAMBOA
(LICENCIADO EM ENGENHARIA CIVIL)

DISSERTAÇÃO PARA OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA CIVIL NA ÁREA DE
ESPECIALIZAÇÃO DE VIAS DE COMUNICAÇÃO E TRANSPORTES

ORIENTADOR:

MESTRE PAULO JOSÉ DE MATOS MARTINS, PROFESSOR ADJUNTO DO ISEL

JÚRI

PRESIDENTE:

DOUTOR LUCIANO ALBERTO DO CARMO JACINTO, PROFESSOR ADJUNTO DO ISEL

VOGAIS:

DOUTOR FILIPE MANUEL MERCIER VILAÇA E MOURA, PROFESSOR ADJUNTO DO IST
MESTRE PAULO JOSÉ DE MATOS MARTINS, PROFESSOR ADJUNTO DO ISEL

FEVEREIRO DE 2016

RESUMO

Nas últimas décadas, com o aumento do poder de processamento dos computadores disponíveis para o utilizador comum, tem-se verificado uma crescente adesão a meios informáticos que permitem a análise e resolução de problemas de engenharia de forma significativamente mais eficiente e, frequentemente, mais fiel à realidade do que recorrendo aos métodos simplificados clássicos.

No âmbito da Engenharia de Tráfego, observa-se não só a tentativa de adaptação dos modelos analíticos a formatos informáticos, mas também o desenvolvimento de programas informáticos que permitem a criação e representação de modelos que simulam o comportamento dos utilizadores da rede viária a diferentes escalas. Por um lado, ao nível macroscópico, observa-se a evolução dos fluxos de tráfego na rede (simulação macroscópica) e, por outro, ao nível microscópico, no qual se desenvolve o modelo à escala do indivíduo, simula-se o comportamento de cada agente e a sua influência no desempenho da rede (simulação microscópica ou microssimulação).

Neste documento, expõem-se os campos de aplicação comercial e académica das ferramentas de simulação de tráfego disponíveis no mercado, comparando as funcionalidades presentes em cinco dos programas de simulação mais populares e guiando o leitor para a escolha da ferramenta de análise de tráfego apropriada às suas necessidades. Adicionalmente, é apresentada uma metodologia para o desenvolvimento de modelos de simulação microscópica, uma avaliação das limitações destes modelos e uma análise da forma como os resultados obtíveis com ferramentas de simulação se comparam com metodologias analíticas clássicas (*HCM – Highway Capacity Manual*).

Finalmente, apresentam-se os modelos comportamentais de *car-following* e *lane changing* que atuam na inteligência artificial dos condutores presentes na rede e desenvolve-se um simulador de tráfego em *Java*. Este último, com a finalidade de permitir a análise da influência dos parâmetros de calibração dos modelos comportamentais no tráfego, possibilitando ao utilizador a sua manipulação em tempo real.

Adicionalmente, efetua-se uma revisão do programa comercial *Vissim* [1], demonstrando as etapas e precauções a tomar no desenvolvimento de um modelo. Neste âmbito, desenvolve-se um modelo da implementação de uma via de alta ocupação na praça de portagens da ponte 25 de Abril, em Lisboa, Portugal.

Palavras-chave: Simulação; Tráfego; Microssimulação; *Car-following*; *Vissim*;

ABSTRACT

Over the course of the last few decades, along with the advancements in processing power of personal computers, there has been a growing gravitation towards the use of programmatic means for the analysis and resolution of engineering problems in a more efficient fashion and, frequently, with a better representation of reality than classic simplified methodologies.

In Traffic Engineering we perceive, not only the adaptation of analytical models to computerized environments for greater efficiency, but also the development of software capable of simulating the behaviour of individual drivers at multiple scales. First, at the macroscopic level, where it's possible to see the evolution of traffic flows throughout the network and, more recently, at the microscopic level, where each driver and its influence on the network's performance measures is tracked and analysed individually. This kind of models are entitled, respectively, macrosimulation and microsimulation.

This document presents the scope of commercial and academic application of the traffic simulation tools available in the market, as well as the comparison between the functionalities of five of the most popular microsimulation software packages, guiding the reader for the choice of the most appropriate tool for the job. Additionally, it demonstrates a methodology for the development of simulation models, the limitations of their use and also the relationship between the results that one can obtain from simulation and classic analytical methodologies such as the Highway Capacity Manual.

Finally, a review of driver behaviour models is presented, such as the *car-following* and *lane changing* models that operate in the artificial intelligence of drivers replicated by simulation applications. A stand-alone software for the analysis of the effects of model parameters in the traffic simulation is developed. This application intends to provide users the ability to change the value of each parameter in real time, visualizing its influence on the driving behaviour.

Additionally, an appraisal of the commercial software package *Vissim (PTV)* is made, demonstrating the steps and precautions required for the development of a microsimulation model. Also, a model of the toll plaza leading to the Tagus Bridge, in Lisbon, Portugal is developed, to showcase the functionalities that are possible to obtain using simulation software.

Keywords: Simulation; Traffic; Microsimulation; Car-following; Vissim;

LISTA DE SIGLAS E ACRÔNIMOS

API	<i>Application Programming Interface</i>
CAD	<i>Computer-Aided Design</i>
COM	<i>Component Object Model</i>
CSV	<i>Comma-Separated Values</i>
DCB	<i>Discrete Choice-Based</i>
DLC	<i>Discretionary Lane Change</i>
FHWA	<i>Federal Highway Administration</i>
FOSS	<i>Free, Open-Source Software</i>
GIS	<i>Geographic Information System</i>
HCM	<i>Highway Capacity Manual</i>
MLC	<i>Mandatory Lane Change</i>
PTV	<i>Planung Transport Verkehr AG</i>
RB	<i>Rule-Based</i>
SIG	Sistemas de Informação Geográfica
TC	Transporte Coletivo
TI	Transporte Individual
TRB	<i>Transportation Research Board</i>
TSS	<i>Transport Simulation Systems</i>
UVE/H	Unidade de Veículos Equivalentes por Hora
VAO	Via de Alta Ocupação

ÍNDICE

1. INTRODUÇÃO	1
1.1. Enquadramento.....	1
1.2. Objetivos e metodologia.....	2
1.3. Estrutura do documento	3
1.4. Estado da arte em Portugal.....	4
2. SIMULAÇÃO DE TRÁFEGO	7
2.1. Enquadramento.....	7
2.2. Aplicações da simulação na engenharia de transportes.....	8
2.3. Metodologias de análise de tráfego.....	9
2.3.1. <i>Modelos de estimação de procura</i>	9
2.3.2. <i>Metodologias analíticas (HCM)</i>	10
2.3.3. <i>Otimização de semáforos</i>	10
2.3.4. <i>Simulação de tráfego</i>	10
2.4. Escalas de simulação de tráfego.....	11
2.4.1. <i>Simulação macroscópica</i>	11
2.4.2. <i>Simulação mesoscópica</i>	12
2.4.3. <i>Simulação microscópica</i>	12
2.4.4. <i>Simulação híbrida</i>	13
2.4.5. <i>Simulação nanoscópica</i>	13
2.4.6. <i>Contexto de aplicação das escalas de simulação</i>	14
2.5. Ferramentas de simulação de tráfego	16
2.5.1. <i>Programas comerciais de simulação microscópica</i>	18
2.6. Desenvolvimento de modelos de simulação de tráfego.....	20
2.6.1. <i>Metodologia</i>	20
2.6.2. <i>Definição do âmbito da análise</i>	21
2.6.3. <i>Recolha e preparação de dados</i>	21

2.6.4.	<i>Desenvolvimento do modelo</i>	22
2.6.5.	<i>Verificação de erros</i>	22
2.6.6.	<i>Calibração do modelo</i>	23
2.6.7.	<i>Análise de soluções e comparação de resultados</i>	23
2.6.8.	<i>Relatório final e documentação técnica</i>	23
2.7.	<i>Limitações da simulação de tráfego</i>	24
2.8.	<i>Comparação de modelos de simulação com o HCM</i>	25
2.8.1.	<i>Vantagens dos procedimentos do HCM</i>	25
2.8.2.	<i>Desvantagens dos procedimentos do HCM</i>	25
2.8.3.	<i>Diferenças nos indicadores de desempenho da rede</i>	26
3.	MODELOS COMPORTAMENTAIS DE TRÁFEGO	27
3.1.	<i>Car-following</i>	27
3.1.1.	<i>Enquadramento</i>	28
3.1.2.	<i>Modelo de Wiedemann '74</i>	30
3.1.3.	<i>Modelo de Gipps</i>	36
3.1.4.	<i>Modelo de Fritzsche</i>	37
3.2.	<i>Lane changing e gap acceptance</i>	41
3.2.1.	<i>Enquadramento</i>	41
3.2.2.	<i>Lane changing no Aimsun</i>	42
3.2.3.	<i>Lane changing no Vissim</i>	42
4.	DESENVOLVIMENTO DE UM SIMULADOR DE TRÁFEGO	43
4.1.	<i>Enquadramento</i>	43
4.1.1.	<i>Apresentação do programa</i>	44
4.2.	<i>Importação de rede SIG</i>	48
4.2.1.	<i>Leitura da rede</i>	49
4.2.2.	<i>Simplificação e suavização da rede</i>	50
4.3.	<i>Tráfego</i>	52

4.3.1.	<i>Definição de percursos</i>	53
4.3.2.	<i>Car-following (Wiedemann '74)</i>	54
4.3.3.	<i>Lane changing</i>	65
4.4.	Representação gráfica	67
4.5.	Resultados	68
4.5.1.	<i>Recolha de dados</i>	69
4.5.2.	<i>Análise dos resultados</i>	70
5.	CONCLUSÕES	73
	REFERÊNCIAS BIBLIOGRÁFICAS	77
	APÊNDICE A. MODELAÇÃO DE REDES EM VISSIM	81
A.1.	Enquadramento.....	81
A.2.	Dados de base para a simulação	82
A.3.	Rede	82
A.3.1.	<i>Plano de fundo</i>	83
A.3.2.	<i>Arcos e conetores</i>	84
A.3.3.	<i>Representação da rede</i>	86
A.3.4.	<i>Limites de velocidade</i>	87
A.3.5.	<i>Prioridades</i>	88
A.3.6.	<i>Semáforos</i>	89
A.3.7.	<i>Objetos 3D</i>	91
A.4.	Tráfego.....	92
A.4.1.	<i>Veículos</i>	92
A.4.2.	<i>Intervalos de tempo</i>	94
A.4.3.	<i>Introdução de fluxos na rede</i>	94
A.4.4.	<i>Modelação de percursos</i>	95
A.5.	Simulação	96
A.5.1.	<i>Definição dos parâmetros para a simulação</i>	96

A.5.2.	<i>Apresentação</i>	97
A.6.	Obtenção de resultados	98
A.6.1.	<i>Tempo de percurso</i>	98
A.6.2.	<i>Filas de espera</i>	98
A.7.	Estudo de caso – <i>Via de Alta Ocupação</i> no acesso à praça de portagens da ponte 25 de Abril	98
A.7.1.	<i>Enquadramento</i>	98
A.7.2.	<i>Objetivos</i>	99
A.7.3.	<i>Recolha de dados</i>	99
A.7.4.	<i>Modelação da rede</i>	101
A.7.5.	<i>Situação atual</i>	104
A.7.6.	<i>Implantação da VAO</i>	106
A.7.7.	<i>Simulação de alternativas</i>	108
A.7.8.	<i>Resultados</i>	110
A.8.	Anexos	111
A.8.1.	<i>Fluxos de input (Vissim)</i>	111
A.8.2.	<i>Output de filas de espera</i>	112
A.8.3.	<i>Output de tempos de viagem</i>	112
ANEXO I.	<i>POSTER TRFS</i>	113

ÍNDICE DE FIGURAS

<i>Figura 2.1 – Simulação macroscópica</i>	11
<i>Figura 2.2 – Simulação mesoscópica</i>	12
<i>Figura 2.3 – Simulação microscópica</i>	12
<i>Figura 2.4 – Simulação híbrida</i>	13
<i>Figura 2.5 – Metodologia de desenvolvimento de modelos de simulação</i>	20
<i>Figura 6 – Exemplo de erro na colocação de sinalização vertical em Vissim</i>	22
<i>Figura 3.1 – Nomenclatura utilizada na descrição de agentes em modelos de car-following.</i>	27
<i>Figura 3.2 – Limites de reação do modelo de Wiedemann '74 [17].</i>	31
<i>Figura 3.3 – Procedimento de decisão no modelo de Wiedemann '74 [54]</i>	34
<i>Figura 3.4 – Limiares de reação do modelo de Fritzsche.[47]</i>	39
<i>Figura 3.5 – Zonas de mudança de via no Aimsun. [49]</i>	42
<i>Figura 4.1 – Logotipo TRFS</i>	43
<i>Figura 4.2 – Estrutura do programa em Java</i>	44
<i>Figura 4.3 – Estrutura da interface gráfica</i>	44
<i>Figura 4.4 – Main Menu (menu inicial principal)</i>	45
<i>Figura 4.5 – Start Menu (menu de iniciação)</i>	45
<i>Figura 4.6 – Simulation Screen (ecrã de simulação)</i>	46
<i>Figura 4.7 – Settings Menu (menu de opções)</i>	47
<i>Figura 4.8 – About TRFS (acerca do TRFS)</i>	47
<i>Figura 4.9 – Layer com a rede que se pretende analisar no TRFS</i>	48
<i>Figura 4.10 – Atributos da rede modelada em SIG</i>	48
<i>Figura 4.11 – Lógica de criação de um mapa no TRFS</i>	49
<i>Figura 4.12 – Efeito do algoritmo de simplificação da rede</i>	50
<i>Figura 4.13 – Efeito da suavização da rede com o método de Chaikin</i>	51
<i>Figura 4.14 - Resultado dos algoritmos de simplificação e suavização no TRFS</i>	51
<i>Figura 4.15 – Ciclo de atualização do estado do tráfego na rede</i>	52
<i>Figura 4.16 – Algoritmo de deteção do veículo em frente</i>	57
<i>Figura 4.17 – Procedimento de decisão no modelo de Wiedemann '74 [54]</i>	59

<i>Figura 4.18 – Representação do troço de estrada.....</i>	<i>61</i>
<i>Figura 4.19 – Modelo em Matlab: Aspecto geral do simulador.....</i>	<i>62</i>
<i>Figura 4.20 – Modelo em Matlab: Fluxo reduzido.....</i>	<i>62</i>
<i>Figura 4.21 – Modelo em Matlab: Fluxo elevado.....</i>	<i>63</i>
<i>Figura 4.22 – Lógica de avaliação do modelo de mudança de via TRFS Lane changing.....</i>	<i>65</i>
<i>Figura 4.23 – Sequência de desenho do cenário.....</i>	<i>67</i>
<i>Figura 4.24 – Ampliação dos agentes autónomos do TRFS.....</i>	<i>68</i>
<i>Figura 4.25 – Comparação da rede em fotografia aérea com a rede importada pelo TRFS.....</i>	<i>69</i>
<i>Figura 4.26 – ID's dos arcos selecionados para a extração de dados.....</i>	<i>69</i>
<i>Figura 4.27 – Representação do fluxo (Veic/h) na rede.....</i>	<i>72</i>
<i>Figura 4.28 – Representação da densidade do tráfego (Veic/Km) na rede.....</i>	<i>72</i>
<i>Figura A-1 – Logotipo Vissim.....</i>	<i>81</i>
<i>Figura A-2 – Altimetria: Janela de Propriedades.....</i>	<i>86</i>
<i>Figura A-3 – Altimetria: Visualização 3D da atribuição de altimetria.....</i>	<i>86</i>
<i>Figura A-4 – Representação da espessura dos arcos.....</i>	<i>87</i>
<i>Figura A-5 – Cenários possíveis em zonas de conflito.....</i>	<i>89</i>
<i>Figura A-6 – Esquematização de Grupos de Semáforos.....</i>	<i>90</i>
<i>Figura A-7 – Temporização dos Sinais.....</i>	<i>90</i>
<i>Figura A-8 – Representação 3D dos Semáforos.....</i>	<i>91</i>
<i>Figura A-9 – Árvores no modo 3D.....</i>	<i>92</i>
<i>Figura A-10 – Diagrama de Vehicle Types no Vissim.....</i>	<i>93</i>
<i>Figura A-11 – Classes de Veículos.....</i>	<i>93</i>
<i>Figura A-12 – Composições de Veículos.....</i>	<i>94</i>
<i>Figura A-13 – Intervalos de Tempo.....</i>	<i>94</i>
<i>Figura A-14 – Volumes de Tráfego.....</i>	<i>95</i>
<i>Figura A-15 - Parque de Estacionamento e Praça de Portagem.....</i>	<i>96</i>
<i>Figura A-16 – Criação de Camera Positions.....</i>	<i>97</i>
<i>Figura A-17 – Modelo da praça de portagens da Ponte 25 de Abril.....</i>	<i>99</i>
<i>Figura A-18 - Percentagem de utilizadores por modo de pagamento.....</i>	<i>100</i>

<i>Figura A-19 – Percentagem de utilizadores por taxa de ocupação</i>	<i>100</i>
<i>Figura A-20 – Alargamento do número de vias</i>	<i>102</i>
<i>Figura A-21 – Modelação dos pórticos de portagem</i>	<i>103</i>
<i>Figura A-22 – Afetação do tráfego aos pórticos de portagem</i>	<i>103</i>
<i>Figura A-23 – Representação 3D da praça de portagem</i>	<i>104</i>
<i>Figura A-24 – Representação dos pares OD.....</i>	<i>104</i>
<i>Figura A-25 – Comprimentos das filas de espera na situação atual</i>	<i>105</i>
<i>Figura A-26 – Planta da implantação da VAO</i>	<i>106</i>
<i>Figura A-27 – Proposta inicial de disposição dos pórticos de portagem</i>	<i>106</i>
<i>Figura A-28 – Congestionamento na VAO resultado da primeira disposição de pórticos.....</i>	<i>107</i>
<i>Figura A-29 – Proposta final de disposição de pórticos de portagem</i>	<i>107</i>
<i>Figura A-30 – Tempos de percurso na alternativa 1.....</i>	<i>108</i>
<i>Figura A-31 – Comprimento das filas de espera na alternativa 1.....</i>	<i>108</i>
<i>Figura A-32 – Comprimento das filas de espera na alternativa 2.....</i>	<i>109</i>
<i>Figura A-33 – Evolução das condições da rede</i>	<i>110</i>

ÍNDICE DE TABELAS

<i>Tabela 2-1 – Relevância da utilização de ferramentas de análise de tráfego</i>	<i>9</i>
<i>Tabela 2-2 – Aplicabilidade de escalas de simulação</i>	<i>14</i>
<i>Tabela 2-3 – Aplicabilidade de escalas de simulação em função dos recursos da organização.</i>	<i>15</i>
<i>Tabela 2-4 – Aplicabilidade de escalas de simulação em função das funcionalidades pretendidas....</i>	<i>15</i>
<i>Tabela 2-5 – Listagem de aplicações comerciais em desenvolvimento ativo</i>	<i>18</i>
<i>Tabela 2-6 - Comparação de funcionalidades de aplicações comerciais de microssimulação</i>	<i>19</i>
<i>Tabela 3-1- Parâmetros de calibração do modelo de Wiedemann '74 propostos em Higgs, 2011.....</i>	<i>35</i>
<i>Tabela 4-1 – Condições iniciais da simulação de demonstração.</i>	<i>70</i>
<i>Tabela 4-2 - Matriz OD da rede.....</i>	<i>70</i>
<i>Tabela 4-3 – Tempo médio de viagem por par OD (segundos).....</i>	<i>70</i>
<i>Tabela 4-4 – Velocidade média por par OD (Km/h).....</i>	<i>71</i>
<i>Tabela 4-5 – Tempo médio em travagem por par OD (segundos).</i>	<i>71</i>
<i>Tabela 4-6 – Tempo médio com o desejo de mudar de via por par OD (segundos).....</i>	<i>71</i>
<i>Tabela A-1 – Barra de Ferramentas do Vissim.....</i>	<i>83</i>
<i>Tabela A-2 – Percentagem de utilizadores por modo de pagamento (Fonte: InIR, IP).....</i>	<i>100</i>
<i>Tabela A-3 – Percentagem de utilizadores por taxa de ocupação (Fonte: InIR, IP).....</i>	<i>100</i>
<i>Tabela A-4 – Fluxos de entrada na rede. (Fonte: InIR, IP).....</i>	<i>101</i>
<i>Tabela A-5 – Fluxos de entrada na rede por tipo de veículo</i>	<i>101</i>
<i>Tabela A-6 – Tempos de percurso na situação atual</i>	<i>105</i>
<i>Tabela A-7 – Tempos de percurso na alternativa 1</i>	<i>108</i>
<i>Tabela A-8 – Tempos de percurso na alternativa 2</i>	<i>109</i>
<i>Tabela A-9 – Evolução das condições da rede.....</i>	<i>110</i>
<i>Tabela A-10 – Fluxos de input: Período das 7h às 11h. (Fonte: InIR, IP)</i>	<i>111</i>
<i>Tabela A-11 – Fluxos de input: Distribuição por classes na situação atual.</i>	<i>111</i>
<i>Tabela A-12 – Fluxos de input: Distribuição por classes na alternativa 1.</i>	<i>111</i>
<i>Tabela A-13 – Output da simulação: Filas de espera.</i>	<i>112</i>
<i>Tabela A-14 – Output da simulação: Tempos de viagem.</i>	<i>112</i>

1. INTRODUÇÃO

1.1. ENQUADRAMENTO

O congestionamento de tráfego em meio urbano exerce um impacto adverso na qualidade de vida dos condutores e amplia o potencial de ocorrência de atrasos e acidentes [2][3]. Considerando o previsível agravamento destes aspetos [3], é fundamental que os profissionais de análise de tráfego procurem otimizar a eficácia das redes de transportes já existentes, através de melhorias operacionais.

Os modelos de simulação microscópica têm vindo a ganhar terreno [4] não só no âmbito do planeamento, mas também como ferramentas de apoio à conceção de vias de comunicação, infraestruturas de transporte público e parques de estacionamento [5]. Permitem recriar fenómenos reais com um nível de detalhe extraordinário, sendo afetados pelos fluxos de tráfego na rede, assim como pela geometria do traçado e pelo modo de condução dos utilizadores.

Um dos motivos para a popularização de programas de simulação foi a necessidade de modelar e analisar sistemas de transportes complexos em condições de saturação. Nestas situações, as metodologias analíticas clássicas perdem sensibilidade por, tipicamente, se focarem numa só interseção, ignorando o efeito da propagação de filas de espera na restante rede [6]. Por sua vez, os modelos de simulação de tráfego são capazes de acompanhar a relação entre diversas áreas de congestionamento e de estimar o modo como todas interagem entre si. Adicionalmente, os sistemas de transporte são inerentemente dinâmicos, uma vez que os seus agentes variam no tempo e no espaço com uma aleatoriedade difícil de prever e estimar com metodologias de cálculo que assumem um modelo de natureza estática.

Atualmente, existem diversas ferramentas de simulação de tráfego disponíveis no mercado [1][7][8][9][10], desenvolvidas para a análise de tráfego a diferentes escalas e com diferentes necessidades no que diz respeito ao tipo de dados a introduzir, à experiência do corpo técnico e ao tempo necessário para o desenvolvimento de modelos. Por conseguinte, não existe um programa de simulação único que esteja capacitado para dar resposta a todas as necessidades de uma determinada entidade.

Apesar da crescente popularidade da simulação como ferramenta de análise de tráfego, existe ainda uma carência no estudo do efeito dos parâmetros de calibração utilizados nos modelos matemáticas que coordenam o comportamento dos agentes de tráfego [11]. Em particular, é notável a escassez de informação no que diz respeito à variabilidade de parâmetros de calibração em função do ambiente cultural simulado e na forma como

diferentes modelos matemáticos se complementam ou interagem, tornando-se interessante o seu estudo.

1.2. OBJETIVOS E METODOLOGIA

Os programas de simulação atualmente disponíveis no mercado, ainda que sigam filosofias semelhantes, operam de formas distintas e têm objetivos, âmbitos e necessidades diferentes. De um modo geral, cada fabricante de *software* implementa modelos matemáticos de comportamento de agentes diferentes dos seus competidores e, tipicamente, negam ao utilizador a manipulação e visualização em tempo real dos efeitos da calibração dos seus parâmetros. Deste modo, estabelece-se o objetivo de desenvolver um simulador de tráfego com um foco na abertura à implementação de diferentes modelos de comportamento de condutores e uma interface que permita o utilizador manipular e estudar os seus efeitos na qualidade da simulação.

Em particular, este documento tem 4 objetivos principais:

Primeiramente, introduzir a temática da simulação de tráfego, enfatizando o seu contexto de utilização em engenharia de transportes, face às ferramentas tradicionais de análise de tráfego, expondo os diferentes tipos de simulação e as suas aplicações.

Seguidamente, apresentar alguns dos fatores mais importantes a considerar na seleção de programas de simulação microscópica e efetuar uma revisão das ferramentas disponíveis no mercado, avaliando a sua portabilidade para outras escalas de simulação e o tipo de licenciamento requerido para a sua utilização. Adicionalmente, é desenvolvida uma análise comparativa de cinco dos programas de simulação mais populares no mercado¹, em função dos aspetos acima mencionados.

Em terceiro lugar, explorar os modelos comportamentais de simulação de tráfego utilizados na modelação do comportamento dos agentes intervenientes em microssimulação, apresentando os modelos de *car-following*² que originaram aqueles hoje aplicados em programas comerciais e explorando em maior detalhe alguns dos mais usados atualmente.

Por fim, com a finalidade de desenvolver uma ferramenta que permita a visualizar o efeito de parâmetros de calibração nos modelos comportamentais de tráfego, é desenvolvido um simulador de tráfego com uma interface gráfica preparada para a manipulação destes valores em tempo real.

¹ *Vissim* [1], *TransModeller* [7], *Cube Dynasim* [8], *Aimsun* [9], *Paramics* [10]

² *GHR*, 1961 [12], *Gipps*, 1981 [13], *Kometani*, 1959 [14], *Helly*, 1961 [15], *Kikuchi*, 1992 [16]

Adicionalmente, efetuar a revisão das funcionalidades de um programa comercial de simulação microscópica [1] e demonstrar o processo de desenvolvimento de modelos de simulação através da modelação de um caso real, a praça de portagens da ponte 25 de Abril, em Lisboa.

1.3. ESTRUTURA DO DOCUMENTO

Capítulo 2:

No capítulo 2, é efetuada uma revisão do estado da arte da simulação de tráfego, expondo as suas aplicações e relação com outras metodologias de análise de tráfego. São enumerados os diferentes tipos de simulação e o contexto da respetiva aplicação. Adicionalmente, é apresentada uma revisão dos programas de simulação de tráfego disponíveis no mercado, comparando as principais características e funcionalidades daqueles com maior popularidade.

É descrita uma metodologia geral de desenvolvimento de modelos de simulação de tráfego, assim como as limitações do uso destas ferramentas e, por último, o modo como se comparam com as metodologias propostas no *HCM* [6].

Capítulo 3:

No capítulo 3, são explorados os modelos matemáticos de comportamento de condutores que servem de base aos programas de microssimulação: *car-following* e *lane changing*. Posteriormente, é brevemente mencionado o contexto histórico das tipologias de modelo que deram origem às formulações atualmente usadas em programas comerciais: *GHR* [12], *CA* [13], *Helly* [14], *AP* [15] e lógica difusa [16]. Finalmente, são expostos os modelos de *car-following* e *lane changing* utilizados em três programas comerciais: *Vissim* (*Wiedemann'74*) [17], *Aimsun* (*Gipps*) [13] e *Paramics* (*Fritzsche*) [18].

Capítulo 4:

No capítulo 4, é apresentado o desenvolvimento de um programa de simulação, designado *TRFS* (*contração de Traffic Simulator*), que permite a manipulação em tempo real dos parâmetros de calibração envolvidos na formulação de modelos de *car-following*, observando o efeito da sua alteração diretamente no tráfego, através de uma infraestrutura preparada para representar redes importadas diretamente de *SIG*.

Apêndice A:

No Apêndice A, apresenta-se o programa comercial de simulação microscópica *Vissim* [1] e o processo de desenvolvimento de modelos de simulação com esta aplicação, desde o desenho da rede à afetação de tráfego, apresentação 3D e obtenção de resultados.

Seguidamente, as metodologias exploradas ao longo do capítulo são aplicadas no desenvolvimento de um modelo da praça de portagens da ponte 25 de Abril, em Lisboa, para analisar a viabilidade da implantação de uma via de alta ocupação no acesso à ponte, partindo do nó de Almada. São simuladas duas soluções alternativas e comparam-se os resultados obtidos.

1.4. ESTADO DA ARTE EM PORTUGAL

As ferramentas de simulação de tráfego têm vindo a ganhar importância no nosso país, com o aparecimento de artigos e documentos académicos acerca de análises de redes de tráfego desenvolvidas com o auxílio de ferramentas de simulação. Paralelamente, surgem documentos realizados no âmbito do desenvolvimento e calibração de modelos de simulação.

António Vasconcelos, Álvaro Seco e Ana Silva (2014), em “*Hybrid calibration of microscopic simulation models*” [19] apresentam um procedimento para calibrar o modelo de *Gipps* de *car-following* com base em dados macroscópicos.

Carlos Azevedo (2014), em “*Probabilistic Safety Analysis Using Traffic Microscopic Simulation*” [20] desenvolve um modelo genérico de avaliação probabilística da segurança rodoviária para integração em ferramentas de simulação microscópica de tráfego. Este autor recorre à simulação microscópica para a geração de trajetórias artificiais de diferentes tipos de acidentes e para a calibração do modelo proposto.

Joaquim M. Macedo, Agostinho Benta e Luís Picado-Santos (2013), em “*Utilização Da Microssimulação No Desenvolvimento De Uma Metodologia De Avaliação Dos Níveis De Serviço Em Autoestradas*” [21] comparam os resultados do cálculo do nível de serviço em autoestradas gerados através do *HCM* com os resultados obtidos utilizando as ferramentas de simulação *Aimsun* e *Vissim*.

Joaquim M. Macedo e Agostinho Benta (2013), em “*Seleção Da Tipologia De Uma Intersecção Utilizando Um Simulador Microscópico De Tráfego – Um Caso De Estudo*” [22] apresentam um caso no qual se pretende analisar, através de um simulador microscópico

de tráfego, o desempenho de várias soluções alternativas para uma intersecção localizada em Aveiro.

António Vasconcelos, Álvaro Seco e Ana Silva (2012), em “*Comparison of procedures to estimate critical headways at roundabouts*” [23] utilizam imagens de vídeo para a análise de trajetórias de veículos em rotundas urbanas, com vista ao desenvolvimento de um modelo de *gap acceptance*.

Pedro Caetano (2011), em “*Propostas de Intervenção para Carreiras de Transportes Coletivos em Troços de Reduzida Velocidade Comercial*” [24] aplica a simulação microscópica com o objetivo de avaliar quais os aspetos que mais influenciam a velocidade comercial das carreiras na Avenida de Moscavide, em Moscavide.

Gonçalo Ferreira (2010), em “*Estudo do desempenho de uma rotunda da Cidade de Aveiro usando a microssimulação de tráfego*” [25] efetua a análise do desempenho de uma rotunda da cidade de Aveiro, através do utilização de um simulador, e compara os resultados obtidos pelos métodos empíricos tradicionais com os de simulação, de acordo com um modelo construído para o efeito, num simulador microscópico de tráfego.

José Ferreira, (2008), em “*Aplicação de modelos de tráfego (Dracula) na avaliação dos impactos em redes rodoviárias urbanas resultantes das alterações nas infraestruturas*” [26] implementa o programa de simulação *Dracula* [27] no estudo do impacto de um novo arruamento na zona envolvente.

Cristina Vilarinho (2008), em “*Calibração de modelos Microscópicos de simulação de Tráfego em redes urbanas*” [28] estuda a aplicação de modelos microscópicos de tráfego em redes urbanas, com recurso ao modelo *Aimsun* [9], designadamente na seleção e quantificação dos parâmetros e respetivos critérios de calibração nos modelos.

António Vasconcelos (2004), em “*Modelos de atribuição/simulação de tráfego: o impacto na qualidade dos resultados de erros no processo de modelação*” [29] analisou o impacto potencial dos erros de modelação na qualidade dos resultados obtidos através do programa de simulação *Saturn* [30].

2. SIMULAÇÃO DE TRÁFEGO

2.1. ENQUADRAMENTO

Ao longo das últimas décadas, verificou-se o aparecimento de uma diversa sucessão de ferramentas de simulação de tráfego, fenómeno impulsionado por desenvolvimentos significativos nas teorias de tráfego e comportamento dos condutores, bem como pela evolução e massificação do poder de computação.

Estas ferramentas são utilizadas correntemente em engenharia de tráfego e planeamento de transportes, mantendo-se em progresso e atualização constantes. Ao oferecer vantagens face às metodologias tradicionais (análise empírica e ajuste de curvas) [6] tal como a possibilidade de estudar diversas alterações físicas na infraestrutura sem afetar o normal funcionamento do tráfego, as ferramentas de simulação contribuem para a resolução de problemas com maior segurança e um custo mais reduzido para a rede. Adicionalmente, a simulação de várias alternativas permite a identificação de debilidades nos conceitos empregues, facilitando a identificação da solução ideal que terá, previsivelmente, maior probabilidade de sucesso.

Alguns dos mais recentes programas de simulação de tráfego, como o *Vissim (PTV)*, *Aimsun (TSS)* e *Paramics (Quadstone)*, incluem interfaces gráficas sofisticadas que possibilitam a visualização em tempo real de operações de tráfego em redes completas e exibindo resultados numa diversa gama de indicadores de desempenho da rede. Estas funcionalidades representam uma evolução significativa face aos programas clássicos, que permitiam apenas a análise de uma interseção ou de outro elemento da rede, frequentemente sem *output* gráfico.

Na verdade, os programas de simulação tráfego têm uma longa história evolutiva. Os primeiros apareceram na investigação teórica de modelos de *car-following* [4] ao longo de um arco. Posteriormente, surgiram programas que permitem a simulação de interseções [31], com ou sem sinalização, tornando-se bastante populares na otimização de semáforos. Apenas recentemente nasceram programas [1][9] preparados para analisar uma rede completa, composta por geometrias diversificadas e infraestruturas de transporte complexas.

Na prática, os resultados obtidos através de modelos de simulação podem ser apresentados em reuniões coletivas, para entidades mais ou menos familiarizadas com a engenharia de transportes, possibilitando aos decisores e ao público em geral uma perceção dos efeitos operacionais de alterações futuras.

2.2. APLICAÇÕES DA SIMULAÇÃO NA ENGENHARIA DE TRANSPORTES

A simulação de tráfego integra um conjunto de metodologias de análise de tráfego que proporcionam um auxílio na avaliação de estratégias e soluções que respondam às necessidades do sistema de transporte de uma região, município, bairro, ou qualquer outra escala geográfica. Algumas das aplicações dos programas de simulação incluem [32]:

Gerir e monitorizar os níveis de serviço de vias de comunicação:

Com ferramentas de simulação é possível estimar indicadores de desempenho, tais como tempos de viagem, atrasos, densidade de tráfego e filas de espera, em função da procura na rede.

Comparação de geometrias a adotar em novas infraestruturas:

À escala microscópica, é possível prever os indicadores de desempenho de diversas geometrias nas fases de planeamento e projeto de uma nova infraestrutura. Adicionalmente, é exequível analisar esquemas de prioridades em infraestruturas com exigências multimodais como operações de transporte público, passagens de peões e vias reservadas.

Auxiliar a proposta de alterações em infraestruturas de transporte:

Entre simular o impacto da implementação de medidas de acalmia de tráfego ou de vias de alta ocupação, da alteração de políticas de estacionamento e da otimização de semáforos, os programas de simulação permitem testar diversas alternativas sem necessidade de afetar o tráfego antes da decisão final.

Avaliar e priorizar operações de manutenção:

A simulação de operações de manutenção em vias de comunicação permite a avaliação do impacto do corte de um arco ou da obstrução de uma via na restante rede, possibilitando a gestão cuidada da sequência de áreas a intervir e dos melhores horários para a atuação em cada uma delas.

Auxiliar o processo de tomada de decisões e divulgação pública:

Os programas de simulação permitem a apresentação visual de resultados, auxiliando a demonstração de soluções alternativas a decisores políticos de forma simplificada. Por outro lado, facilitam o *output* de animações 3D que servem de plataforma para a educação do público quanto às alterações a efetuar numa determinada área.

Para melhor descrever o modo como as ferramentas de simulação se integram na engenharia de transportes, é importante analisar a sua relação com as ferramentas de análise clássicas.

2.3. METODOLOGIAS DE ANÁLISE DE TRÁFEGO

De forma a contextualizar a utilização de ferramentas de simulação na análise de tráfego, a FHWA [32] divide as metodologias de análise em sete grupos: esboço, modelos de estimação de procura, metodologias analíticas, otimização de semáforos e os três principais tipos de simulação, macroscópica, mesoscópica e microscópica. Para cada um destes grupos, é avaliada a legitimidade da sua utilização no contexto de planeamento, projeto ou operação.

Tabela 2-1 – Relevância da utilização de ferramentas de análise de tráfego. (traduzido de FHWA [32]).
(✓ - Aplicável; ◇ - Ocasionalmente aplicável; ✘ - Geralmente não aplicável)

CONTEXTO DA ANÁLISE	MODELOS DE ANÁLISE / METODOLOGIAS					
	ESTIMAÇÃO DE PROCURA	ANALÍTICOS (HCM)	OTIMIZAÇÃO DE SEMÁFOROS	SIMULAÇÃO MACROSCÓPICA	SIMULAÇÃO MESOSCÓPICA	SIMULAÇÃO MICROSCÓPICA
PLANEAMENTO	✓	◇	✘	◇	◇	✘
PROJETO	◇	✓	✓	✓	✓	✓
EXECUÇÃO / OPERAÇÃO	✘	✓	✓	✓	✓	✓

2.3.1. MODELOS DE ESTIMAÇÃO DE PROCURA

Tratam-se de modelos matemáticos de previsão de procura da rede, com base nas condições atuais, em antevisões de crescimento económico e no conseqüente aumento de empregos e residências. Um exemplo é o modelo dos quatro passos [33], no qual se analisam a geração de viagens e a sua distribuição por vários destinos, a repartição modal e a afetação do tráfego à rede.

No entanto, os modelos de estimação de procura não foram criados para a avaliação de metodologias de gestão de tráfego como sistemas inteligentes de transporte (ITS) e não possuem a capacidade de avaliar características operacionais como velocidades, atrasos e filas de espera.

Assim, este tipo de modelo é útil sobretudo na fase de planeamento, para comparar estimativas de procura resultantes de soluções alternativas, geralmente à escala macroscópica.

2.3.2. METODOLOGIAS ANALÍTICAS (HCM)

Este tipo de ferramenta permite prever rapidamente a capacidade, densidade, velocidade, atrasos e filas de espera numa variedade de geometrias de elementos das vias de comunicação, recorrendo a procedimentos de cálculo validados no terreno, em condições reais ou em experiências a escala reduzida. Um exemplo popular são as metodologias propostas pelo *Highway Capacity Manual (HCM)* [6].

As metodologias analíticas são adequadas à análise do desempenho de geometrias isoladas, porém não têm em conta o efeito de cada infraestrutura na restante rede.

2.3.3. OTIMIZAÇÃO DE SEMÁFOROS

As ferramentas de otimização de semáforos servem principalmente para ajustar a temporização de semáforos numa interseção, em função do desempenho resultante na restante rede. Este tipo de ferramenta é útil na fase de projeto e na fase de operação, sendo habitualmente acompanhada por outra ferramenta de análise de tráfego.

2.3.4. SIMULAÇÃO DE TRÁFEGO

A simulação de tráfego consiste na modelação matemática de sistemas de transporte [34], tais como convergências em autoestradas, rotundas, redes urbanas ou interfaces multimodais, através de um programa informático. Tem como finalidade auxiliar o planeamento, projeto e operação de sistemas de transporte, possibilitando a avaliação de modelos demasiado complexos para ferramentas analíticas e produzindo resultados numéricos e visuais a partir dos quais é executável a identificação de relações de causalidade que seriam impossíveis de observar por outro meio.

Tratando-se de modelos matemáticos, os modelos de simulação variam na escala ou escopo da análise (macro, meso (médio) e micro), na escala temporal (estáticos ou dinâmicos), na resolução dos objetos simulados (contínuos ou discretos) e na natureza das variáveis do sistema (determinísticas ou estocásticas) [35].

Assim, modelos de simulação macroscópica são geralmente estáticos (calculados para as condições de equilíbrio e invariáveis no tempo), contínuos (modelando o tráfego na rede de forma agregada ou como um fluido) e determinísticos (sem aleatoriedade no sistema, ou seja, as mesmas condições iniciais implicam sempre o mesmo resultado).

Por sua vez, os modelos microscópicos são geralmente dinâmicos (evoluindo no tempo), discretos (modelando individualmente cada agente na rede) e estocásticos (integrando variáveis aleatórias que simulam a variabilidade e imprevisibilidade de sistemas de transporte).

2.4. ESCALAS DE SIMULAÇÃO DE TRÁFEGO

As três escalas de simulação predominantes são a macroscópica, mesoscópica e microscópica [36], com o surgimento de variações como a simulação híbrida e nanoscópica. Estas diferem em dimensão e detalhe, pelo que exigem dados diferentes e produzem resultados distintos.

À escala macroscópica, o tráfego é analisado como uma carga ou fluido na rede. À escala mesoscópica o tráfego é abordado em pelotões. Finalmente, à escala microscópica, cada agente é modelado individualmente.

Nesta secção, descrevem-se sucintamente cada uma das escalas de simulação e o seu contexto de aplicação.

2.4.1. SIMULAÇÃO MACROSCÓPICA

Os modelos macroscópicos fundamentam-se na existência de relações determinísticas entre fluxo, velocidade e densidade do tráfego, ou seja, para uma gama de condições iniciais constantes como os fluxos, classe hierárquica e velocidade máxima de cada arco, obtém-se sempre o mesmo resultado, não existindo aleatoriedade no cálculo de atrasos na rede ou na previsão de congestionamentos [37].

Em comparação com modelos microscópicos, a simulação macroscópica requer consideravelmente menor capacidade computacional, mas não permite analisar tão detalhadamente as alterações introduzidas pelo utilizador na rede de transportes.



Figura 2.1 – Simulação macroscópica (Representação da carga na rede produzida em QGIS)

2.4.2. SIMULAÇÃO MESOSCÓPICA

Os modelos de simulação mesoscópica combinam propriedades da simulação macro e microscópica. Tal como na última, a unidade de tráfego representada é o veículo individual. No entanto, os veículos são agregados em células ou pelotões de tráfego e o seu movimento segue a abordagem aplicada nos modelos macroscópicos, sendo determinado pela velocidade média no arco respetivo, de acordo com capacidades e funções velocidade-densidade pré-definidas.

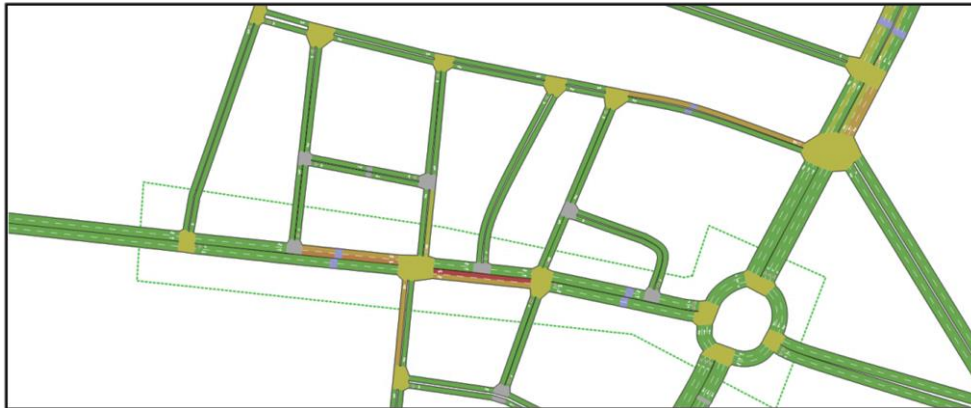


Figura 2.2 – Simulação mesoscópica (Simulação produzida em Aimsun)

2.4.3. SIMULAÇÃO MICROSCÓPICA

Estes modelos simulam o movimento dos veículos individualmente, com base em modelos comportamentais dos condutores como *car-following* e *lane changing*. Tipicamente, os veículos são introduzidos na rede afetados de variáveis aleatórias que modelam a variabilidade no comportamento ou personalidade dos condutores [38]. A posição de cada veículo é acompanhada e atualizada no espaço e no tempo em função das condições envolventes, pelo que as necessidades computacionais são bastante superiores, face às restantes escalas.



Figura 2.3 – Simulação microscópica (Simulação produzida em Aimsun)

2.4.4. SIMULAÇÃO HÍBRIDA

Nos modelos de simulação híbrida, interligam-se as escalas de simulação mesoscópica e microscópica.

Em função das necessidades ou dos dados disponíveis, uma parte da rede pode ser modelada à escala microscópica, enquanto a restante rede pode ser modelada com uma escala menos detalhada. Esta metodologia permite a análise de áreas extensas sem necessidade de elevadas capacidades computacionais, mantendo um nível de grande detalhe nas áreas de maior interesse, sem perda de sensibilidade quanto ao estado da restante rede.

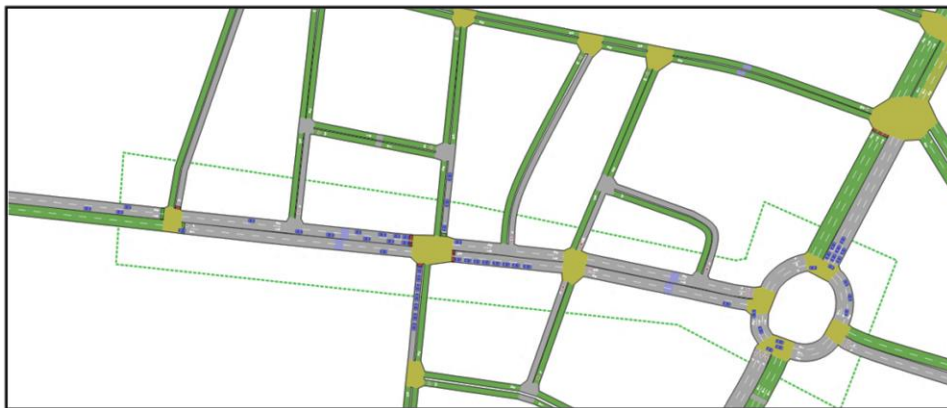


Figura 2.4 – Simulação híbrida (Simulação produzida em Aimsun)

2.4.5. SIMULAÇÃO NANOSCÓPICA

A simulação nanoscópica consiste na modelação da viagem de cada indivíduo, acompanhando todos os modos de transporte que este utilizar. Esta abordagem permite a medição de indicadores de desempenho e a determinação de custos em cada parte do trajeto, separadamente [39].

Torna-se viável determinar, para cada indivíduo, o tempo, custo e distância percorrida em transporte individual, transporte público e em modo pedonal. Paralelamente, é possível simular o tempo decorrido à espera de transporte público e o preço de estacionamento.

Este tipo de abordagem designa-se “avaliação económica porta-a-porta”, na qual o custo é baseado em todas as partes da viagem, independentemente do modo de transporte.

2.4.6. CONTEXTO DE APLICAÇÃO DAS ESCALAS DE SIMULAÇÃO

A seleção da escala de simulação a adotar em determinado projeto é, por vezes, determinada por fatores externos ao contexto técnico do caso particular, tais como a experiência do corpo técnico, o tempo disponível para o desenvolvimento do estudo e os recursos já adquiridos pela entidade. No entanto, a qualidade dos resultados que se obtêm com ferramentas de simulação é fortemente dependente da sua adaptabilidade à escala que se pretende analisar e aos dados disponíveis para *input* no programa.

Adicionalmente, o aumento do detalhe da simulação acarreta elevação do custo envolvido e, regra geral, pode prever-se um aumento de dez vezes o custo entre cada escala de simulação, sendo a microscópica a que implica um custo mais elevado [40].

Portanto, é importante refletir antecipadamente sobre a escala que melhor se adapta às características do estudo, aos recursos disponíveis e às necessidades do projeto.

2.4.6.1. CARACTERÍSTICAS DO ESTUDO

De um modo geral, modelos microscópicos exigem maior detalhe nos dados disponíveis ou a recolher, na quantidade de arcos a modelar e na variabilidade do tráfego ao longo do dia [40], pelo que a simulação de grandes áreas a esta escala se torna demasiado dispendiosa.

Tabela 2-2 – Aplicabilidade de escalas de simulação em função das características gerais do estudo.

(✓ - Aplicável; ◇ - Ocasionalmente aplicável; ✗ - Geralmente não aplicável) [40]

CARACTERÍSTICA	CRITÉRIO	MACROSCÓPICO	MESOSCÓPICO	MICROSCÓPICO
ESCALA	REGIONAL	✓	◇	✗
	CORREDOR	✓	✓	◇
	PEQUENA ÁREA	◇	✓	✓
DIMENSÃO DA REDE	GRANDE (> 10000 ARCOS)	✓	◇	✗
	MEDIO (≈ 5000 ARCOS)	✓	✓	◇
	PEQUENO (<1000 ARCOS)	✓	✓	✓
PERÍODO	24 HORAS	✓	◇	✗
	6 HORAS	✓	✓	◇
	PERÍODO DE PONTA	✓	✓	✓
	HORA DE PONTA	◇	✓	✓
PROCURA	GRANDE (> 1 MILHÃO DE VEÍC.)	✓	◇	✗
	MEDIO (≈ 500 MIL VEÍC)	✓	✓	◇
	PEQUENO (< 200 MIL VEÍC.)	✓	✓	✓

Certos projetos podem ainda beneficiar de uma análise híbrida, em que se analisa uma parte da rede à escala mesoscópica e outra parte à escala microscópica.

2.4.6.2. RECURSOS DA ORGANIZAÇÃO

É importante ter em consideração a experiência prévia e a dimensão do corpo técnico da entidade, em função do tempo disponível para o desenvolvimento do estudo.

Tabela 2-3 – Aplicabilidade de escalas de simulação em função dos recursos da organização.

(✓ - Aplicável; ◊ - Ocasionalmente aplicável; ✖ - Geralmente não aplicável) [40]

RECURSO	CRITÉRIO	MACROSCÓPICO	MESOSCÓPICO	MICROSCÓPICO
EXPERIÊNCIA PRÉVIA EM SIMULAÇÃO	MICROSCÓPICA	◊	✓	✓
	MESOSCÓPICA	◊	✓	◊
	MACROSCÓPICA	✓	◊	◊
DIMENSÃO DA EQUIPA	MEDIA (> 2)	✓	✓	✓
	PEQUENA (< 2)	✓	✓	◊
PRAZO DE ENTREGA	GRANDE (> 12 MESES)	✓	✓	✓
	MEDIO (4 A 12 MESES)	✓	✓	◊
	PEQUENO (< 4 MESES)	✓	◊	✖

2.4.6.1. FUNCIONALIDADES PRETENDIDAS

Diferentes programas comerciais oferecem funcionalidades distintas e a escala de simulação para a qual estão orientados é um fator determinante na disponibilidade de elementos de análise. Por exemplo, a presença de sinalização e semaforização é expectável em modelos microscópicos e menos prática em modelos macroscópicos. Analogamente, a análise de entrecruzamentos, convergências e filas de espera não é viável às escalas macroscópica e mesoscópica.

Tabela 2-4 – Aplicabilidade de escalas de simulação em função das funcionalidades pretendidas.

(✓ - Aplicável; ◊ - Ocasionalmente aplicável; ✖ - Geralmente não aplicável) [40]

FUNCIONALIDADE	MACROSCÓPICO	MESOSCÓPICO	MICROSCÓPICO
ANIMAÇÃO	✖	✓	✓
ENTRECRUZAMENTOS E CONVERGÊNCIAS	✖	✖	✓
FILAS DE ESPERA	✖	◊	✓
MODELAÇÃO AO ARCO	✓	✓	✓
MODELAÇÃO À VIA	✖	✖	✓
SINALIZAÇÃO	✖	◊	✓
SEMAFORIZAÇÃO	✖	◊	✓

2.5. FERRAMENTAS DE SIMULAÇÃO DE TRÁFEGO

A popularização da simulação de tráfego como ferramenta de análise levou ao aparecimento de diversos programas comerciais e é improvável que um único preencha todos os requisitos da entidade e dos projetos a desenvolver. Um dos fatores que mais diferencia as funcionalidades de programas comerciais é a escala da simulação (abordada na secção anterior), mas outros fatores como o tipo de resultados exportados, a interoperabilidade com outras ferramentas de simulação e o licenciamento, que devem ser considerados na escolha de um programa específico.

Limitações na modelação da rede:

É importante avaliar a adaptação de um programa à análise de redes urbanas ou autoestradas. Os modelos matemáticos de comportamento dos condutores implementados em cada programa de simulação têm vantagens e limitações relativas ao ambiente que se pretende simular.

Por outro lado, certos programas estão originalmente preparados para lidar com infraestruturas específicas como vias de alta ocupação, parqueamentos e praças de portagens, enquanto outros apenas possuem ferramentas que dão liberdade suficiente para modelar situações similares. A existência de funções dedicadas pode ser fundamental quando o tempo disponível para a realização do estudo é um fator crítico.

Agentes possíveis de modelar:

O realismo de um modelo, nomeadamente em ambiente urbano, não está apenas dependente do tráfego ligeiro. Ainda que grande parte dos programas disponíveis prevejam a presença de tráfego pesado na rede, a influência das operações do transporte público na rede, como a paragem de autocarros para a entrada e largada de passageiros, pode alterar profundamente o desempenho da simulação. Assim, é relevante não só a apreciação da disponibilidade de agentes de transporte público rodoviário e ferroviário, mas também a presença de peões e ciclistas na rede.

Controlo de tráfego:

A modelação de medidas de controlo de tráfego pode ser feita na forma de otimização de semáforos, na modelação de prioridades, na presença de sinalização vertical ou painéis de mensagem variável [11].

Muitos programas oferecem estas capacidades de origem, mas certos contextos podem exigir a operacionalidade com modelos externos de otimização semaforica ou a programação de comportamentos em condições particulares.

Afetação de tráfego:

A introdução de dados de tráfego na rede deve ser possível na forma como a entidade pretende efetuar a sua recolha no terreno, ou em função do tipo de dados já disponíveis.

A afetação de tráfego pode ser feita com base em matrizes OD (Origem/Destino) e, quando for relevante a modelação exata dos movimentos dentro da rede, complementada com uma afetação ao nível do arco ou mesmo da via.

Por um lado, pode ser importante a afetação estática do tráfego com base numa sequência de nós ou arcos fixa. Este tipo de afetação depende de contagens de veículos em cada interseção e produz resultados mais próximos da realidade.

A afetação dinâmica, por sua vez, permite a afetação do tráfego com base em origens e destinos, como uma área da rede ou um estacionamento. Retirando alguma liberdade da modelação exata do tráfego, permite a sua variabilidade no tempo, constituindo uma ferramenta útil na previsão de condições futuras.

Resultados:

A expectativa do cliente quanto aos resultados que pretende obter é fundamental na escolha do programa a utilizar numa simulação. É conveniente avaliar a possibilidade de contar volumes tráfego em determinados locais, a contagem de tempos de espera e a medição de filas de espera oferecidas por cada programa.

Por outro lado, a estrutura como estes resultados são disponibilizados pode ser relevante quando se pretende efetuar análises posteriores à simulação. A análise de dados ao nível do arco pode não ser comparável com dados ao nível da interseção, limitando a avaliação comparativa com outros programas de simulação.

Enquanto determinados programas permitem de origem a análise de parâmetros ambientais como emissões de CO₂ ou consumos de combustível, outros implicam a utilização de *plugins* que implicam um licenciamento adicional.

Animação:

Um dos fatores mais emblemáticos da utilização de ferramentas simulação é a possibilidade de produzir animações 2D ou 3D, tanto para a análise do desempenho do modelo como para fins comerciais, como a divulgação pública de alterações numa rede ou a demonstração de soluções alternativas a decisores políticos e outras entidades não familiarizadas com engenharia de tráfego. Atualmente, todos os programas de simulação exportam animações, mas é importante a avaliação da qualidade gerada por cada um e a possibilidade de introduzir elementos estéticos no modelo.

2.5.1. PROGRAMAS COMERCIAIS DE SIMULAÇÃO MICROSCÓPICA

Cada programa de simulação no mercado oferece vantagens e desvantagens. De um modo geral, os fabricantes de programas comerciais oferecem funcionalidades semelhantes, nomeadamente aqueles em desenvolvimento permanente como a *TSS (Aimsun)*, *Citilabs (Dynasim)*, *Quadstone (Paramics)*, *Caliper (TransModeler)* e *PTV (Vissim)*. Alguns destes fabricantes oferecem ferramentas que funcionam a várias escalas de simulação (*Aimsun* e *TransModeler*) o que pode ser vantajoso em entidades que efetuem estudos em variados âmbitos. Outros fabricantes constroem ferramentas distintas para cada escala, o que implica a obtenção de licenças adicionais (*PTV* e *Citilabs*).

Surgem também ferramentas *open-source* como o *SUMO* e o *MATSim* que conferem liberdade total na implementação de funcionalidades, mas exigem conhecimentos de programação inteiramente dispensáveis nos programas comerciais. Em mercados relativamente pequenos como o da simulação de tráfego, o desenvolvimento de ferramentas *open-source* torna-se menos competitivo face a ferramentas comerciais, mas a sua aplicabilidade, designadamente no meio académico, deve ser mencionada.

Tabela 2-5 – Listagem de aplicações comerciais em desenvolvimento ativo, por escopo da simulação e respetiva tipologia de licenciamento. (FOSS – Free, open-source software; português: programa grátis de código aberto)

APLICAÇÃO	MICROSCÓPICO	MESOSCÓPICO	MACROSCÓPICO	HÍBRIDO	LICENÇA
<i>AIMSUN</i>	✓	✓	✓	✓	Comercial
<i>CITYTRAFFICSIMULATOR</i>	✓				FOSS
<i>COMMUTER</i>	✓				Comercial
<i>CORSIM</i>	✓				Comercial
<i>CUBE DYNASIM</i>	✓	<i>VOYAGER</i>	<i>AVENUE</i>		Comercial
<i>DRACULA</i>	✓				Comercial
<i>MATSIM</i>	✓	✓	✓		FOSS
<i>PTV VISSIM</i>	✓		<i>VISUM</i>		Comercial
<i>QUADSTONE PARAMICS</i>	✓	✓		✓	Comercial
<i>ROADTRAFFICSIMULATOR</i>	✓				FOSS
<i>SIDRA INTERSECTION</i>	✓				Comercial
<i>SIDRA TRIP</i>	✓				Comercial
<i>SUMO</i>	✓				FOSS
<i>SYNCHRO / SIMTRAFFIC</i>	✓		✓		Comercial
<i>TRANSMODELER</i>	✓	✓	✓		Comercial
<i>TRITONE SIMULAZIONE</i>	✓				Grátis

Efetuada uma comparação dos programas comerciais de maior popularidade no mercado, verifica-se um equilíbrio das funcionalidades disponíveis mencionadas no capítulo 2.5. Ainda que não exaustiva, é notório o acompanhamento da funcionalidade global dos programas na competição.

Tabela 2-6 - Comparação de funcionalidades de aplicações comerciais de microsimulação. (✓ - Dedicado; ◇ - Implementável via COM / API / Plugin; ✗ - Não implementável)

FUNCCIONALIDADES	VISSIM	AIMSUN	PARAMICS	TRANSModelER	DYNASIM
REDE					
AUTOESTRADA	✓	✓	✓	✓	✓
URBANO	✓	✓	✓	✓	✓
RAMP METERING	◇	✓	◇	✓	✓
ESTACIONAMENTO	✓	✓	✓	✓	✓
PORTAGENS	◇	✓	✓	✓	✓
VAO	◇	✓	✓	✓	◇
VIAS RESERVADAS	✓	✓	✓	✓	✓
IMPORTAÇÃO GIS / CAD	✓	✓	✓	✓	✓
AGENTES					
TP RODOVIÁRIO	✓	✓	✓	✓	✓
TP FERROVIÁRIO	✓	✓	✓	✓	✓
PEÕES	✓	✓	✓	◇	✓
CICLISTAS	✓	◇	◇	◇	✓
CONTROLO DE TRÁFEGO					
OTIMIZAÇÃO DE SEMÁFOROS	✓	✓	✓	✓	✓
PRIORIDADE AO TRANSPORTE PÚBLICO	✓	✓	✓	✓	✓
PAINÉIS DE MENSAGEM VARIÁVEL	◇	✓	✓	✓	✓
AFETAÇÃO					
AFETAÇÃO ESTÁTICA	✓	✓	✓	✓	✓
AFETAÇÃO DINÂMICO	✓	✓	✓	✓	✓
MATRIZES OD	✓	✓	✓	✓	✓
PROCURA VARIÁVEL NO TEMPO	✓	✓	✓	✓	✓
RESULTADOS					
DETETORES DE VEÍCULOS	✓	✓	✓	✓	✓
FILAS DE ESPERA	✓	✓	✓	✓	✓
EMISSIONES CO ₂	◇	✓	✓	✓	✗
CONSUMO DE COMBUSTÍVEL	◇	✓	✓	✓	✗
ANIMAÇÃO					
2D E 3D	✓	✓	✓	✓	✓
OUTRO					
PACOTE DE APLICAÇÕES	PTV VISTRO	---	---	CALIPER	CUBE

2.6. DESENVOLVIMENTO DE MODELOS DE SIMULAÇÃO DE TRÁFEGO

Os modelos de simulação de tráfego são significativamente mais complexos e sensíveis a pequenos detalhes dos sistemas de transporte e procura na rede do que as metodologias de análise de tráfego tradicionais [40].

Assim, no desenvolvimento de um modelo de simulação, deverão ser tomados cuidados consideráveis na preparação da rede, na introdução da procura de tráfego e na calibração do modelo.

2.6.1. METODOLOGIA

Na inexistência prévia de um guia ou manual que aborde genericamente os procedimentos para o desenvolvimento de um modelo de simulação, a *Federal Highway Administration* [11] propõe sete tarefas fundamentais:

1. Definição do âmbito da análise;
2. Recolha e preparação de dados;
3. Desenvolvimento do modelo;
4. Verificação de erros;
5. Calibração do modelo;
6. Análise de soluções e comparação de resultados
7. Relatório final e documentação técnica

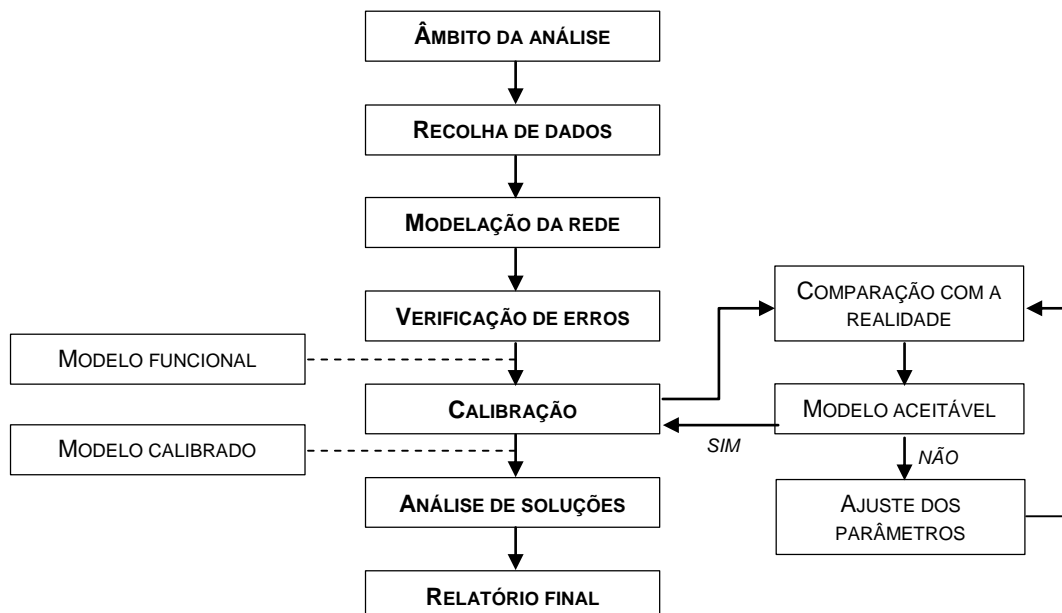


Figura 2.5 – Metodologia de desenvolvimento de modelos de simulação. (adaptado de FHWA [53])

2.6.2. DEFINIÇÃO DO ÂMBITO DA ANÁLISE

O primeiro passo no desenvolvimento de um modelo de simulação é a definição do âmbito ou escopo da escala de simulação pretendida, em função do qual é possível definir o programa a implementar e avaliar os recursos e experiência necessários. Nesta fase, é importante alocar o tempo necessário para o desenvolvimento do estudo e planejar todo o procedimento.

2.6.3. RECOLHA E PREPARAÇÃO DE DADOS

Modelos de simulação, particularmente microscópica, requerem uma grande quantidade de informação. A qualidade, detalhe e consistência dos dados disponíveis são de grande importância para a validade do modelo e é imprudente assumir que estão livres de erros, independentemente da fonte. Pequenas inconsistências podem não ter qualquer impacto à escala macroscópica, mas alterar profundamente os resultados obtidos à escala microscópica.

Geometria da rede

Dependendo do programa em utilização, a geometria da rede pode ser obtida através de ficheiros de sistemas de informação geográfica (*SIG*), desenhos de projeto (*CAD*), fotografia aérea ou levantamento no terreno. Em função da fonte, a geometria pode ser desenhada diretamente no programa (se este o permitir) ou importada diretamente.

Em todo o caso, é importante verificar a atribuição do número de vias, o comprimento e sentido dos arcos, principalmente quando a rede tem origem num modelo macroscópico, onde certos detalhes são irrelevantes.

Controlo de tráfego

É imperativo efetuar o levantamento da sinalização existente, temporização de semáforos e elementos como passagens de peões, paragens e linhas transporte público, vias reservadas, etc.

Procura existente

Os dados de procura consistem geralmente em volumes de entrada na rede e viragens em cada interseção [11]. Dependendo do programa, poderá ser vantajosa a construção de uma matriz OD para efeitos de validação, ou como fonte de primária de afetação.

Quando for relevante, o levantamento deverá ser registado em função da hora do dia para permitir a afetação dinâmica de tráfego à rede.

As contagens deverão ser sempre efetuadas em função do tipo de veículo (ligeiro, pesado, *BUS*).

Dados para calibração

O processo de calibração de modelos implica a comparação dos resultados da simulação para as condições atuais com os dados recolhidos no terreno [11]. Deste modo, é importante recolher uma amostra de tempos de viagem, fluxos em determinados arcos e filas de espera que sirvam de referência ao longo do processo de calibração.

2.6.4. DESENVOLVIMENTO DO MODELO

A cada programa de simulação corresponde uma metodologia de modelação da rede. Certos programas utilizam um sistema de arcos e nós, enquanto outros utilizam um sistema de arcos e conetores. Em ambos os casos, a construção da rede é um processo moroso e exigente.

Um modelo começa pela camada base com o diagrama da rede. Nesta é modelada a geometria dos arcos e interseções (nós/conetores), o número de vias, o tipo de via (urbana/autoestrada), restrições de acesso e vias reservadas.

A segunda camada inclui os atributos de controlo de tráfego como sinalização (vertical estática ou variável) e horizontal, controlo de prioridades, limites de velocidade e, quando aplicável, zonas de obra ou acidente.

Finalmente, é introduzido e modelado o tráfego na rede. Este passo inclui a definição dos pontos de entrada na rede, dos movimentos de viragem e das composições de veículos.

2.6.5. VERIFICAÇÃO DE ERROS

Após a modelação é prudente efetuar uma verificação de potenciais erros na rede e proceder à sua correção para que não interfiram com o processo de calibração da mesma, dificultando a obtenção de um modelo válido.

Estes erros aparecem na forma de restrição de vias mal atribuídas, sinalização vertical que é ignorada em zonas grande densidade de arcos ou prioridades mal definidas.

Uma inspeção visual do modelo em execução pode despistar grande parte dos erros grosseiros na rede.

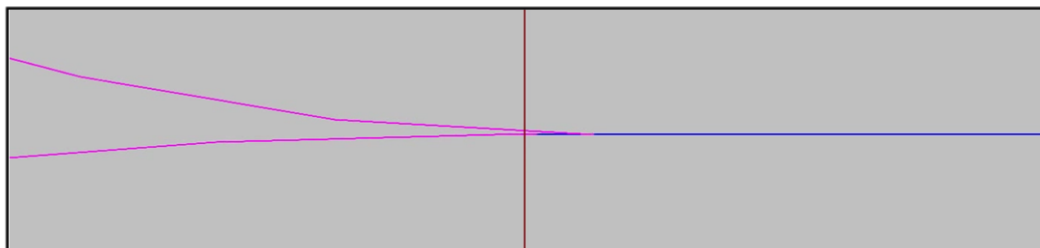


Figura 6 – Exemplo de erro na colocação de sinalização vertical em Vissim. Neste caso, a implantação acidental do elemento no conector provoca que a sinalização não seja respeitada por todos os agentes.

2.6.6. CALIBRAÇÃO DO MODELO

Cada programa de simulação oferece um conjunto de parâmetros que podem ser ajustados pelo utilizador para calibrar o comportamento dos condutores de acordo com as condições locais, uma vez que é impossível prever toda a variabilidade de comportamentos possíveis de modelar. Para além dos parâmetros de comportamento, poderão ser ajustados valores como a procura em cada arco, composições de veículos, distribuições de peso e potência.

A calibração de um modelo é um processo iterativo e exige um utilizador experiente e com algum conhecimento teórico sobre os modelos de comportamento utilizados pelo programa [11]. Em cada iteração, alteram-se ligeiramente os parâmetros necessários até que as condições simuladas reflitam as condições reais da rede.

É um processo moroso para o qual não deve ser menosprezada a alocação de tempo suficiente. Adicionalmente, deve ser bem documentado para que de futuro seja possível entender os ajustamentos efetuados.

2.6.7. ANÁLISE DE SOLUÇÕES E COMPARAÇÃO DE RESULTADOS

Após a calibração do modelo está-se em condições de o aplicar na análise de diferentes soluções. Definem-se pontos de medição de indicadores de desempenho da rede em locais que não sejam afetados por alterações de geometria e corre-se o modelo durante o tempo de simulação relevante.

Cada solução deve ser corrida várias vezes para que se obtenha uma amostra sólida dos indicadores de desempenho. Alguns programas permitem a introdução de uma variável que afeta a aleatoriedade na definição do comportamento dos condutores para garantir que os resultados que se obtém não são fruto de uma sequência de condutores que é repetida de cada vez que o modelo é corrido.

Finalmente, comparam-se os resultados obtidos em cada alternativa para avaliar o benefício obtido nas condições da rede.

2.6.8. RELATÓRIO FINAL E DOCUMENTAÇÃO TÉCNICA

Independentemente do contexto em que o estudo seja realizado, a documentação do processo e dos resultados obtidos não deve ser menosprezada. A grande quantidade de dados resultantes da simulação poderá ser difícil de entender por terceiros ou, numa futura análise, pelo próprio modelador da rede, pelo que é importante documentar as deliberações tomadas na modelação da geometria da rede e na calibração dos agentes de tráfego, os objetivos da decisão e as suas consequências.

2.7. LIMITAÇÕES DA SIMULAÇÃO DE TRÁFEGO

As ferramentas de simulação de tráfego são úteis e eficazes no apoio à engenharia de transportes, mas são bastante exigentes no que diz respeito à qualidade dos dados disponíveis e da experiência necessária para a sua utilização.

Disponibilidade e consistência de dados:

O levantamento de dados no terreno pode ter um custo bastante elevado e é indispensável no caso da simulação microscópica. Por outro lado, dados adquiridos a outra entidade ou para outro fim podem não ser consistentes ou serem detalhados o suficiente para a construção de um modelo de simulação.

Disponibilidade de fundos:

A necessidade de adquirir um programa comercial pode ser um fator limitativo. O orçamento disponível pode não cobrir a aquisição da ferramenta certa e a obtenção do treino necessário, levando à utilização de um programa já adquirido, mas menos adaptado à situação em estudo. Adicionalmente, dependendo do programa e da complexidade do modelo, uma simulação poderá levar entre alguns minutos a várias horas, pelo que é importante a disponibilidade de um equipamento adequado para correr a simulação.

Disponibilidade de recursos humanos:

Um corpo técnico com o treino e experiência apropriados é importante para a validação de um modelo de simulação, mas certas entidades poderão não ter elementos adequados por serem compostas por um número reduzidos de elementos ou por estes serem especializados noutra escala de simulação.

Adequabilidade do programa de simulação:

Nenhum programa comercial está, à partida, adequado para modelar todas as situações que se pretenda analisar. Estas limitações são mais notáveis quando se pretende modelar condições complexas como praças de portagem, parqueamentos, praças de táxi, bombas de combustível ou vias de alta ocupação. Ainda que muitas vezes seja possível contornar as limitações utilizando outras funcionalidades do programa como, por exemplo, sinais de *STOP* com um tempo de paragem pré-definido para modelar pórticos de portagem, é desejável a utilização de uma funcionalidade dedicada para limitar a oportunidade de comportamentos imprevisíveis.

Finalmente, existem ainda algumas condições que a generalidade dos programas não aborda como a interação de veículos e ciclistas na mesma via, o efeito do estacionamento ilegal em zonas urbanas e a modelação de arcos com dois sentidos em zonas rurais.

2.8. COMPARAÇÃO DE MODELOS DE SIMULAÇÃO COM O *HCM*

O *Highway Capacity Manual* é a ferramenta analítica clássica aceite para diversas aplicações na análise de tráfego como a determinação de níveis de serviço e geometria de interseções, pelo que é importante referir o modo como se relaciona com as ferramentas de simulação abordadas neste documento.

De facto, tanto o *HCM2010* [6] como *FHWA* [11] já fazem referência ao uso de ferramentas alternativas como *softwares* de simulação, comparando as vantagens de cada abordagem e as diferenças conceptuais da sua utilização.

Por outro lado, surgem alguns estudos que efetuam uma comparação prática dos resultados obtidos com a utilização dos dois tipos de ferramentas [21] [41] e tentam entender a forma como se relacionam, evitando responder à questão de qual das ferramentas é a melhor ou a mais adequada. No entanto, o consenso é de que, na sua forma atual, as ferramentas de simulação não têm uma ponte com as metodologias propostas no *HCM* e existe ainda algum caminho a percorrer na sua interligação [41].

2.8.1. VANTAGENS DOS PROCEDIMENTOS DO *HCM*

As metodologias propostas no *HCM* são úteis na análise de elementos isolados da rede de forma rápida e coerente para prever o nível de serviço da infraestrutura, implementando formulações testadas exaustivamente no terreno e em condições reais.

Adicionalmente, as metodologias do *HCM* consideram explicitamente pormenores geométricos (largura da berma), a influência das condições atmosféricas, condições de luminosidade, zonas de obra e acidentes.

Finalmente, o procedimento de cálculo de indicadores de desempenho como velocidade praticada e densidade é transparente no *HCM*, enquanto os resultados produzidos por ferramentas de simulação está dependente das considerações tomadas pelo autor do programa que nem sempre são de conhecimento público.

2.8.2. DESVANTAGENS DOS PROCEDIMENTOS DO *HCM*

A maioria das metodologias do *HCM* assume que a operação de uma interseção ou segmento não é afetado pelas condições das interseções adjacentes, perdendo fiabilidade em zonas urbanas onde as filas de espera se podem estender entre duas ou mais interseções.

O *HCM* está preparado para a análise de elementos isolados e não considera a utilização de rotas alternativas para o cálculo de tempos de viagem.

Finalmente, existe uma gama limitada de geometrias possíveis de analisar com o *HCM*, resultando em várias situações reais sem correspondência no manual.

2.8.3. DIFERENÇAS NOS INDICADORES DE DESEMPENHO DA REDE

As metodologias propostas no *HCM* tomam uma abordagem estática na previsão do desempenho da rede enquanto os modelos de simulação têm uma abordagem dinâmica. Por conseguinte, o *HCM* prevê a análise da rede em diferentes condições de saturação e oferece metodologias de cálculo distintas para cada caso, enquanto os modelos de simulação modelam a rede no seu todo, seguindo cada veículo no espaço e no tempo e gerindo a ocorrência de filas de espera de forma realista.

Por outro lado, o *HCM* estima valores médios para a densidade, velocidade e atrasos nos 15 minutos de débito máximo da hora de ponta enquanto os modelos de simulação fazem uma previsão destes valores para o período da análise que for relevante.

Adicionalmente, é importante referir as diferenças nos conceitos e definições empregues por cada ferramenta:

- Os modelos de simulação exportam valores de densidade e fluxo em função do número de veículos reais simulados, enquanto o *HCM* emprega a unidade de veículo equivalente, equacionando veículos pesados no valor final;
- Nos modelos de simulação, os atrasos e filas de espera são refletidos nos arcos e na rede, enquanto o *HCM* considera todo o atraso causado por um determinado congestionamento, independentemente da posição física dos veículos.

No entanto, o *HCM2010* inclui algumas recomendações para o ajustamento dos indicadores de desempenho obtidos em simuladores de tráfego aos valores possíveis de obter pelo *HCM*.

É possível observar que não existe uma ponte direta entre o *HCM* e as ferramentas de simulação, uma vez que cada ferramenta usa conceitos base bastantes distintos e, fundamentalmente, tem objetivos e âmbitos de aplicação diferentes.

3. MODELOS COMPORTAMENTAIS DE TRÁFEGO

3.1. CAR-FOLLOWING

O comportamento de *car-following*, que descreve o modo como dois veículos interagem entre si, constitui um fator chave em modelos de simulação de tráfego, pelo que a sua compreensão é fundamental na avaliação do desempenho de um simulador.

A modelação do comportamento de determinado condutor pode ser influenciada por características individuais, designadamente idade, género, experiência de condução, tamanho e potência do veículo, grau de condução defensiva e fatores biopsicológicos, tais como distrações, estado de alcoolémia, condução sob o efeito de drogas, *stress*, fadiga, assim como o propósito e duração da viagem. Por outro lado, a condução é simultaneamente afetada por fatores externos, como o período do dia, dia da semana, as condições atmosféricas e o tipo de via.

Um modelo de simulação de tráfego é tão realista quanto maior for o seu detalhe relativamente à modelação do comportamento dos condutores. Diferentes modelos implementam diferentes aspetos com influência no condutor e, de uma forma global, cada programa comercial de simulação microscópica utiliza um modelo de *car-following* distinto daqueles da competição, pelo que a revisão dos modelos em prática na atualidade se revela interessante e vantajosa.

Assim, apresenta-se uma exposição resumida dos modelos de *car-following* que originaram os modelos atualmente aplicados em programas comerciais de simulação microscópica. Seguidamente, efetua-se uma revisão mais detalhada da formulação e procedimento lógico que dão origem aos modelos utilizados nos programas *Vissim*, *Aimsun* e *Paramics*.

Ao longo do capítulo são analisadas diversas condições de *car-following*, onde se utiliza o termo *veículo n* ou condutor *n* como referência ao veículo ou condutor em análise, e o termo *veículo n-1* ou condutor *n-1* designa o veículo imediatamente em frente, ou líder, de acordo com a figura seguinte.

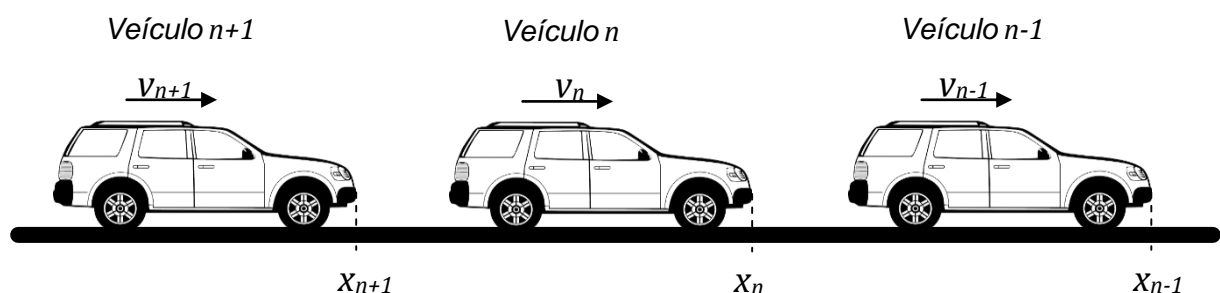


Figura 3.1 – Nomenclatura utilizada na descrição de agentes em modelos de *car-following*.

3.1.1. ENQUADRAMENTO

Os modelos de *car-following* podem ser categorizados em cinco grupos [42]:

- Modelo *Gazis-Herman-Rothery* (*GHR*) [12];
- Modelo de distância de segurança ou prevenção de colisão (*CA*) [13];
- Modelo linear (*Helly*) [15];
- Modelo psicofísico (*AP*) [43];
- Modelo de lógica difusa (*fuzzy logic*) [16];

3.1.1.1. MODELO GAZIS-HERMAN-ROTHERY (*GHR*)

Formulado em 1958 pela *General Motors*, o modelo *GHR* (3-1) é baseado na hipótese de que a aceleração de um condutor é proporcional à diferença de distância e velocidade entre o seu veículo e o veículo em frente.

$$a_n(t) = cv_n^m(t) \frac{\Delta v(t-T)}{\Delta x^l(t-T)} \quad (3-1)$$

O modelo determina a aceleração do veículo n no instante t em função da velocidade do veículo $n-1$ e da distância entre ambos. É tido em conta o tempo de reação T do condutor n e os parâmetros c , m e l são constantes de calibração que devem ser determinados para cada rede.

O *GHR* foi alvo de investigação nos quinze anos subsequentes ao seu desenvolvimento, nomeadamente no que diz respeito aos valores a adotar para os seus parâmetros de calibração m e l que variaram profundamente em cada estudo desenvolvido para encontrar os seus valores recomendados. Isto levou a que o estudo e uso do *GHR* viesse a perder interesse a partir dos anos setenta. [42]

3.1.1.2. MODELO DE DISTÂNCIA DE SEGURANÇA OU PREVENÇÃO DE COLISÃO (*CA*);

A formulação original deste tipo de modelo foi proposta em 1959 por *Komenati* e *Sasaki* [14]. A ideia base é procurar a especificação da distância de segurança para o veículo n (através das equações do movimento básicas), a partir da qual um acidente seria inevitável caso o veículo $n-1$ adote um comportamento imprevisto. A formulação original foi a seguinte:

$$\Delta x(t-T) = \alpha v_{n-1}^2(t-T) + \beta_l v_n^2(t) + \beta v_n(t) + b_0 \quad (3-2)$$

Com α , β , β_l e b_0 como constantes de calibração e T como o tempo de reação do condutor n .

O modelo *CA* foi posteriormente desenvolvido por *Gipps* (1981) [13], com a introdução de fatores de calibração que conduziram a resultados mais realistas, dando origem ao modelo de *Gipps* que é hoje utilizado em diversos programas de simulação [42] e é explorado em maior detalhe no capítulo 3.1.3 deste documento.

3.1.1.3. MODELO LINEAR (HELLY)

O modelo linear tem origem na formulação inicial do modelo *GHR* que era linear nas primeiras fases do seu desenvolvimento. No entanto, o modelo linear é habitualmente atribuído a *Helly* (1959) [15] que propôs a introdução de atributos para a modelação da distância de seguimento desejada pelo condutor do veículo n , em função do comportamento do veículo $n-1$.

$$a_n(t) = C_1 \Delta v(t - T) + C_2 (\Delta x(t - T) - D_n(t)), \quad (3-3)$$

Onde $D(t)$ é a distância de seguimento desejada:

$$D_n(t) = \alpha + \beta v(t - T) + \gamma a_n(t - T) \quad (3-4)$$

Com α , β , γ , C_1 e C_2 como constantes de calibração e T o tempo de reação do condutor n .

Apesar da evolução face ao modelo *GHR*, o modelo de *Helly* não ofereceu uma melhoria importante e atrativa para o seu uso em modelos de simulação [42].

3.1.1.4. MODELO PSICOFÍSICO (AP)

O modelo psicofísico tem como base a suposição de que um condutor n toma uma decisão e atua quando um determinado limiar, expresso em função da diferença de distância e velocidades entre n e $n-1$, é ultrapassado.

O modelo inicial foi definido por *Michaels* (1963) [43] e consiste nos seguintes limiares:

- Aproximação (*closing*), definido pela perceção de diferença de velocidades elevada;
- Seguimento (*following*), composto pelos limiares de aproximação e afastamento em diferenças de velocidades reduzidas;
- Limiares de perceção de diferenças de distância.

O modelo psicofísico tem o melhor desempenho na modelação do comportamento dos condutores, mas envolve uma série de parâmetros de calibração para os quais não é fácil obter valores conclusivos e é difícil a sua extração para diferentes ambientes culturais, sociais e climáticos. [42]

Nos anos seguintes, autores como *Wiedemann* (1974) [17], *Reiter* (1994) [44] e *Fritzsche* (1994) [18] desenvolveram adaptações e atualizações ao modelo base que são hoje utilizados em programas comerciais de simulação microscópica como o *Vissim* (*Wiedemann*) e o *Paramics* (*Fritzsche*).

3.1.1.5. MODELO DE LÓGICA DIFUSA

Um modelo de lógica difusa funciona na extensão da lógica booleana (VERDADEIRO - 1 ou FALSO - 0) com a inclusão da hipótese TALVEZ (0,5) e permite descrever a forma como uma variável se insere na descrição de um determinado termo. Um modelo desta natureza permite avaliar eventos e criar uma reação de operadores lógicos para modelar a tomada de decisões de condutores, por exemplo: SE 'PERTO' E 'A APROXIMAR' ENTÃO 'TRAVA' mas SE 'LONGE' E 'A APROXIMAR' ENTÃO 'SEMAÇÃO'.

Este tipo de lógica foi adaptada a modelos como o *GHR*, por *Kikuchi* e *Chakraborty* (1992) [16], e *Wiedemann '74* [17], por *Rekersbrink* (1995) [45] mas a sua utilização isolada não ganhou tração no mercado de aplicações comerciais.

3.1.2. MODELO DE WIEDEMANN '74

O modelo de *Wiedemann* opera simulando cada condutor individualmente com base em modelos de percepção humana e limites de reação, classificando-se como um modelo psicofísico de *car-following*. É o modelo de *car-following* utilizado pelo programa comercial *Vissim*, onde cada unidade condutor-veículo é descrito como um único elemento (*DVE – driver unit element*).

A suposição do modelo de *Wiedemann* é de que o condutor do veículo n opera em um de quatro estados de comportamento em relação aos quais este tem uma reação:

- *Free-driving* (condução livre / sem reação), onde não existe influência de nenhum veículo a jusante e o condutor tenta apenas atingir a sua velocidade desejada;
- *Approaching* (aproximação), em que o condutor de veículo n se apercebe conscientemente de que se está a aproximar do veículo $n-1$;
- *Following* (seguimento), que representa o estado em que o condutor do veículo n acelera e desacelera mantendo a sua distância ao veículo $n-1$ dentro dos limiares que definem a percepção de aproximação e afastamento do veículo em frente;
- *Breaking* (travagem), que ocorre quando a distância entre n e $n-1$ é inferior à distância de segurança, o que dá origem a uma travagem de emergência.

Um dos desafios de um modelo psicofísico está na definição dos limiares de percepção, os quais definem cada estado de comportamento em que o condutor se poderá encontrar.

No modelo de *Wiedemann '74*, estes limiares são função da distância e diferença de velocidades entre o veículo n e $n-1$, de determinados parâmetros de calibração que modelam o comportamento da generalidade dos condutores na rede e variáveis aleatórias que simulam a aleatoriedade da personalidade dos condutores.

Os limiares propostos por *Wiedemann* são:

- *AX*: a distância mínima (frente a frente) em situação de paragem;
- *ABX*: a distância de perseguição mínima desejada;
- *SDX*: a distância de perseguição máxima desejada;
- *SDV*: o limite no qual o condutor se apercebe de que se está a aproximar de um veículo mais lento;
- *OPDV*: o limite, numa situação de perseguição, para a diferença de velocidades em que o condutor se apercebe que se está a afastar do veículo em frente;
- *CLDV*: o limite, numa situação de perseguição, para a diferença de velocidades em que o condutor se apercebe que se está a aproximar do veículo em frente.

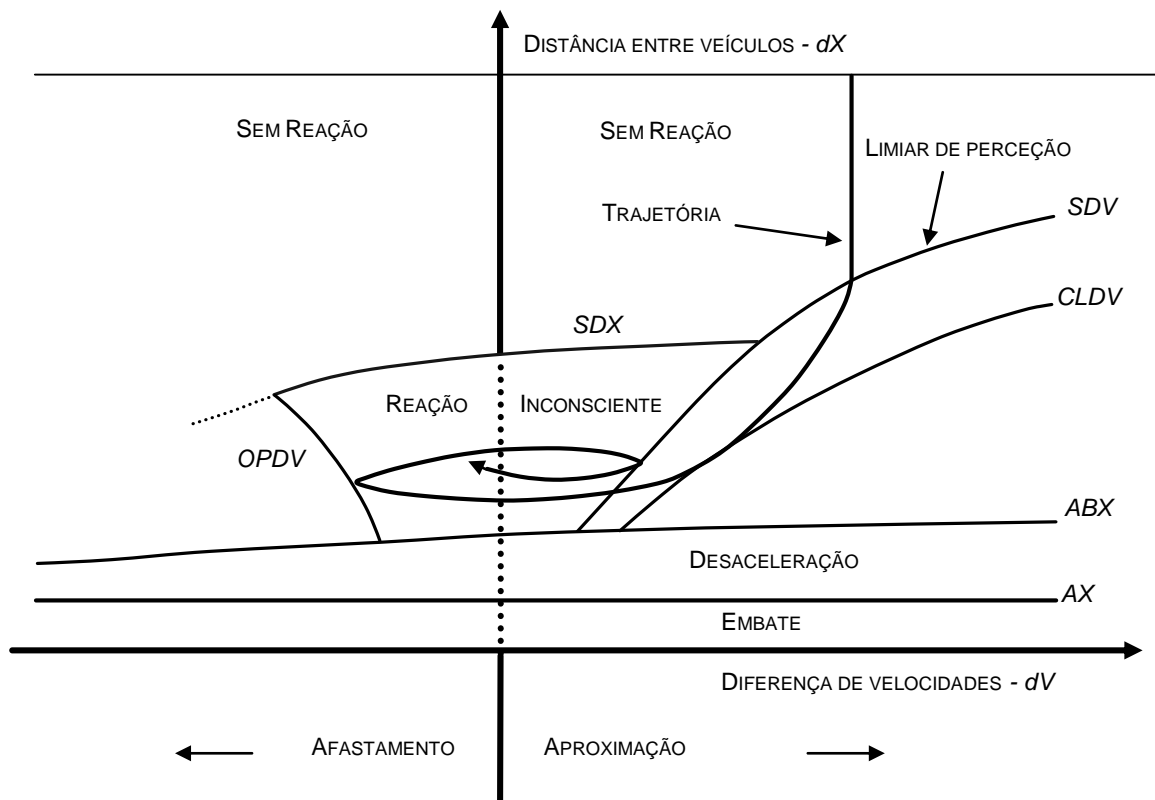


Figura 3.2 – Limites de reação do modelo de Wiedemann '74 [17]

A determinação dos limites de reação de cada condutor é feita com base nas equações propostas em *Wiedemann & Reiter, 1992* [17]:

AX:

A distância desejada entre veículos estacionários, AX , é um limiar composto pelo comprimento do veículo $n-1$ e pela distância frente-traseira desejada, com a formulação:

$$AX = L_{n-1} + AX_{add} + RND1 * AX_{mult}; \quad (3-5)$$

Onde AX_{add} e AX_{mult} são parâmetros de calibração e $RND1$ um valor aleatório obtido através da distribuição normal $N(0,5; 0,15)$, sugerida por *Wiedemann* [17].

ABX:

A distância mínima de seguimento a velocidades reduzidas, ABX :

$$ABX = AX + BX, \quad \text{com } BX: \quad (3-6)$$

$$BX = (BX_{add} + BX_{mult} * RND1) * \sqrt{v}; \quad (3-7)$$

Onde BX_{add} e BX_{mult} são parâmetros de calibração e a velocidade, v :

$$v = \begin{cases} v_{n-1} & \text{se } v_n > v_{n-1} \\ v_n & \text{se } v_n \leq v_{n-1} \end{cases} \quad (3-8)$$

SDV:

O limiar de aproximação, SDV , que descreve o ponto a partir do qual um condutor se apercebe de que se está a aproximar de um veículo mais lento, tem a seguinte formulação:

$$SDV = \left(\frac{\Delta x - L_{n-1} - AX}{CX} \right)^2, \quad \text{com } CX: \quad (3-9)$$

$$CX = CX_{const} * (CX_{add} + CX_{mult} * (RND1 + RND2)); \quad (3-10)$$

Onde CX_{const} , CX_{add} , e CX_{mult} são parâmetros de calibração e $RND1$ e $RND2$ valores aleatórios obtidos através da distribuição normal $N(0,5; 0,15)$, sugerida por *Wiedemann* [17].

SDX:

O valor máximo para a distância de seguimento, *SDX*, que varia entre 1,5 e 2,5 vezes a distância mínima de seguimento *ABX*, tem a definição:

$$SDX = AX + EX * BX, \quad \text{com } EX: \quad (3-11)$$

$$EX = EX_{add} + EX_{mult} * (NRND - RND2); \quad (3-12)$$

Onde *EX_{add}* e *EX_{mult}* são parâmetros de calibração e *RND2* e *NRND* valores aleatórios obtidos através da distribuição normal $N(0,5; 0,15)$, sugerida por *Wiedemann* [17].

CLDV:

O limiar *CLDV* representa o limite para a diferença de velocidades a partir do qual um condutor se apercebe que se está a aproximar de um veículo mais lento, a velocidades reduzidas. É semelhante ao limiar *SDV* e o *Vissim* ignora o seu cálculo, assumindo *CLDV* igual a *SDV*.

$$CLDV = SDV * EX^2; \quad (3-13)$$

OPDV:

O limiar *OPDV* representa o ponto em que um condutor se apercebe que está a circular a uma velocidade inferior ao veículo em frente, com a seguinte formulação:

$$OPDV = CLDV * (-OPDV_{add} - OPDV_{mult} * NRND); \quad (3-14)$$

Com *OPDV_{add}* e *OPDV_{mult}* como parâmetros de calibração e *NRND* um valor aleatório obtido através da distribuição normal $N(0,5; 0,15)$, sugerida por *Wiedemann* [17].

A cada estado psicofísico corresponde uma reação que é traduzida pela alteração da aceleração do veículo. O processo lógico para a avaliação do estado em que cada veículo se encontra é:

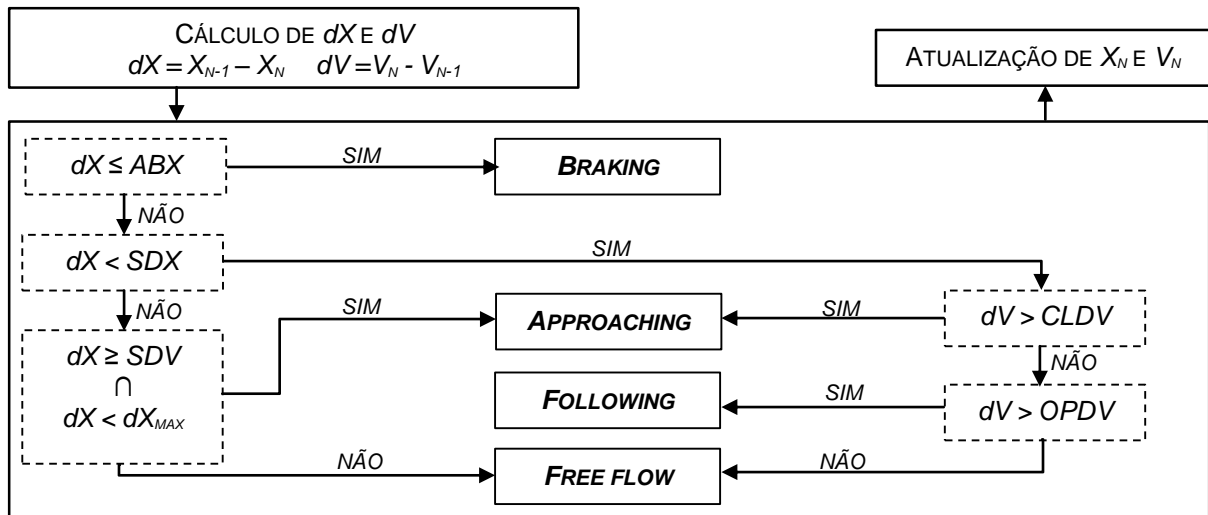


Figura 3.3 – Procedimento de decisão no modelo de Wiedemann '74 [54]

E a aceleração determina-se com base na correspondente equação proposta em Wiedemann & Reiter, 1992:

Braking:

Quando a distância frente-traseira é inferior a ABX , o DVE encontra-se no estado de travagem de emergência, cuja aceleração tem a seguinte formulação:

$$a_n = \frac{1}{2} \frac{dV^2}{ABX - \Delta x - L_{n-1}} + a(n-1) + \left(-20 + \frac{1,5}{60} * v_n\right) * \frac{ABX - \Delta x - L_{n-1}}{BX}; \quad (3-15)$$

Approaching:

Ultrapassando o limiar SDV , o condutor desacelera de modo a evitar uma colisão. O valor da aceleração a adotar pode ser definido por BN ou $BMIN$:

$$BN = \frac{1}{2} * \frac{dV^2}{ABX - \Delta x - L_{n-1}} + a_{n-1}; \quad (3-16)$$

$$BMIN = -20 + \frac{1,5}{60} * v_n; \quad (3-17)$$

De acordo com:

$$a_n = \begin{cases} BN, & \text{se } BN \leq BMIN; \\ BMIN, & \text{se } BN > BMIN; \end{cases} \quad (3-18)$$

Following:

Constituído pelos limites *SDV*, *SDX*, *OPDV* e *ABX* encontra-se o regime de seguimento. Quando um veículo ultrapassa os limiares *CLDV/SDV* ou *ABX* é-lhe atribuído a aceleração $-b_{null}$ e quando ultrapassa os limiares *OPDV* ou *SDX*, obtém o valor b_{null} .

$$b_{null} = b_{nullMult} * (RND3 + NRND); \tag{3-19}$$

$$a_n = \begin{cases} -b_{null}, & \text{se } dV > CLDV/SDV; \\ b_{null}, & \text{se } dV > OPDV; \end{cases} \tag{3-20}$$

Com $b_{nullMult}$ como um parâmetro de calibração e *NRND* e *RND4* um valor aleatório obtido através da distribuição normal $N(0,5; 0,15)$.

Free Flow:

No regime livre, *free flow*, o condutor viagem fora de qualquer um dos limiares definidos anteriormente e sem influência do restante tráfego. O condutor conduz com a aceleração máxima até alcançar a sua velocidade desejada.

$$a_n = b_{maxMult} * (v_{max} - v_n * FaktorV), \quad \text{com:} \tag{3-21}$$

$$FaktorV = \frac{v_{max}}{v_{des} + FaktorV_{mult} * (v_{max} - v_{des})}; \tag{3-22}$$

Com $b_{maxMult}$ e $FaktorV_{mult}$ como parâmetros de calibração.

Uma vez atingida a velocidade desejada, o condutor aplica alternadamente b_{null} e $-b_{null}$ conforme seja necessário para orbitar a velocidade desejada.

Parâmetros de calibração do modelo de Wiedemann '74

Não existe qualquer referência quanto aos valores a utilizar para os parametros de calibração nas publicações originais de *Wiedemann* e não são conhecidos publicamente os valores utilizados pela *PTV*, pelo que se apresentam os valores propostos por Higgs [46]:

Tabela 3-1- Parâmetros de calibração do modelo de Wiedemann '74 propostos em Higgs, 2011.

PARÂMETRO	VALOR	PARÂMETRO	VALOR	PARÂMETRO	VALOR
<i>AXadd</i>	1, 25	<i>CXadd</i>	4	<i>OPDVmult</i>	1,5
<i>AXmult</i>	2, 0	<i>CXmult</i>	4,6	<i>BNULLmult</i>	0,1
<i>BXadd</i>	2	<i>EXadd</i>	1,5	<i>BMAXmult</i>	1
<i>BXmult</i>	1	<i>EXmult</i>	0,55	<i>FAKTORVmult</i>	0,1
<i>CXconst</i>	5	<i>OPDVadd</i>	1,5		

3.1.3. MODELO DE GIPPS

O modelo de *Gipps* é construído tendo por base o modelo de *car-following* de distância de segurança ou prevenção de colisão ou modelo CA.³ É o modelo utilizado no programa comercial de simulação de tráfego *Aimsun*.

Neste modelo, os veículos aceleram até atingirem a sua velocidade desejada e desaceleram para evitar colisões. Quando constrangido por um veículo a jusante, o veículo n tenta ajustar a sua velocidade de modo a obter uma distância de segurança do veículo $n-1$. O modelo considera como distância de segurança, um afastamento suficiente que garanta que qualquer ação do veículo $n-1$ não culmina na ocorrência de uma colisão.

Em cada iteração, são calculadas as velocidades máximas a que o condutor do veículo n está disposto a circular para em condições desimpedidas, v_n^a , e condições constrangidas por um veículo a jusante, v_n^b , e o valor mínimo é aplicado ao veículo.

$$v_n = \min \{v_n^a(t + T), v_n^b(t + T)\} \quad (3-23)$$

O modelo de *Gipps* considera três tipos de aceleração possíveis de aplicar a um veículo:

- a_n^{max} – A máxima aceleração desejada pelo condutor do veículo n ;
- d_n^{max} – A máxima desaceleração desejada pelo condutor do veículo n ;
- d_{n-1}^{est} – A estimativa da máxima desaceleração desejada pelo veículo $n-1$;

A velocidade máxima em regime livre para a qual um veículo pode acelerar numa iteração é determina para o instante t em função da aceleração e velocidade corrente do veículo, de acordo com a seguinte formulação:

$$v_n^a = v_n + 2.5a_n \times T \times \left(1 - \frac{v_n}{v_n^{des}}\right) \sqrt{0.025 + \frac{v_n}{v_n^{des}}} \quad (3-24)$$

Onde:

- v_n – A velocidade do veículo n no instante t ;
- v_n^{des} – A velocidade desejada pelo veículo n ;
- a_n – A aceleração do veículo n ;
- T – O tempo de reação do condutor do veículo n ;

³ Exposto na secção 3.1.1.2 deste documento.

A máxima velocidade do veículo n em regime constrangido por um veículo $n - 1$ é determinada de acordo com a seguinte formulação:

$$v_n^b = d_n^{max}.T + \sqrt{(d_n^{max}.T)^2 - d_n^{max} \cdot \left[2(x_{n-1} - s_{n-1} - x_n) - v_n.T - \frac{v_{n-1}^2}{d_{n-1}^{est}} \right]} \quad (3-25)$$

Onde:

- s_{n-1} – O comprimento efetivo do veículo líder; é composto pelo comprimento do veículo $n-1$ e uma distância mínima entre veículos definida manualmente.
- x_n – A posição do veículo n ;
- x_{n-1} – A posição do veículo $n-1$;

A estimativa da máxima desaceleração desejada pelo veículo $n-1$, d_{n-1}^{est} , pode ser determinada de duas hipóteses. A primeira é assumir que a desaceleração desejada pelo veículo $n-1$ é igual à do veículo n , ou seja:

$$d_{n-1}^{est} = d_n^{max} \quad (3-26)$$

A segunda é calculando a máxima desaceleração desejada como a média das desacelerações desejadas por n e $n-1$:

$$d_{n-1}^{est} = \frac{d_n^{max} + d_{n-1}^{max}}{2} \quad (3-27)$$

3.1.4. MODELO DE FRITZSCHE

O modelo de *Fritzsche*, à semelhança do modelo de *Wiedemann*, é baseado no modelo psicofísico de *car-following* exposto na secção 3.1.1.4 deste documento, e é o modelo utilizado pelo programa comercial *Paramics*.

Este modelo tem por base a percepção do condutor face às diferenças de velocidade entre o seu veículo e o veículo a jusante, para a definição de cinco estados de comportamento: *Following I* (seguimento I), *Following II* (seguimento II), *Danger* (perigo), *Closing In* (aproximação) e *Driving Freely* (condução livre).

Estes estados são definidos em função de limiares de percepção de diferença de velocidades e distância entre dois veículos.

PTN e PTP:

Os limiares de percepção de uma diferença de velocidades negativa, PTN , e positiva, PTP , são determinados de acordo com as seguintes formulações:

$$PTN = -k_{PTN}(\Delta x - s_{n-1})^2 - f_x \quad (3-28)$$

$$PTP = k_{PTP}(\Delta x - s_{n-1})^2 - f_x \quad (3-29)$$

Com k_{PTN} , k_{PTP} e f_x como parâmetros de calibração e s_{n-1} o comprimento do veículo $n-1$.

AD:

A distância desejada, AD , representa a distância que o condutor do veículo n pretende manter do veículo em frente:

$$AD = s_{n-1} + T_D \times v_n \quad (3-30)$$

Onde T_D é um parâmetro de calibração.

AR:

A distância de risco, AR , representa a distância a partir da qual o veículo n é obrigado a uma travagem de emergência para evitar uma colisão:

$$AR = s_{n-1} + T_R \times v_{n-1} \quad (3-31)$$

Onde T_R é um parâmetro de calibração.

AS:

A distância de segurança, AS , representa a distância mínima a partir da qual o veículo n pode ter aceleração positiva, caso a distância entre veículos esteja a aumentar:

$$AS = s_{n-1} + T_S \times v_n \quad (3-32)$$

Onde T_S é um parâmetro de calibração.

AB:

A distância de travagem, AB , representa a distância a partir da qual, caso a velocidade do veículo n seja muito superior à velocidade do veículo $n-1$, poderá ocorrer uma colisão:

$$AB = AR + \frac{\Delta v^2}{\Delta b_m} \quad (3-33)$$

Onde Δb_m é dado por:

$$\Delta b_m = |b_{min}| + a_{n-1}^- \quad (3-34)$$

Com b_{min} e a_{n-1}^- como parâmetros de calibração.

Estes limiares definem os cinco regimes do modelo de *Fritzsche* que podem ser representados no plano $\Delta v, \Delta x$, de acordo com a Figura 3.4.

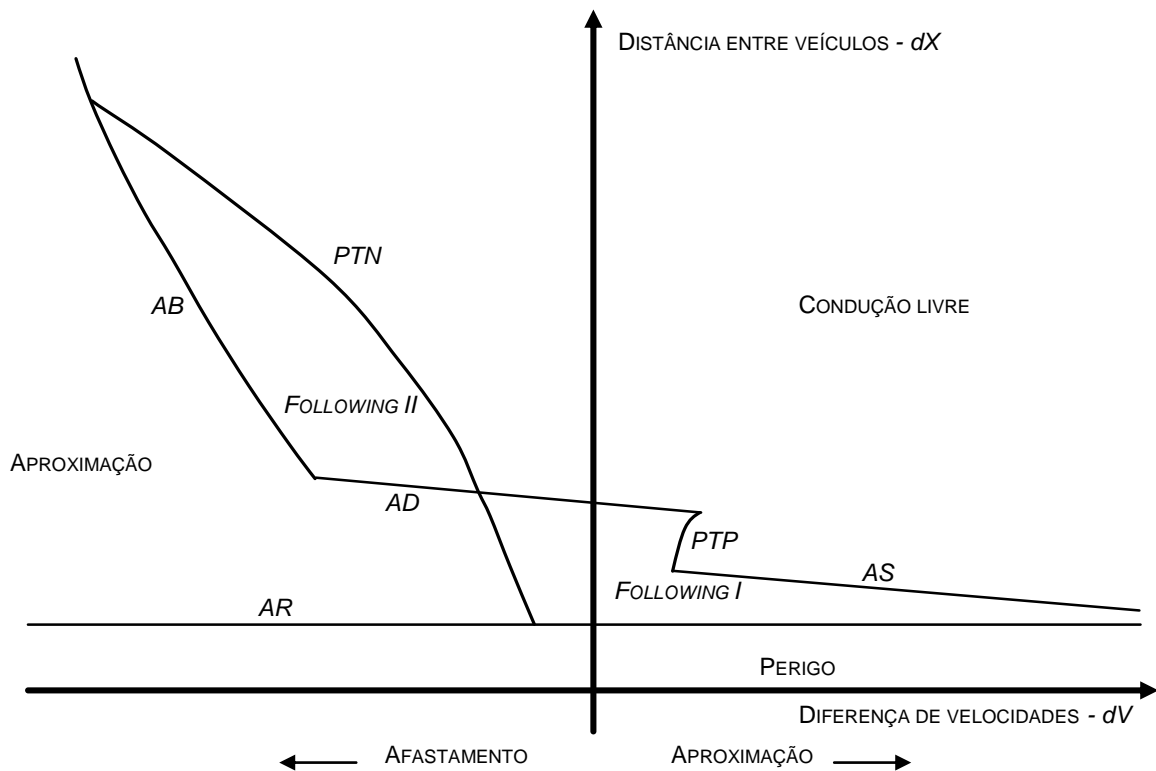


Figura 3.4 – Limiares de reação do modelo de Fritzsche.[47]

O condutor reage a cada estado alterando a sua aceleração:

Danger:

A distância ao veículo da frente é menor do que a distância de risco AR e o condutor aplica a velocidade b_{min} de modo a aumentar a distância.

Closing in:

A diferença de velocidades é maior do que PTN e a distância entre veículos está entre AB ou AD e AR , pelo que o condutor do veículo n de modo a obter a mesma velocidade do veículo $n-1$. A aceleração adotada por veículo n tem a seguinte definição:

$$a_n = \frac{(v_{n-1}^2 - v_n^2)}{2 \cdot d_c}, \text{ com:} \quad (3-35)$$

$$d_c = x_{n-1} - x_n - AR + v_{n-1} \cdot \Delta t, \text{ com:} \quad (3-36)$$

Following I:

Ocorre quando Δv está entre PTN e PTP e a Δx entre AR e AD ou Δv é superior a PTP e Δx entre AS e AR . Neste regime, o veículo n não tem uma ação consciente e é aplicada a aceleração b_{null} para modelar a incapacidade do condutor para manter uma velocidade constante.

Following II:

O regime *Following II* ocorre quando Δv é superior a PTN e Δx é superior a AB ou AD . Neste estado, o condutor toma percepção de que se está a aproximar do veículo em frente mas a distância entre veículos é demasiadamente grande para que seja necessário tomar qualquer ação.

Free driving:

No regime de *Free driving*, Δv é inferior a PTN e Δx superior a AD ou Δv é superior a PTP e Δx superior a AS . O condutor acelera até alcançar a sua velocidade desejada sem influência de outros veículos na rede.

3.2. LANE CHANGING E GAP ACCEPTANCE

Os modelos de mudança de via estudam a forma como um veículo passa da via onde se encontra para uma via adjacente. Este tipo de movimento tem um forte impacto na fluência do tráfego, tornando os modelos de mudança de via num componente vital na modelação de mudanças de direção, convergências, divergências, desvio de obstáculos e ultrapassagens.

3.2.1. ENQUADRAMENTO

Do ponto de vista da lógica de funcionamento de um programa de simulação, os modelos de mudança de via são de uma complexidade superior aos modelos de *car-following*, na medida em que exigem a consideração dos movimentos do tráfego circulante nas vias adjacentes e não apenas atenção ao veículo em frente. Adicionalmente, a modelação de mudanças de via envolve a implementação de algoritmos que simulem a vontade e necessidade do condutor para mudar de via e simulação da sua perceção quanto ao perigo da manobra face ao espaço disponível para efetuar a mesma. Por outro lado, em ambiente urbanos é necessário prever condições de congestionamento em que não existe espaço disponível para mudanças de via, sendo necessário modelar comportamentos de mudança forçada e demonstrações de cortesia e cooperação por parte de outros veículos.

Os modelos de mudança de via presentes em programas de simulação microscópica são geralmente na forma de *rule-based models (RBM)* (modelos baseados em regras) e, em menor escala, *discrete choice-based models (DCB)* (modelos de escolha discreta). Outros modelos, relacionados com teoria de jogos e redes neuronais, foram propostos, mas não obtiveram popularidade no mercado da microssimulação. [48]

As manobras de mudança de via são habitualmente classificadas como obrigatórias (*MLC*), ocorrem quando um condutor é obrigado a mudar de via para cumprir a rota desejada e discricionárias (*DLC*), quando a mudança de via tem por base um benefício para o condutor como ultrapassar um veículo mais lento ou evitar o tráfego convergente de uma via de aceleração. Ambos os comportamentos são posteriormente modelados em função da necessidade de mudança de via, da via alvo e do espaço disponível para a manobra (*gap acceptance*).

Os modelos de mudança de via utilizados em programas comerciais são habitualmente modelados e calibrados por cada fabricante, pelo que não é de conhecimento público a formulação utilizada. Assim, são abordados, ainda que superficialmente, os modelos de mudança de via utilizados pelos programas comerciais *Aimsun* e *Vissim*.

3.2.2. LANE CHANGING NO AIMSUN

O modelo de mudança de via aplicado no *Aimsun* é desenvolvido com base no modelo de *Gipps* (1986) [13] para mudança de via. Este modelo é baseado em regras e avalia, do ponto de vista do condutor, a necessidade de mudar de via, o desejo de mudar de via (motivação) e a viabilidade da manobra.

O desejo de mudar de via depende da perspectiva de vantagens obtidas com a mudança de via como, por exemplo, uma velocidade de circulação superior ou uma fila de espera mais reduzida. A viabilidade da manobra é determinada em função da existência de aberturas na via destino, da distância à próxima mudança de direção e das condições do tráfego na via em que se encontra.

No *Aimsun* são modeladas três zonas às quais correspondem motivações diferentes. Na zona 1, a principal preocupação no que diz respeito a mudança de via são as condições do tráfego na via alvo. Na zona 2, acresce a preocupação de mudar para a via que dá acesso à divergência. Na zona 3, é modelada a viabilidade de mudança de via.

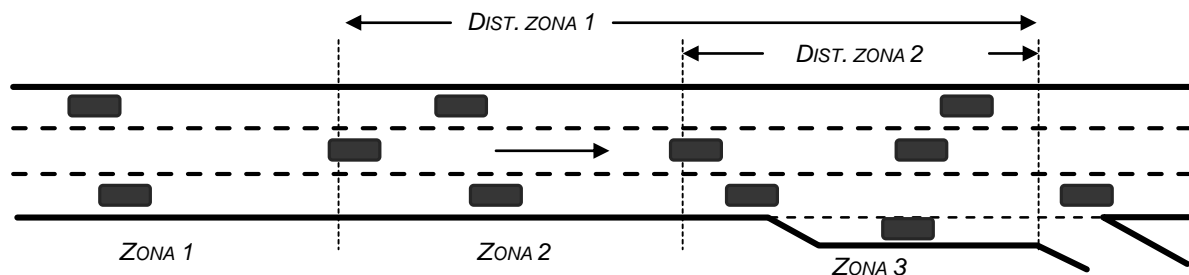


Figura 3.5 – Zonas de mudança de via no *Aimsun*. [49]

3.2.3. LANE CHANGING NO VISSIM

O modelo de mudança de via no *Vissim* é baseado no modelo desenvolvido originalmente por *Sparmann* (1978) [50]. É um modelo baseado em regras complexas que têm em consideração fatores como o tipo de via modelada (urbana ou autoestrada), o tipo de veículos envolvidos na manobra e a diferença entre mudar para uma via com velocidades superiores e uma via com velocidades inferiores.

À semelhança do modelo utilizado pelo *Aimsun*, o *Vissim* avalia a necessidade, o desejo e a viabilidade da mudança de via e categoriza as mudanças de via como obrigatórias (*MLC*) ou discricionárias (*DLC*).

Para além de mudanças de via, são modelados movimentos laterais dentro da mesma via para simular a ultrapassagem de velocípedes junto à berma.

4. DESENVOLVIMENTO DE UM SIMULADOR DE TRÁFEGO

4.1. ENQUADRAMENTO

O desenvolvimento de modelos de *car-following* foi intensivo entre o final da década de cinquenta e década de setenta e a procura do seu melhoramento e calibração tem sido bastante ativa durante as décadas seguintes [42]. No entanto, apesar da gama de estudos e experiências efetuadas para a calibração de modelos ou parte deles, pouco trabalho tem sido executado na tentativa de desenvolver um modelo de raiz completo.

Adicionalmente, as aplicações disponíveis no mercado, independentemente do tipo de licenciamento praticado, têm o seu foco na habilidade comercial do programa, dando pouca ou nenhuma liberdade ao utilizador na manipulação dos parâmetros que afetam os modelos de comportamento dos condutores. Isto resulta em utilizadores de ferramentas de simulação com um vazio no conhecimento dos parâmetros de calibração dos modelos, que têm uma forte implicação no desenvolvimento de modelos de tráfego em culturas diferentes daquela para que o modelo original foi desenvolvido. Deste modo, com o objetivo de criar uma plataforma open-source que permita, não só a manipulação dos parâmetros de calibração dos modelos comportamentais de tráfego, mas também a implementação de diferentes modelos, desenvolveu-se o programa *TRFS*.

O *TRFS*, abreviatura de *TRaFfic Simulator* é desenvolvido em *Java*, uma das mais populares linguagens de programação, utilizada numa grande variedade de aplicações, incluindo a programação de simuladores e videojogos e dispõe de uma sólida gama de bibliotecas desenvolvidas para auxiliar a programação deste género de aplicações, tornando-se uma opção atrativa para o desenvolvimento de um simulador de tráfego.

O *TRFS* permite a importação de uma rede criada em qualquer programa *SIG* no formato *GeoJSON*, um formato aberto criado para a codificação de uma variedade de dados geográficos, bastante popular no mercado de aplicações *open-source*. Os cenários são interativos e o *TRFS* permite verificar a influência dos parâmetros de calibração diretamente no tráfego ou nos gráficos representados ao longo da simulação.



Figura 4.1 – Logotipo TRFS

4.1.1. APRESENTAÇÃO DO PROGRAMA

O programa foi subdividido em quatro grandes grupos. A *user interface*, que representa toda a estrutura de interação com o utilizador, os *scenarios* (cenários), que tratam da importação e interpretação do mapa e restantes elementos pertencentes à rede, os *vehicles* (veículos) e os modelos matemáticos de comportamento dos condutores.

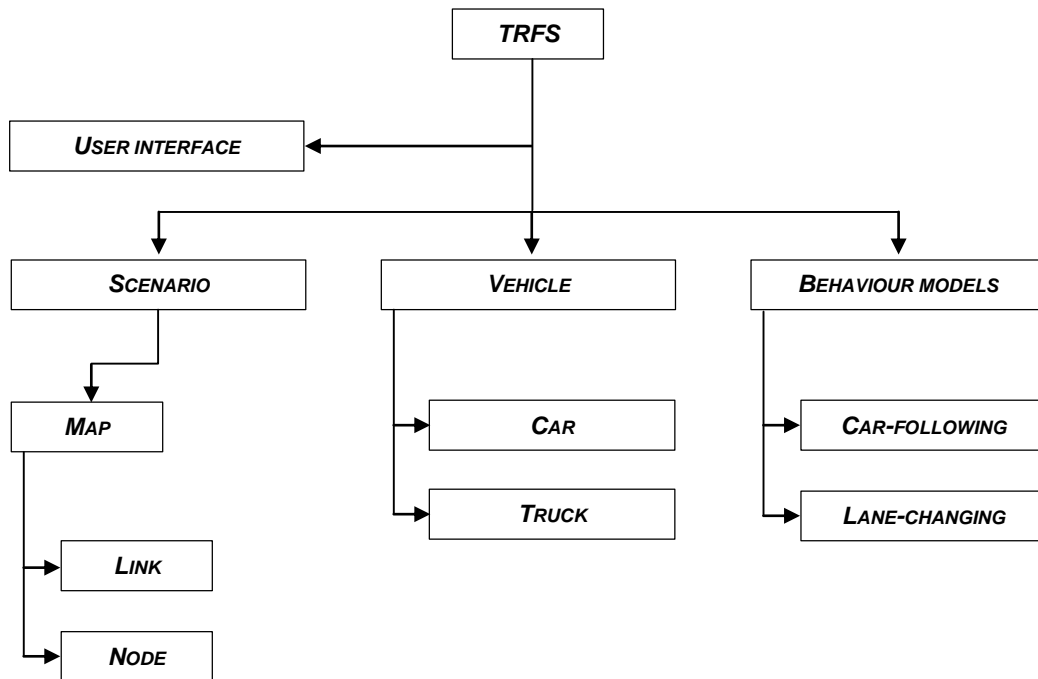


Figura 4.2 – Estrutura do programa em Java

Por sua vez, a interface gráfica é composta por vários elementos como ecrãs, botões, *sliders* e janelas que proporcionam uma experiência mais agradável ao utilizador.

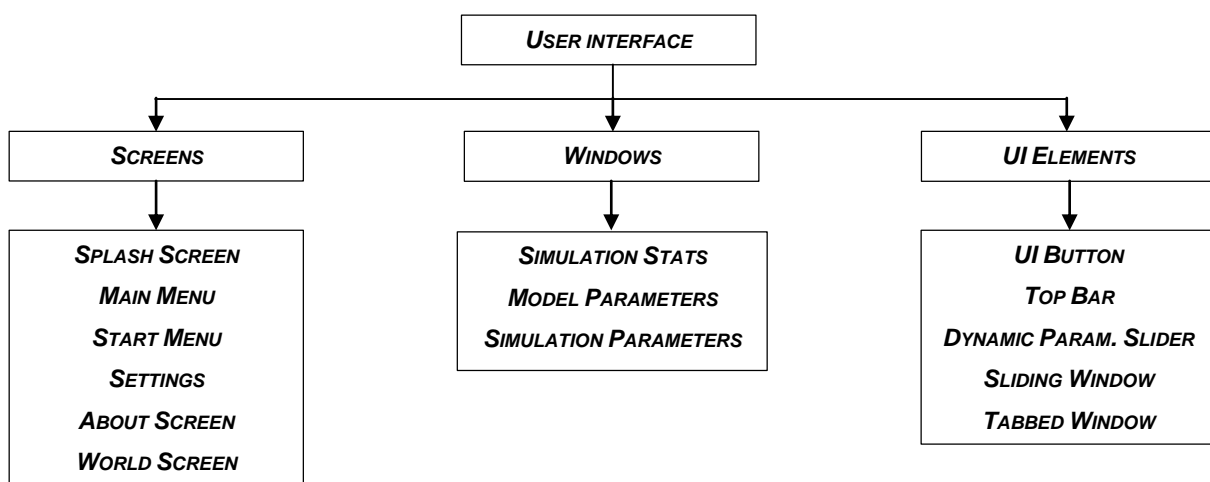


Figura 4.3 – Estrutura da interface gráfica

4.1.1.1. MAIN MENU

O menu principal apresenta o logotípo do simulador e serve apenas para direccionar o utilizador para os menus *Start*, *Settings* ou *About*.

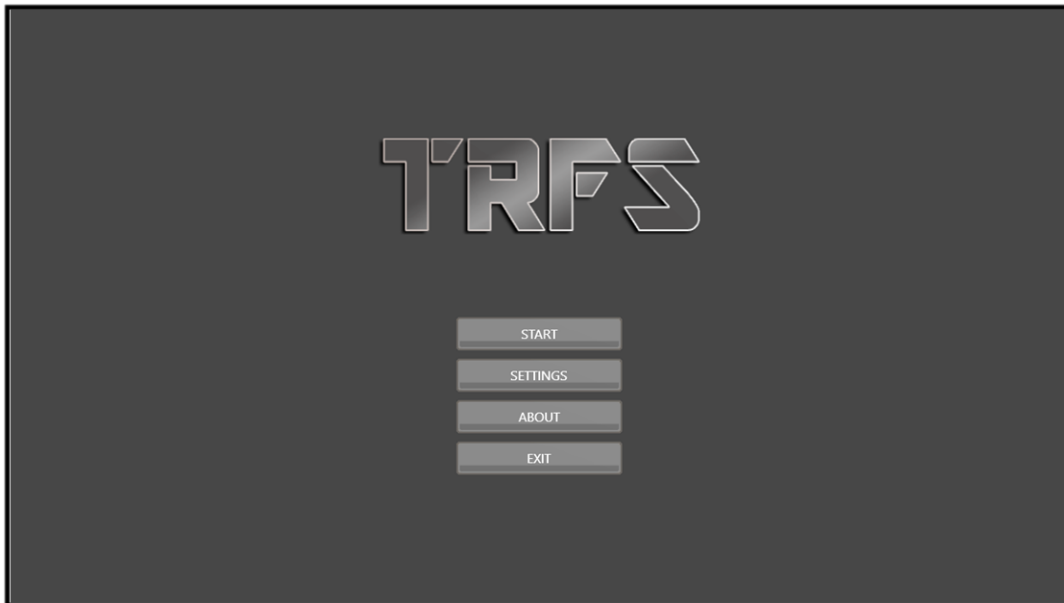


Figura 4.4 – Main Menu (menu inicial principal)

4.1.1.2. START MENU

No menu de iniciação, o utilizador selecciona o cenário que pretende simular, sendo-lhe facultada uma pré-visualização da rede, e poderá escolher se deseja que a geometria da rede seja simplificada e suavizada (útil em redes cujas curvas estejam pouco detalhadas ou com vértices muito acentuados).

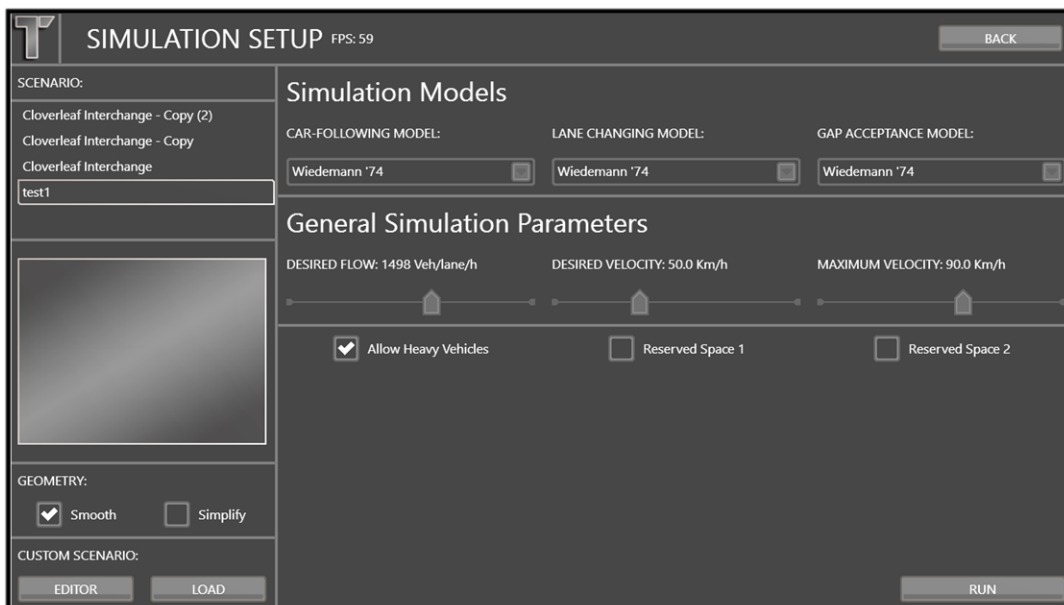


Figura 4.5 – Start Menu (menu de iniciação)

É possível selecionar quais os modelos comportamentais a utilizar na modelação dos agentes ao longo da simulação e os parâmetros a utilizar por defeito no início da simulação, relativos aos fluxos e velocidades praticadas na rede. Estes parâmetros são válidos apenas quando esta informação não está devidamente detalhada na rede.

4.1.1.3. SIMULATION

A janela de simulação representa de imediato a rede seleccionada. Na barra de topo disponibilizam-se botões para a ativação ou desativação das janelas de controlo dos parâmetros da simulação, parâmetros dos modelos matemáticos e da janela de representação de gráficos.

É possível mover, rodar e manipular o *zoom* do mapa com o rato e, quando uma janela se encontra ativa, manipular as barras de deslocamento de cada parâmetro relevante para a simulação.

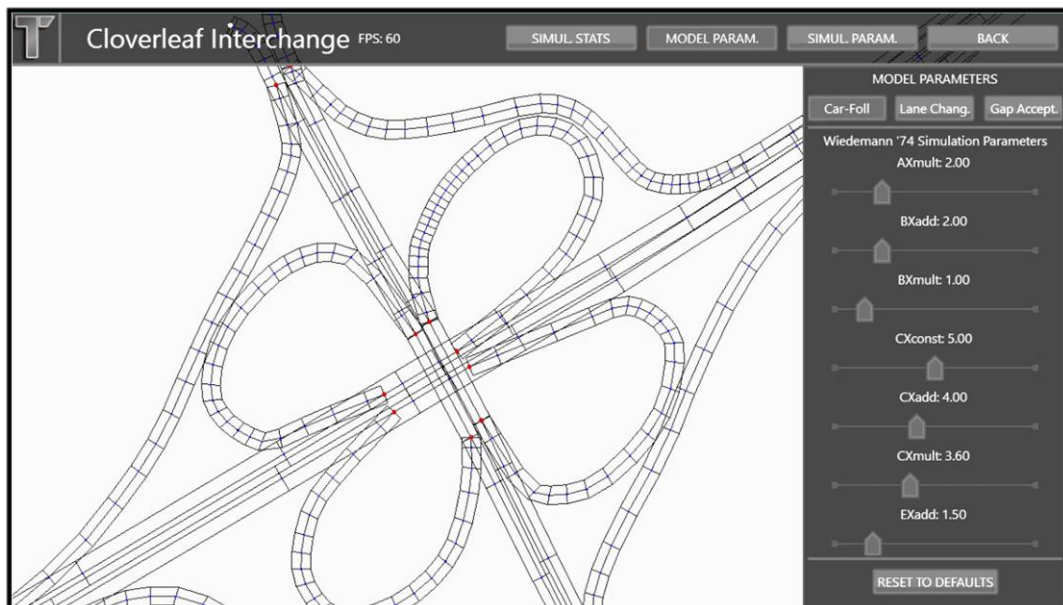


Figura 4.6 – Simulation Screen (ecrã de simulação)

4.1.1.4. SETTINGS

No menu *settings* (opções), é possível definir preferências relativas à *performance* gráfica do programa.

- **Enable Fullscreen** – Coloca o simulador em modo de ecrã inteiro;
- **Enable vSync** – Sincroniza a *frame rate* (cadência) com a frequência de atualização do monitor, devolvendo melhor qualidade gráfica, mas podendo diminuir a *performance* do programa;
- **Show FPS** – Apresenta a *frame rate* do programa na barra de topo do simulador;

- **Draw Debug** - Desenha a diretriz das vias e os pontos representativos dos nós para efeitos de depuração;

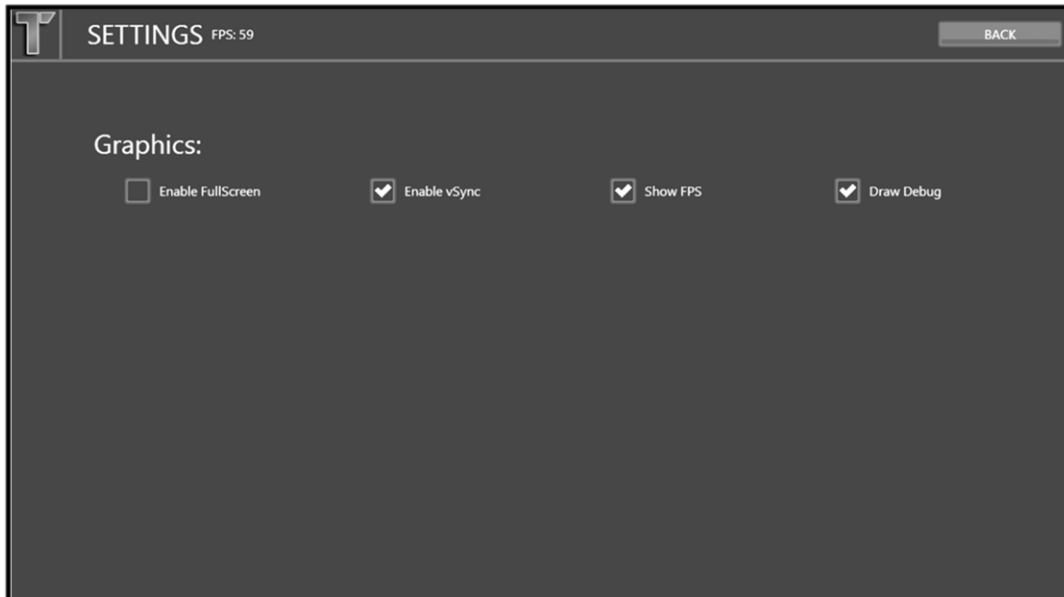


Figura 4.7 – Settings Menu (menu de opções)

4.1.1.5. ABOUT

Finalmente, no ecrã *About* (acerca) são apresentadas informações gerais sobre o programa e a sua autoria.

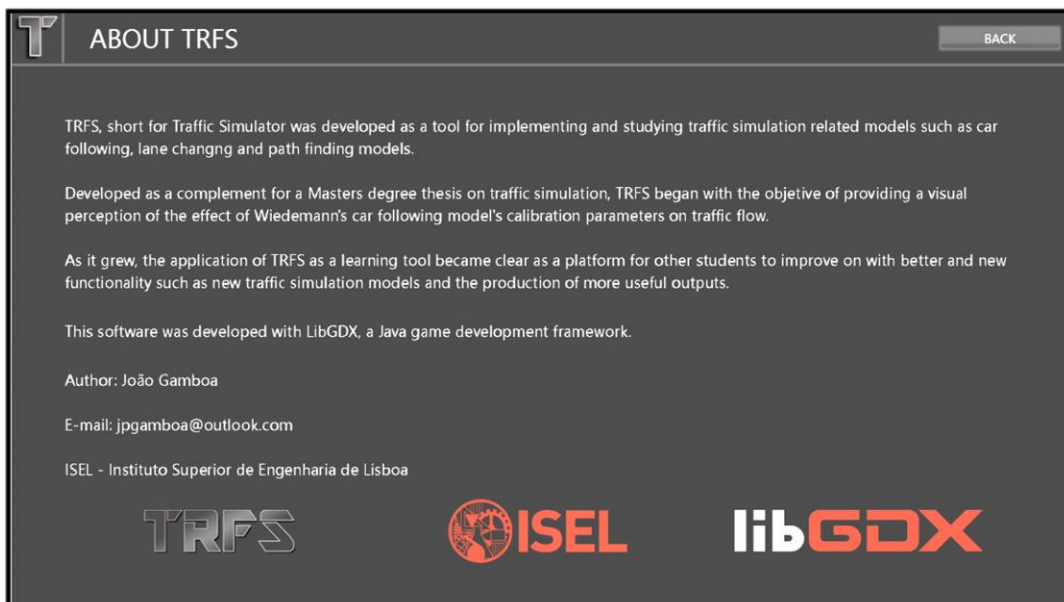


Figura 4.8 – About TRFS (acerca do TRFS)

4.2. IMPORTAÇÃO DE REDE SIG

O *TRFS* aceita redes no formato *GeoJSON* que podem ser desenvolvidas em qualquer programa *SIG* como, por exemplo, o *QGIS*.

O primeiro passo para a preparação da rede é o isolamento numa *layer* da rede que se pretende importar. A Figura 4.9 mostra, com a cor vermelha, um nó em trevo isolado da restante rede, a azul, para importação.



Figura 4.9 – Layer com a rede que se pretende analisar no TRFS

De seguida, é necessário preparar os atributos de cada arco que o *TRFS* necessita para modelar a rede:

- **Oneway** : Define se o arco contém tráfego apenas num sentido, com **1** representado uma via de sentido único, e **0** uma via de dois sentidos.
- **Maxspeed**: Representa a velocidade máxima do arco;
- **Hierarchy**: Define a hierarquia da via, a partir da qual é possível deduzir o comportamento dos condutores e outros parâmetros;
- **InFlow**: É o fluxo que se pretende avaliar no arco. No entanto, este valor pode ser alterado em tempo real ao longo da simulação;
- **Z**: Representa o nível do arco, permitindo a modelação de passagens superiores com a atribuição do valor **1**. Deste modo, o *TRFS* entende que o tráfego circulante em vias com um **z** diferente não deverão interagir.

	oneway	maxspeed	UID	hierarchy	lanes	inFlow	z
10	1	120	15595	1	3	1000.00000000...	0
11	1	120	15610	1	3	1000.00000000...	0
12	1	120	27805	2	2	500.00000000...	0
15	1	70	423234	2	1	200.00000000...	0
0	1	70	15579	2	2	0.000000000000	0

Figura 4.10 – Atributos da rede modelada em SIG

4.2.1. LEITURA DA REDE

O *TRFS* lê o ficheiro *GeoJSON* percorrendo as *features* (características) nele contidas, guardando as suas coordenadas e decidindo o tipo de arco a criar a partir dos atributos a ele associados.

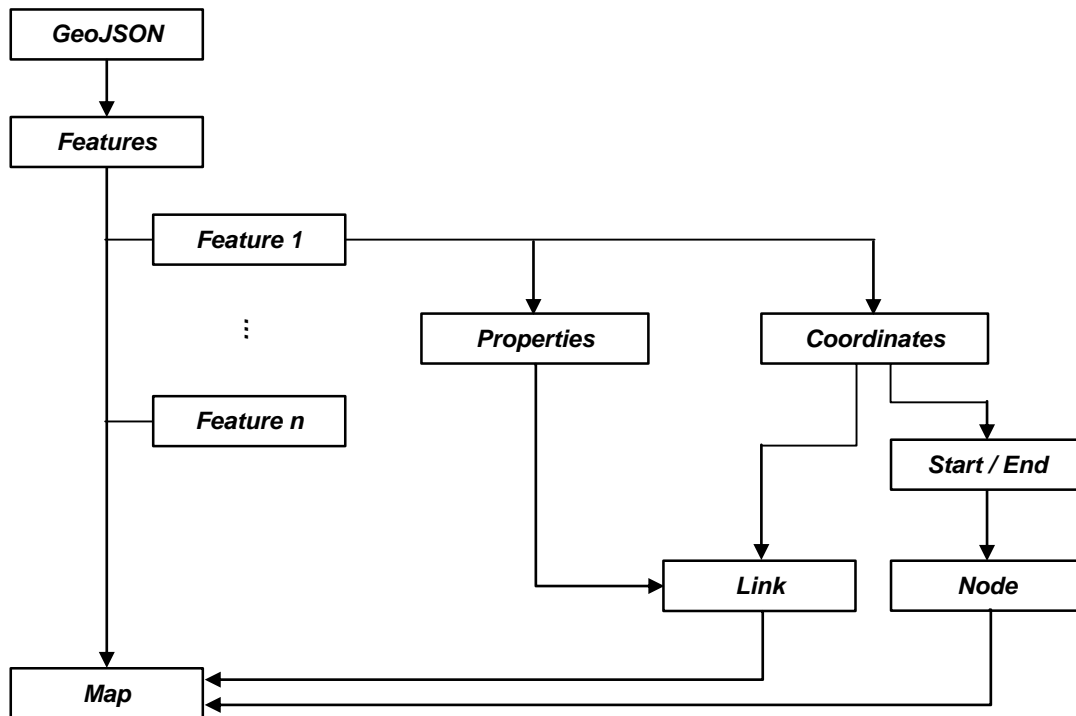


Figura 4.11 – Lógica de criação de um mapa no *TRFS*

Para cada arco, são guardadas as coordenadas do ponto inicial e final e transformados num elemento *Node* (nó), que representa uma interseção na rede e, uma vez concluída a leitura do ficheiro importado, efetua-se uma verificação da ocorrência de nós duplicados, eliminando-se as entradas desnecessárias.

Os arcos e nós são agregados no objeto *Map* (mapa) que serve de referência futura para o acesso a qualquer elemento da rede e efetua cálculos auxiliares tais como as coordenadas no centroide da rede, as dimensões totais do mapa e a conversão do sistema de coordenadas de coordenadas geográficas latitude e longitude para um sistema de coordenadas cartesianas.

Finalmente, caso o utilizador assim opte, são recalculadas as coordenadas dos arcos com a finalidade de produzir curvas mais suaves ou simplificar a geometria.

4.2.2. SIMPLIFICAÇÃO E SUAUIZAÇÃO DA REDE

À escala microscópica, irregularidades no desenho da rede tomam uma importância acrescida, não só no que diz respeito à satisfação visual, mas também na sua influência no percurso a descrever pelos veículos. Curvas com vértices muito acentuados, aliado a uma incapacidade dos veículos para compensar grandes mudanças de direção a velocidade elevadas, provocam perturbações irrealistas nas rotas dos veículos, levando-os a sair da via ou mesmo da faixa de rodagem onde circulam.

De modo a mitigar este efeito, foram implementados dois algoritmos, de simplificação e suavização, que, em conjunto, promovem curvas mais suaves sem perda das características fundamentais que definem a rede.

O algoritmo de simplificação remove pontos que se encontrem a uma distância inferior à tolerância pretendida. Este procedimento remove pontos que se encontrem muito próximos e, conseqüentemente, de importância reduzida para a geometria global da rede, eliminando o efeito representado na Figura 4.12.

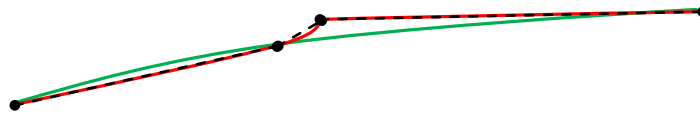


Figura 4.12 – Efeito do algoritmo de simplificação da rede

```

public static Array<Coordinate> simplifyGeometry(
    Array<Coordinate> coordinates, float tolerance) {

    Array<Coordinate> newCoordinates = new Array<Coordinate>();

    Coordinate point;
    Coordinate prevPoint = coordinates.get(0);

    newCoordinates.add(prevPoint);

    for (int i = 1; i < coordinates.size - 1; i++) {
        point = coordinates.get(i);

        float dx = point.x - prevPoint.x, dy = point.y
            - prevPoint.y;
        float distSQ = dx * dx + dy * dy;

        if (distSQ > tolerance) {
            newCoordinates.add(point);
            prevPoint = point;
        }
    }
    newCoordinates.add(coordinates.get(coordinates.size - 1));
    return newCoordinates;
}

```

Por sua vez, o algoritmo de suavização tem por base o método de Chaikin [51], que gera dois novos pontos por cada ponto original da rede. Iterações sucessivas produzem curvas incrementalmente mais suaves, como demonstra a Figura 4.13.

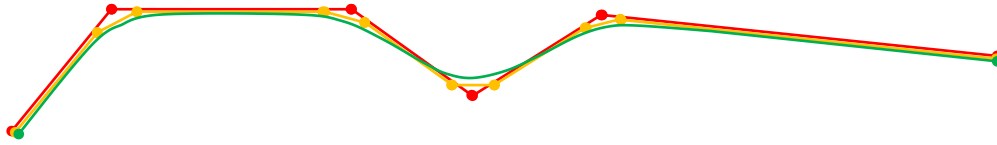


Figura 4.13 – Efeito da suavização da rede com o método de Chaikin

```

public static Array<Coordinate> smoothGeometry(
    Array<Coordinate> coordinates, int iterations) {
    Array<Coordinate> newCoordinates = new Array<Coordinate>();

    if (iterations > 0) {
        if (coordinates.size > 2) {
            newCoordinates.add(new Coordinate(coordinates.get(0).x,
                coordinates.get(0).y));
            for (int j = 0; j < coordinates.size - 1; j++) {
                Coordinate p0 = coordinates.get(j);
                Coordinate p1 = coordinates.get(j + 1);

                Coordinate n1 = new Coordinate(0.75f * p0.x + 0.25f
                    * p1.x, 0.75f * p0.y + 0.25f * p1.y);
                Coordinate n2 = new Coordinate(0.25f * p0.x + 0.75f
                    * p1.x, 0.25f * p0.y + 0.75f * p1.y);

                newCoordinates.add(n1); newCoordinates.add(n2);
            }
            newCoordinates.add(new Coordinate(coordinates.get(
                coordinates.size - 1).x, coordinates.get(
                coordinates.size - 1).y));
        } else newCoordinates = coordinates;
    } else newCoordinates = coordinates;
    return newCoordinates;
}

```

O resultado produzido pelo TRFS para ao fim de quatro iterações é visível na Figura 4.14.

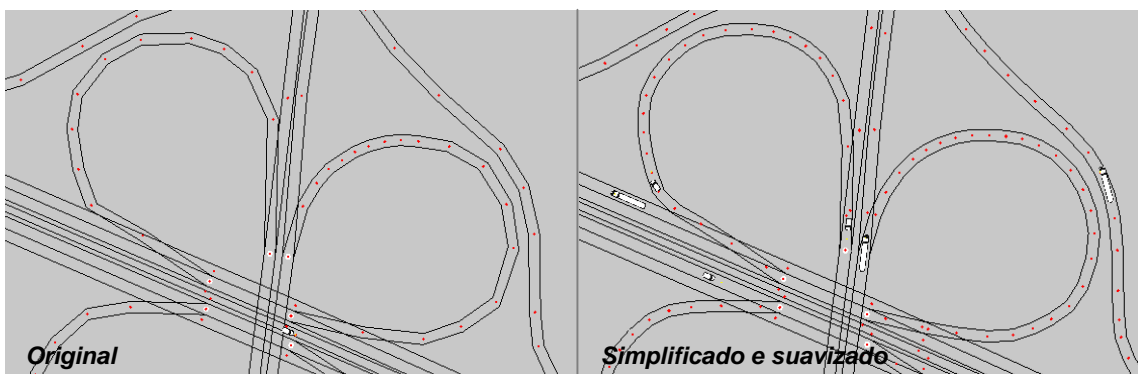


Figura 4.14 - Resultado dos algoritmos de simplificação e suavização no TRFS

4.3. TRÁFEGO

No *TRFS*, todos os veículos contêm uma referência ao *Link* (arco) e *Lane* (via) em que se encontram em qualquer momento da simulação. Esta metodologia permite que cada veículo apenas tenha de comparar as suas propriedades com as propriedades de outros veículos no mesmo arco e nos adjacentes, tornando o processo mais eficiente.

Em cada iteração do simulador, cada veículo efetua uma avaliação do seu meio, detetando e registando o seu líder e, quando exista desejo de mudar de via, os veículos na via adjacente de destino. Em função das condições apuradas, são atualizados os modelos de *car-following*, *lane changing* e *path following*⁴ para determinar a aceleração do veículo que, por sua vez, é utilizada para determinar a velocidade e posição do mesmo na iteração seguinte.

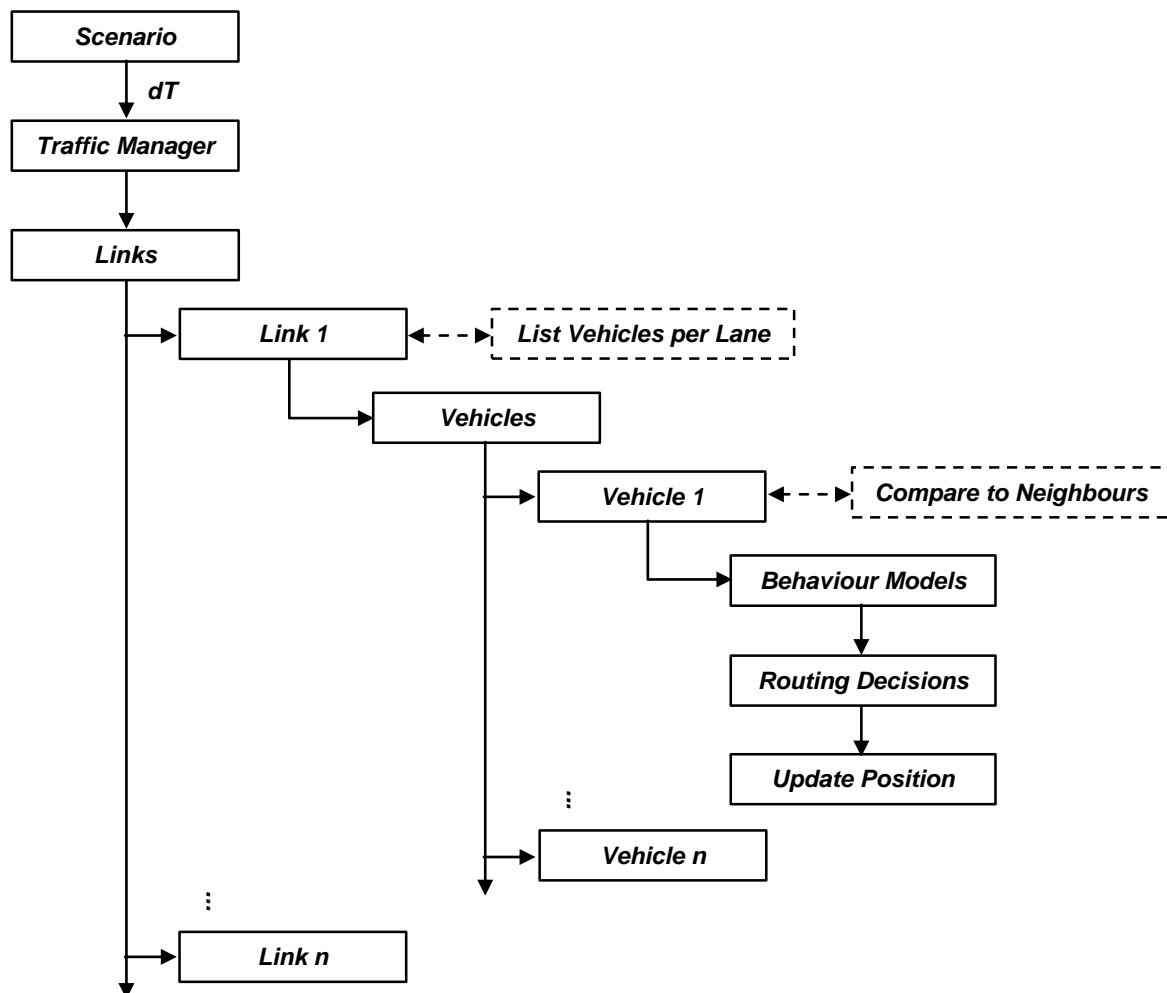


Figura 4.15 – Ciclo de atualização do estado do tráfego na rede

⁴ *Path-following*: o modelo que descreve a forma como os agentes se movem ao longo da rede.

4.3.1. DEFINIÇÃO DE PERCURSOS

Os percursos a descrever pelos veículos na rede são definidos por uma sequência de *IDs* de *Links* a partir dos quais o veículo poderá determinar as coordenadas que deverá seguir. Uma vez que é um procedimento lento, todos os percursos possíveis são determinados e guardados em memória no início da simulação. É calculado um percurso desde cada nó que contenha um fluxo de entrada para cada nó de saída da rede.

A determinação do caminho mais curto entre cada par OD é efetuada com recurso ao algoritmo *Depth-first search* (busca em profundidade), *DFS*, que retorna todos os caminhos possíveis entre dois nós, guardando-se apenas o caminho com o menor peso (*weight*) ou custo. No TRFS, nesta fase, apenas se considera o comprimento de cada arco para o cálculo da impedância, sendo possível a implementação de outros parâmetros no futuro.

```

private Array<Link> iterDepthFirst(Node entranceNode, Node exitNode) {
    Array<Array<Link>> results = new Array<Array<Link>>();

    Array<Node> visited = new Array<Node>();
    Array<Node> stack = new Array<Node>(); stack.add(entranceNode);
    Node current, neighbour;

    boolean justVisitedExit = false;
    while (stack.size > 0) {
        current = stack.peek();
        neighbour = getUnvisitedNeighbourNode(current, visited,
            exitNode, justVisitedExit);
        if (neighbour != null) {
            visited.add(neighbour);
            stack.add(neighbour);
            if (neighbour.equals(exitNode)) { //Save stack
                justVisitedExit = true;
                results.add(nodeStackToLinkArray(stack));
                stack.pop();
            }
        } else stack.pop();
    }
    return cheapestPath(results);
}

private Node getUnvisitedNeighbourNode(Node node, Array<Node> visited,
    Node exitNode, boolean justVisitedExit) {
    for (Link link : node.toLinks) {
        if (!visited.contains(link.toNode, false)) return link.toNode;
        if (link.toNode.quickEquals(exitNode)) {
            if (justVisitedExit) {
                justVisitedExit = false; return null;
            } else return link.toNode;
        }
    }
    return null;
}

```

No momento da criação de um veículo, é-lhe atribuído um percurso aleatório em função do seu nó de origem.

4.3.2. CAR-FOLLOWING (WIEDEMANN '74)

Para um estudo mais didático da implementação programática do modelo de *Wiedemann* optou-se pela utilização do *Matlab*, com o objetivo de correr o modelo numa rede simples sem preocupações de grafismo, escalabilidade e comunicação com os restantes elementos do programa.

O programa desenvolvido em *Matlab* é composto por cinco funções:

- *getInput* Recolhe dados do utilizador sobre as condições iniciais do sistema, no que diz respeito à velocidade máxima e desejada, comprimento da rede e número de vias a implementar;
- *addCar* Responsável pela adição de veículos na rede; define o comprimento, largura, posição e velocidade de entrada do veículo na rede. Define ainda os parâmetros aleatórios do comportamento do condutor e a via em que este entra na rede;
- *moveCar* Move o veículo ao longo da rede com base nas equações do movimento, efetua a deteção dos veículos em redor, e implementa o modelo de *car-following*;
- *removeCar* Apaga os vestígios de veículos que saíram da rede;
- *play* Controla o *input* do utilizador em tempo real, faz a gestão das restantes funções ao longo do funcionamento do programa e implementa a representação gráfica do modelo.

Apresentam-se os aspetos importantes da implementação do código em *Matlab*, com excertos de código nas situações relevantes.

4.3.2.1. ADIÇÃO DE VEÍCULOS

A função *addCar()* tem como objetivo reunir todas as operações necessárias para adicionar um novo veículo à rede.

4.3.2.1.1. COMPRIMENTO

O comprimento do novo veículo é determinado com base numa distribuição normal de valor médio 6 m, com um desvio padrão de 0,5 m. Assim, o modelo representa apenas veículos ligeiros, mas será possível adicionar um algoritmo que adicione veículos pesados na percentagem desejada.

Os valores são registados na variável *comp*:

```
%Car Lengt (leng -> local var) (6m +- 0.5m)
leng = random('normal',6,0.5);
car.comp(end + 1) = leng;
```

4.3.2.1.2. VELOCIDADE DE ENTRADA

A velocidade de entrada na rede, v , é definida com base na velocidade desejada ($VDES$), definida pelo utilizador durante a simulação, através de uma distribuição normal com um desvio padrão de 5 m/s ou 18 Km/h. Este valor foi obtido através de sucessivas iterações do valor do desvio padrão, verificando-se fornecer resultados com uma variabilidade satisfatória.

```
%Entrance Speed (Defined according to desired velocity)(vel ->
%local var)
vel = random('normal',car.VDES,5);
car.v(end + 1) = vel;
```

4.3.2.1.3. SELEÇÃO DA VIA DE ENTRADA

A via em que o novo veículo entra na rede depende do número de vias do troço de estrada (1, 2 ou 3). No primeiro caso, é sempre associado o valor 1 por não existirem outras opções. É ainda adicionado um contador “*ctrlane*” para controlar o número de veículos que entra em cada via para uso futuro.

```
switch car.NLANES
case 1 %For 1 lane
% Every car goes on lane 1 (rightmost)
LANE = 1;
car.ctrlaneone = car.ctrlaneone + 1;
```

No caso de existirem duas vias, a escolha é efetuada com base na velocidade desejada, $VDES$. Veículos que circulem a uma velocidade inferior a $VDES$ são colocados na via número 1 e os restantes na via número 2.

```
case 2 %For 2 lanes
% If entrance speed is smaller than the desired speed,
% the vehicle chooses lane 1
if vel < car.VDES
LANE = 1;
car.ctrlaneone = car.ctrlaneone + 1;
% If entrance speed is greater than the desired speed,
% the vehicle chooses lane 2
else
LANE = 2;
car.ctrlanetwo = car.ctrlanetwo + 1;
end
```

Para uma estrada de três vias, a seleção é feita de acordo com a seguinte condição:

$$VIA = \begin{cases} 1 & \text{se } Vel.Inicial < 0,9 VDES \\ 2 & \text{se } Vel.Inicial < 1,1 VDES \\ 3 & \text{nos restantes casos} \end{cases} \quad (4-1)$$

```

case 3 %For 3 lanes
    % If entrance speed is smaller than 10% the desired speed,
    % the vehicle chooses lane 1
    if vel < 0.9*car.VDES
        LANE = 1;
        car.ctrlaneone = car.ctrlaneone + 1;
    %If entrance speed is 10% smaller or greater than the
    %desired speed, the vehicle chooses lane 2
    elseif vel < 1.1*car.VDES
        LANE = 2;
        car.ctrlanetwo = car.ctrlanetwo + 1;
    % If entrance speed is greater than 10% the desired speed,
    % the vehicle chooses lane 3
    else
        LANE = 3;
        car.ctrlanethr = car.ctrlanethr + 1;
    end
end

```

Finalmente, determina-se a posição Y do veículo para representação gráfica de acordo com:

$$\begin{aligned}
 \text{Pos. } Y = & \text{Larg. Berma} + \text{Larg. Via} - \text{Larg. Veíc.} - \frac{L. \text{Via} - L. \text{Veíc.}}{2} \\
 & + (\text{Nr. Via} - 1) * L. \text{Via}
 \end{aligned} \quad (4-2)$$

E regista-se o valor na variável y:

```

%Defining y Position for graphical representation (posy -> local var)
posy = Cars.SHOULDERWID+Cars.LANEWID-car.larg-(Cars.LANEWID-
car.larg)/2+(LANE-1)*Cars.LANEWID;
car.y(end + 1) = posy;

```

4.3.2.2. MOVIMENTO

A função *movecar()* inclui todas as operações que são necessárias efetuar em cada iteração para determinar o estado psicofísico e a aceleração de cada veículo e, em função destes, atualizar a sua velocidade e posição. Este bloco de código é corrido para cada veículo na rede em cada iteração do programa, pelo que constitui o maior desafio no que diz respeito à eficiência da simulação e à gestão de memória.

Em cada instante da simulação existe um determinado número de veículos na rede, quantificável pelo comprimento dos vetores ou matrizes com as características de cada veículo que vão sendo armazenadas.

De acordo com o seguinte exemplo, representando os vetores *ncar*, *lane* e posição *x* com os primeiros três veículos a entrar na rede, observa-se que é possível obter a via e a posição *x* em que um veículo se encontra fazendo corresponder o seu índice no vetor *ncar* com os valores no mesmo índice em vetores diferentes.

$$\begin{array}{l}
 \text{Índice 1} \\
 \text{Índice 2} \\
 \text{Índice 3}
 \end{array}
 \quad
 \text{ncar} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}; \quad
 \text{lane} = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}; \quad
 x = \begin{bmatrix} 580,8 \\ 530,3 \\ 356,9 \end{bmatrix};$$

Deste modo, é imperativo preservar esta correspondência quando se pretende adicionar ou remover veículos da rede, uma vez que os algoritmos de deteção e movimento dependem da integridade destes vetores.

4.3.2.2.1. DETEÇÃO DE PROXIMIDADE E DETERMINAÇÃO DO VEÍCULO EM FRENTE

O algoritmo de deteção consiste em percorrer a lista de veículos na rede e comparar a sua posição e via com os restantes veículos, incluindo o próprio. Em cada *subloop*, o programa avalia se os veículos estão na mesma via e, em caso afirmativo, regista a distância entre eles, caso seja a mínima calculada até ao momento.

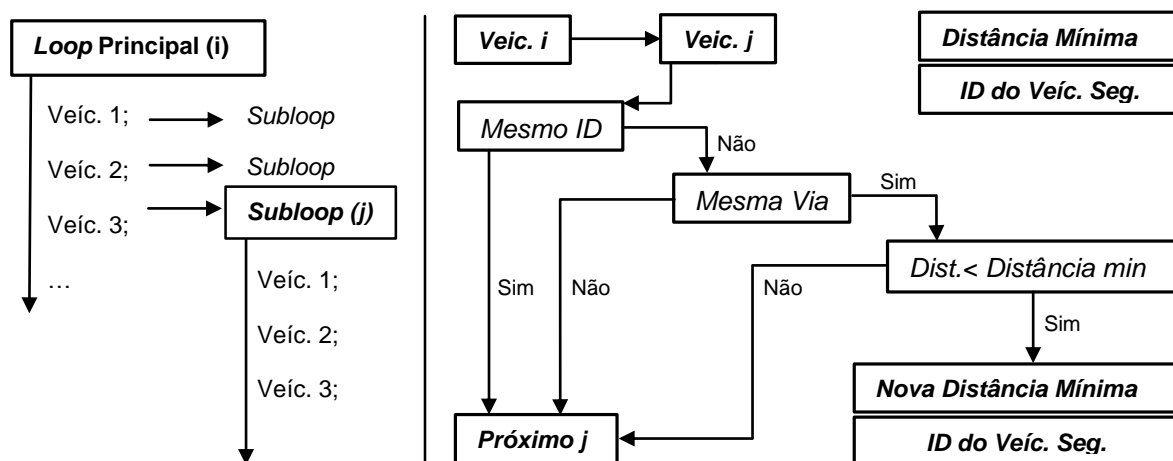


Figura 4.16 – Algoritmo de deteção do veículo em frente

Traduzindo o algoritmo em código obtém-se:

```

%For each car deployed
for i=1: numel(car.ncar)
    minDist = car.ROADLEN;
    %Compare car i to every other car
    for j = 1: numel(car.ncar)
        %If not comparing the same car
        if i~=j
            %Check if both cars are on the same lane
            lanedif = car.lane(i) - car.lane(j);
            %Compute distance between the two cars
            dist = car.x(j) - car.x(i);
            switch abs(lanedif)
                %If both cars are on the same lane
                case 0
                    %If car j is ahead of car i
                    if dist >= 0
                        %Store minimum distance and next
                        %car's ID
                        if dist < minDist
                            minDist = dist;
                            nextID = j;
                        end
                    end
                end
            end
        end
    end
end
end
end
end
  
```

Saindo do *subloop*, ficará registado o *ID* do veículo em frente na variável *nextID*, que servirá para recolher as informações necessárias para o funcionamento do modelo de *car-following*. No entanto, o algoritmo não é aplicável aos primeiros veículos de cada via a entrar na rede, não se obtendo qualquer valor para *nextID*. Nestes casos, atribui-se uma velocidade constante e o veículo avança sem consideração por qualquer fator externo.

O código permite a implementação futura de um modelo mais realista para os primeiros veículos.

```
%If there is no car ahead
if ~exist('nextID','var')
    %Keep moving at constant speed
    car.a(i) = 0;
    car.v(i) = car.v(i);
    car.x(i) = car.x(i) + car.v(i)*car.dt;
```

Caso se obtenha um alvo, regista-se o valor de *nextID* na variável global *nextcar* para utilização futura e procede-se para o modelo de *Wiedemann*.

```
%If there is a car ahead
else
    %Store next car's ID to a global variable
    car.nextcar(i) = car.ncar(nextID);
```

4.3.2.3. IMPLEMENTAÇÃO DO MODELO DE CAR-FOLLOWING

O modelo desenvolve-se em torno da distância frente a frente, dX , e a diferença de velocidade entre os dois veículos, dV , pelo que se começa por calcular estes dois valores. Determina-se ainda a distancia efetiva entre os dois veículos, DX , subtraindo o comprimento do líder a dX .

```
%Distance Front to Front
dX = car.x(nextID)-car.x(i);
%Speed Difference (>0 means closing in)
dV = car.v(i)-car.v(nextID);
%Distance Front to Back
DX = dX - car.comp(nextID);
```

De seguida, calculam-se os parâmetros dinâmicos do modelo de *Wiedemann* recorrendo aos parâmetros estáticos determinados com a criação do veículo em estudo, na função *addCar*.

```
car.BX(i) = car.BXp1(i) * sqrt(car.v(i));
car.ABX(i) = car.AX(i) + car.BX(i);
car.SDV(i) = ((DX-car.AX(i))/car.CX(i))^2;
car.SDX(i) = car.AX(i) + car.EX(i)*car.BX(i);
car.CLDV(i) = car.SDV(i)*car.EX(i)^2;
car.OPDV(i) = car.CLDV(i)*(car.OPDVp2(i));
```

Estes parâmetros são comparados com dX , DX e dV , de acordo com algoritmo exposto na secção 3.1.2 e permitem determinar qual o estado psicofísico do condutor: *Braking*, *Following*, *Approaching* ou *Free Flow*. Adicionou-se ainda um quinto estado, *Impact*, com o intuito de tornar visível a ocorrência de acidentes e prevenir a ocorrência de erros provenientes da sobreposição de veículos.

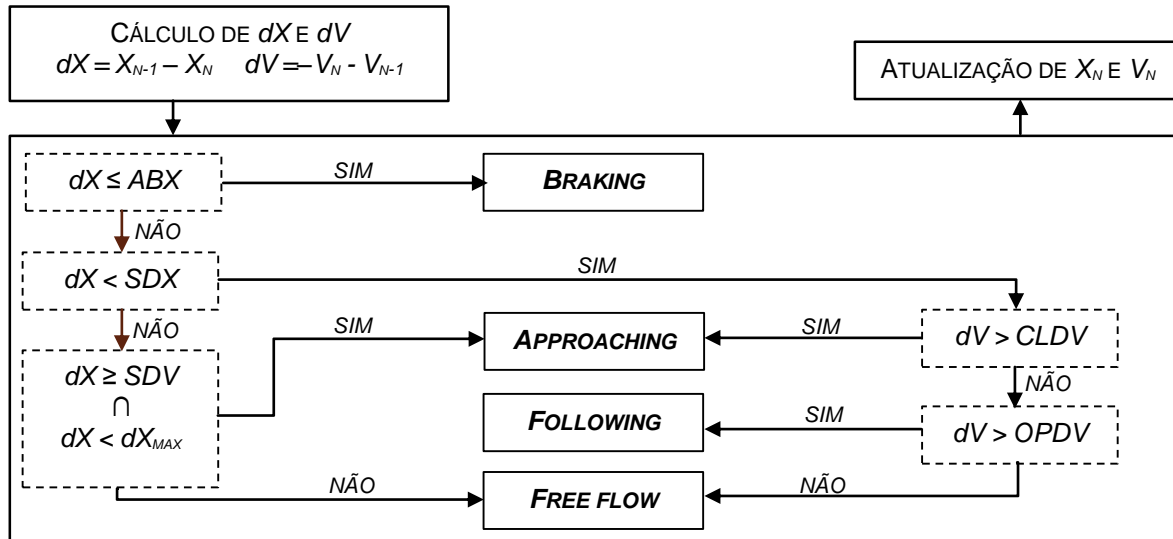


Figura 4.17 – Procedimento de decisão no modelo de Wiedemann '74 [54]

Traduzindo o algoritmo em código, obteve-se:

```
%Impact
if DX <= 0
    car.cor(i,:) = [255,0,0]/255;
    car.v(i) = car.v(nextID);

%Braking
elseif dX <= car.ABX(i)
    car.a(i) = (1/2)*((dV^2)/
        (car.ABX(i)- DX))+
        car.a(nextID) +
        (-20+(1.5/60)*car.v(i))*
        (car.ABX(i)-DX)/car.BX(i);
    car.cor(i,:) = [255,69,0]/255;

%Following
elseif dX < car.SDX(i)
    %Positive
    if dV > car.CLDV(i)
        car.a(i) = -car.BNULL(i);
        car.cor(i,:) = [0,0,255]/255;
    %Negative
    elseif dV > car.OPDV(i)
        car.a(i) = car.BNULL(i);
        car.cor(i,:)=[106,90,205]/255;
    else %Keep old values
        continue

%Approaching (Closing)
elseif dX < 50*sqrt(dV) && dV >=
car.SDV(i)
    BMIN = -20+(1.5/60)*car.v(i);
    %Máx Desaceleração Possível
    BN =(1/2)*((dV^2)/(car.ABX(i)-
    DX))+car.a(nextID);
    if BN <= BMIN
        car.a(i) = BN;
        car.cor(i,:) =[255,215,0]/255;
    else
        car.a(i) = BMIN;
        car.cor(i,:) = [255,215,0]/255;
    end

%Free Flow
elseif dX > car.SDX(i) && dV <=
car.SDV(i) %BMAX
    car.a(i) = car.BMAXmult*car.VMAX-
    car.v(i)*car.FaktorV(i));
    car.cor(i,:) = [50,205,50]/255;
end
```

A cada um dos estados corresponde uma nova aceleração, a , e uma nova cor, cor , que servem, respetivamente, para calcular uma nova velocidade e atualizar a representação gráfica do veículo.

$$v(i) = v(i) + a(i) \times dt; \quad (4-3)$$

Com o novo valor para a velocidade do veículo, determina-se a nova posição x , em função do intervalo de tempo, dt , pretendido entre cada iteração alterado em tempo real pelo utilizador durante a simulação.

$$x(i) = x(i) + v(i) \times dt + \frac{a(i) \times dt^2}{2}; \quad (4-4)$$

4.3.2.4. SIMULAÇÃO E REPRESENTAÇÃO GRÁFICA

A função *play()* engloba não só os elementos gráficos que compõem o troço da estrada, mas também o controlo da simulação e o *input* do utilizador em tempo real.

4.3.2.4.1. DESENHO DO TROÇO

A simulação desenrola-se numa janela controlada pela função *figure()* do *MATLAB*, que permite, entre outros, pré-definir um título e o tamanho desejado e, no seu interior, é possível inserir vários *subplots* e elementos tais como barras de deslocamento.

As vias e as bermas são desenhadas em função do número de vias selecionadas pelo utilizador, e representadas com a função *rectangle()* do *MATLAB*.

```
rectangle('Parent',car.sp1,...
          'FaceColor',[143/255,188/255,143/255],...
          'Position',[0,0,car.ROADLEN,...
                    2*Cars.SHOULDERWID+car.NLANES*Cars.LANEWID],...
          'EdgeColor','none');
rectangle('Parent',car.sp1,...
          'FaceColor',[79/255,79/255,79/255],...
          'Position',[0,Cars.SHOULDERWID,car.ROADLEN,...
                    car.NLANES*Cars.LANEWID],...
          'EdgeColor','none');
```

Desenham-se as marcações na via recorrendo a um pequeno *loop* em função do número de vias, *NLANES*.

```
for i = 1:(car.NLANES-1)
    line([0,car.ROADLEN],[Cars.SHOULDERWID+i*Cars.LANEWID,
                      Cars.SHOULDERWID+i*Cars.LANEWID],...
         'LineStyle','--','Color','w','LineWidth',2)
end
line([0,car.ROADLEN],[Cars.SHOULDERWID,Cars.SHOULDERWID],
     'LineStyle','-','Color','w','LineWidth',2)
line([0,car.ROADLEN],[Cars.SHOULDERWID+(car.NLANES*Cars.LANEWID),...
                      Cars.SHOULDERWID+(car.NLANES*Cars.LANEWID)],
     'LineStyle','-','Color','w','LineWidth',2)
```

Obtendo-se o resultado:

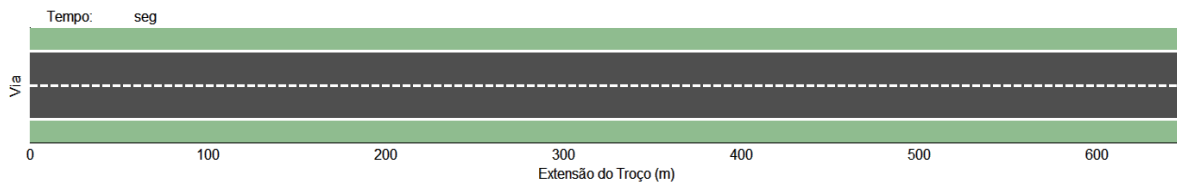


Figura 4.18 – Representação do troço de estrada

4.3.2.5. CONTROLO DO FLUXO NA REDE

Para saber quando introduzir um novo agente na rede, toma-se como base o fluxo, *FLOW*, definido pelo utilizador. A variável *flowdt* permite controlar quantos *loops* deverão decorrer até a introdução de um novo veículo, fazendo:

$$flowdt = \frac{Fluxo \times dt}{3600}; \quad (4-5)$$

Em cada iteração, a variável *fullcar* é incrementada de *flowdt*. Um veículo é inserido na rede quando *fullcar* toma um valor igual ou superior a 1, utilizando a função *addCar()*.

```
%Adding Cars
flowdt = (car.FLOW/3600)/(1/car.dt);
fullcar = fullcar + flowdt;
if fullcar >= 1
    addCar(car);
    fullcar = fullcar-1;
end
```

4.3.2.6. DETEÇÃO, REGISTO E REMOÇÃO DOS VEÍCULOS FORA DA REDE

Para que a quantidade de informação não sobrecarregue o computador e mantenha o número de iterações num valor razoável, é imperativo eliminar do sistema os veículos que já tenham saído da zona de observação. Para tal, regista-se o *ID* dos veículos cuja posição seja superior a 1,5 vezes o comprimento do troço na variável *carstoremove*. Uma vez fora do *loop* principal, procede-se à remoção dos veículos invocando a função *removecar()*, utilizando como *input* os *IDs* guardados na variável *carstoremove*.

```
%Save the ID of cars to be removed
if car.x(i) > 1.5*car.ROADLEN
    car.carstoremove(end + 1) = car.ncar(i);
end
```

A função *removecar()* remove todos os indícios da existência de um veículo que se pretenda eliminar, apagando o valor correspondente ao índice registado em *carstoremove* de cada uma das matrizes que contêm informação sobre o agente.

4.3.2.7. RESULTADOS

O programa desenvolvido permite a observação da evolução do tráfego em função do fluxo na rede, da velocidade máxima, da velocidade desejada e do intervalo de tempo entre cada iteração, permitindo ao utilizador a manipulação de cada parâmetro em tempo real. Deste modo, é possível obter uma perceção visual do efeito de cada variável no modelo de *Wiedemann* e da forma como este modela a interação entre veículos, através do código de cores descritivo dos diferentes estados de aceleração.

Por outro lado, o programa foi desenvolvido com o objetivo de permitir uma futura adaptação de modelos de *car-following* de diferentes autores, permitindo a comparação do comportamento resultante de cada modelo. Estando preparado para a simulação de um troço retilíneo com entre uma e três vias de circulação, constitui também uma plataforma para a implementação e estudo de modelos de *lane changing* (mudança de via).

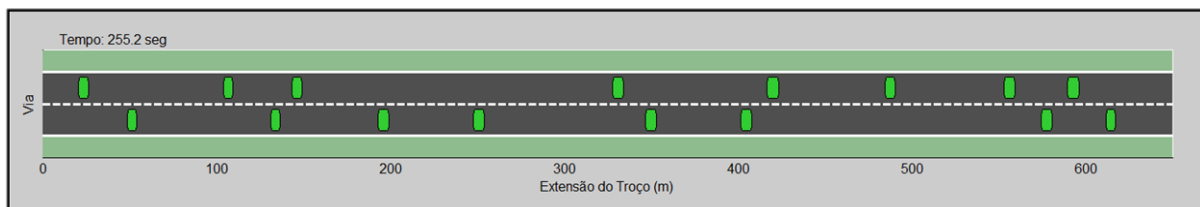


Figura 4.19 – Modelo em Matlab: Aspeto geral do simulador

O programa demonstra um funcionamento estável para fluxos reduzidos e intervalos de tempo inferiores a 0,2 segundos por iteração. No entanto, o estudo de um modelo de *car-following* tem maior interesse com fluxos elevados onde a interação entre veículos é mais frequente e, neste cenário, a *performance* demonstrada pelo *Matlab* na gestão de uma maior quantidade de veículos revelou-se insatisfatória, devolvendo uma simulação lenta e pouco fluída.

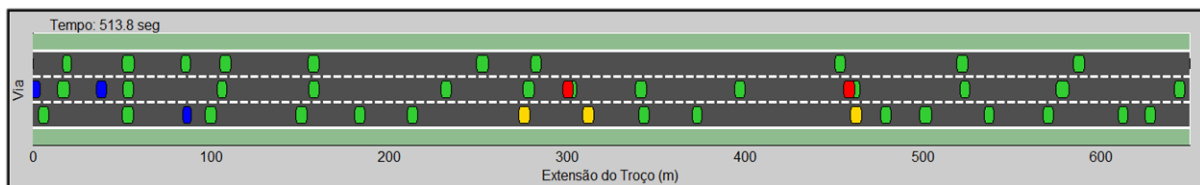


Figura 4.20 – Modelo em Matlab: Fluxo reduzido

Aumentando o intervalo de tempo por iteração, é possível obter velocidades de simulação mais satisfatórias, mas verifica-se uma perda substancial na precisão do modelo com a ocorrência de sobreposição de veículos e consequente incoerência na determinação do veículo em frente, o que se traduz num modelo pouco fiável.

Esta perda de resolução é resultante do facto de que o estado de aceleração de cada veículo ser calculado relativamente às condições presentes no instante t no tempo de

simulação e a correção na aceleração do veículo ser tomada no instante t_{+1} . Para intervalos de tempo muito grandes, as correções planeadas deixam de ser aplicáveis ao instante seguinte.

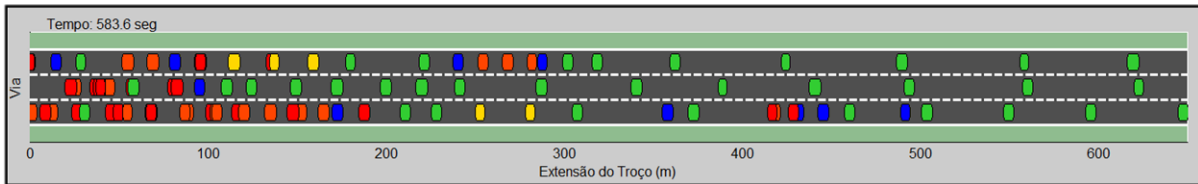


Figura 4.21 – Modelo em Matlab: Fluxo elevado

Ainda que constitua um modelo com interesse para o estudo e desenvolvimento de algoritmos de modelação de tráfego, não representa uma modelo prático para a análise de cenários reais quer pela *performance* demonstrada, quer pela limitação da rede a apenas um troço retilíneo.

4.3.2.8. IMPLEMENTAÇÃO NO TRFS

A implementação do modelo de *Wiedemann* em *Java* replica as considerações tomadas no desenvolvimento do modelo em *Matlab*, com a adição de uma estrutura hierárquica que permita a implementação de diferentes modelos.

Qualquer modelo de *car-following* no TRFS é herda da classe *CarFollowingModel*, que atribui aos seus herdeiros as variáveis dX e dV , comuns a qualquer modelo, e obriga-os a possuir o método *update()* que o programa pode chamar em cada iteração independentemente do modelo seleccionado pelo utilizador.

```
|public class W74CarFollowing extends CarFollowingModel {
```

Adicionalmente, introduz-se o conceito de parâmetro de simulação dinâmico, denominado no TRFS por *DynamicSimParam* que oferece a possibilidade estabelecer valores mínimos e máximos dentro dos quais o utilizar tem liberdade de manipulação.

```
|name, min, max, default, increment, decimalformat
|public static DynamicSimParam
|  AXadd = newDynamicSimParam("AXmult",0f,10f,1.25f,0.05f,"0.00",""),
|  AXmult = newDynamicSimParam("AXmult",0f,10f,2f,0.05f,"0.00",""),
|  BXadd = newDynamicSimParam("BXadd",0f,10f,2f,0.05f,"0.00",""),
|  BXmult = newDynamicSimParam("BXmult",0f,10f,1f,0.05f,"0.00",""),
|  CXconst = newDynamicSimParam("CXconst",0f,10f,5f,0.05f,"0.00",""),
|  CXadd = newDynamicSimParam("CXadd",0f,10f,4f,0.05f,"0.00",""),
|  CXmult = newDynamicSimParam("CXmult",0f,10f,3.6f,0.05f,"0.00",""),
|  EXadd = newDynamicSimParam("EXadd",0f,10f,1.5f,0.05f,"0.00",""),
|  EXmult = newDynamicSimParam("EXmult",0f,10f,0.55f,0.05f,"0.00",""),
|  OPDVadd = newDynamicSimParam("OPDVadd",0f,10f,1.5f,0.05f,"0.00",""),
|  OPDVMult = newDynamicSimParam("OPDVMult",0f,10f,1.5f,0.05f,"0.00",""),
|  BNULLmult = newDynamicSimParam("BNULLmult",0f,10f,0.1f,0.05f,"0.00",""),
```

```

FAKTORVmult = newDynamicSimParam("FAKTORVmult",0f,10f,1f,0.05f,"0.00",""),
BMAXmult = newDynamicSimParam("BMAXmult",0f,10f,0.1f,0.05f,"0.00","");

```

A implementação final no TRFS, em Java, toma a seguinte forma:

```

dX = leader.physics.position.dst(vehicle.physics.position);
dV = leader.physics.speed - vehicle.physics.speed;

float length1 = leader.config.length;
float Vn1 = leader.physics.speed;
float an1 = leader.physics.acceleration;

AX = length1 + AXadd.getCurrentVal() + RND1 * AXmult.getCurrentVal();
BX = (float) ((BXadd.getCurrentVal() + BXmult.getCurrentVal() * RND1) * Math
    .sqrt(dV > 0 ? Vn : Vn1));
ABX = AX + BX;

if (dX <= ABX) {
    state = 1;
} else {
    EX = EXadd.getCurrentVal() + EXmult.getCurrentVal() * (NRND - RND2);
    SDX = AX + EX * BX;

    CX = CXconst.getCurrentVal() * (CXadd.getCurrentVal() +
        CXmult.getCurrentVal() + (RND1 + RND2));
    SDV = (float) Math.pow(((dX - length1 - AX) / CX), 2);

    if (dX < SDX) {
        OPDV = SDV * (-OPDVadd.getCurrentVal() -
            OPDVmult.getCurrentVal() * NRND);

        if (dV > SDV) state = 2;
        else if (dV > OPDV) state = 3;
        else state = 4;

    } else if (dX < SDV && dX < 50 * Math.sqrt(dV)) state = 2;
    } else state = 4;
}

switch (state) {
case 1:
    w74Acceleration = (float) ((0.5 * (Math.pow(dV, 2) / (ABX - dX - length1)))
        + an1 + (-20 + 1.5 * Vn / 60) * ((ABX - dX - length1) / BX));
case 2:
    w74Acceleration = (float) Math.min(0.5 * (Math.pow(dV, 2) /
        (ABX - dX - length1)) + an1, (-20 + 1.5 * Vn / 60));
case 3:
    w74Acceleration = dV > OPDV ? 1 : -1 * BNULLmult.getCurrentVal()
        * (RND3 + NRND);
case 4:
    FaktorV = maxSpeed / ((desiredSpeed) + FAKTORVmult.getCurrentVal() *
        (maxSpeed - desiredSpeed));
    w74Acceleration = BMAXmult.getCurrentVal() * (maxSpeed - Vn * FaktorV);
}

```

4.3.3. LANE CHANGING

A implementação de um modelo de *lane changing* implica uma infraestrutura de base preexistente que permita inquirir o veículo quanto à sua vizinhança, para que seja possível construir condições que determinem a vontade ou necessidade do condutor de mudar de via. O programa tem de ser capaz de informar o modelo de *lane changing* quanto à existência de um veículo líder na mesma via e veículos que se encontrem na via adjacente.

No decorrer da construção desta infraestrutura foram estabelecidas regras e condições para testar o funcionamento dos modelos de detecção de proximidade que constituem por si só um modelo simplificado de *lane changing*. Denominado como *TRFS Lane changing*, é o modelo que, de momento, modela as operações de mudança de via no TRFS.

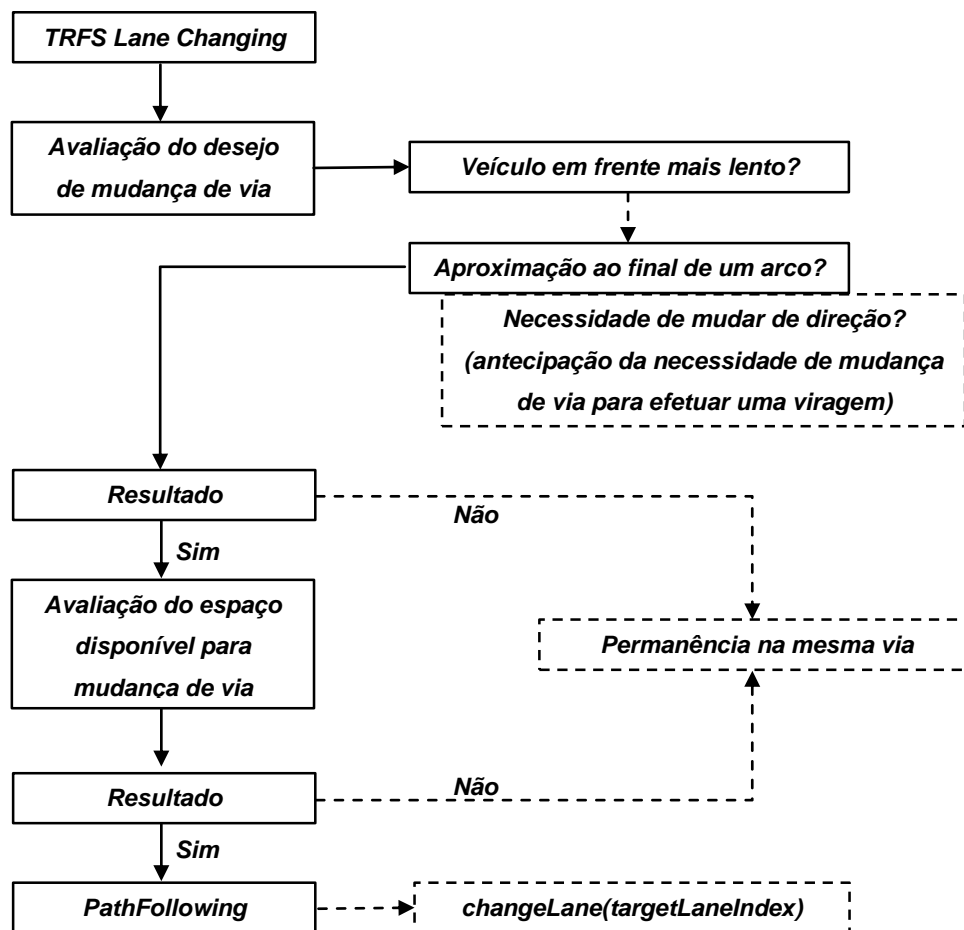


Figura 4.22 – Lógica de avaliação do modelo de mudança de via TRFS Lane changing

Programaticamente, o modelo toma a forma:

```

public void update() {
    PathFollowing pF = vehicle.behavior.pathFollowing;
    desireToChange = false;

    if (leader != null) {
        if (leader.physics.speed < vehicle.physics.speed) {
            desireToChange = true;
            //Moves one lane to the left to overtake
            targetLaneIndex = pF.state.currentLane.index + 1;
        } else desireToChange = false;
    }

    //Might need to change lanes to for an intersection
    if (pF.state.approachingLinkEnd && pF.state.currentLink.toNode.toLinks.size > 1) {
        if (pF.nextLink().toTheRightOf(pF.state.currentLink) &&
            pF.state.currentLane.index > 0) {
            desireToChange = true;
            targetLaneIndex = pF.state.currentLane.index - 1;
        } else if (pF.nextLink().toTheLeftOf(pF.state.currentLink) &&
            pF.state.currentLane.index < pF.state.currentLink.nrOfLanes - 1) {
            desireToChange = true;
            targetLaneIndex = pF.state.currentLane.index + 1;
        }
    }

    if (desireToChange) {
        //Check GapAccepance
        float rearGap = 2, frontGap = 1;
        boolean rearPass = false, frontPass = false;
        gapAccepted = false;
        if (rearOnTargetLane != null) {
            if ((rearOnTargetLane.behavior.pathFollowing.state.distanceOnPath +
                rearOnTargetLane.config.length/2) < pF.state.distanceOnPath -
                vehicle.config.length/2 - rearGap)
                rearPass = true;
        } else rearPass = true;
        if (frontOnTargetLane != null) {
            if ((frontOnTargetLane.behavior.pathFollowing.state.distanceOnPath -
                rearOnTargetLane.config.length/2) > pF.state.distanceOnPath -
                vehicle.config.length/2 + frontGap ) frontPass = true;
        } else frontPass = true;
        if (rearPass == frontPass == true) gapAccepted = true;
    }

    if (desireToChange && gapAccepted) {
        if (targetLaneIndex > 0 && targetLaneIndex < pF.state.currentLink.lanes.size)
            pF.changeLane(targetLaneIndex);
    }
}

```

4.4. REPRESENTAÇÃO GRÁFICA

A representação gráfica da rede e dos seus elementos é executada percorrendo todos os *Links* e *Nodes* do mapa e chamando a função *render()* em cada um destes objetos. A representação dos veículos está associada ao arco onde se encontram, para que seja respeitada a ordem de desenho desejada.

De modo a respeitar a posição altimétrica das vias, os arcos são organizados em listas de acordo com o atributo *z* definido par cada arco na rede em *S/G*.

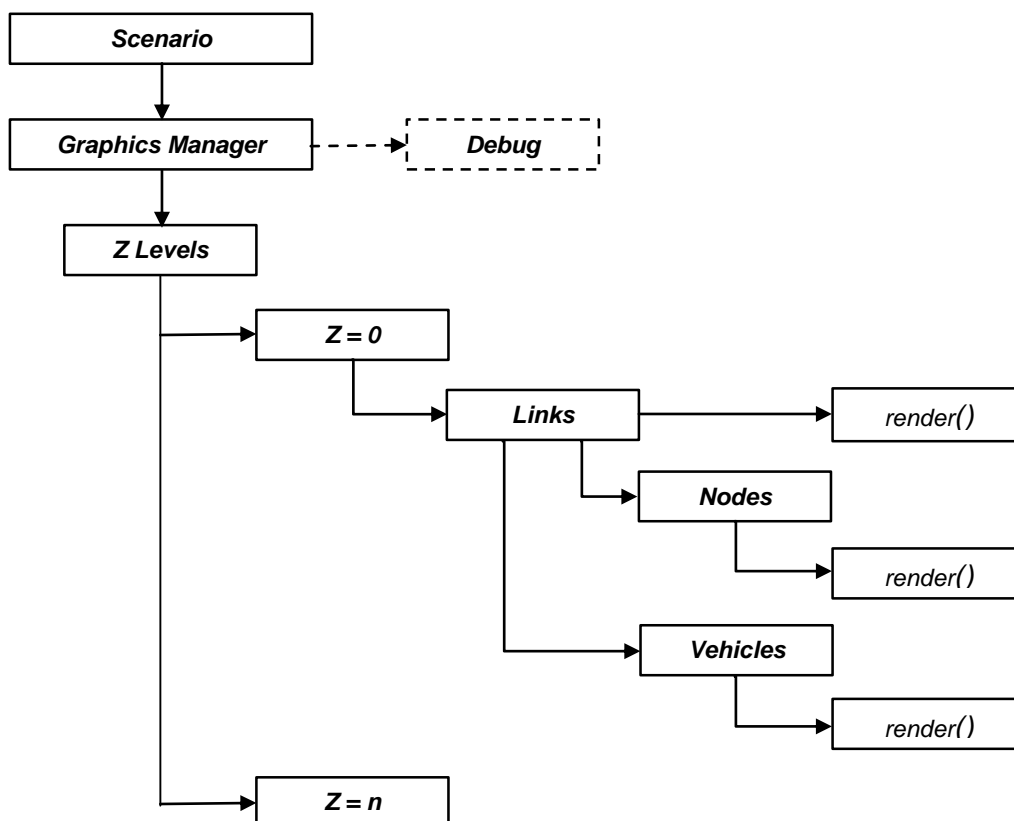


Figura 4.23 – Sequência de desenho do cenário

No momento da criação de cada arco, são calculados dois arcos paralelos com um afastamento correspondente a metade da largura da via. Estes novos elementos são utilizados para a triangulação e definição das áreas a “pintar” com a cor pretendida para as vias.

Adicionalmente, caso o utilizador pretenda, é possível desenhar a rede em modo de depuração (*debug*), onde são representados os elementos estruturais do mapa e dos veículos, possibilitando a deteção de erros geométricos que seriam impercetíveis de outro modo.

4.5. RESULTADOS

O programa desenvolvido constitui apenas o primeiro passo na construção de um simulador de tráfego. Ainda que tenha sido desenvolvido com um fim académico, a modelação realista da interação de veículos exige um significativo esforço de calibração, a adição de diversos elementos de controlo de tráfego e a programação do seu efeito nos condutores, bem como a previsão de inúmeras condições de tráfego não contempladas por modelos de *car-following* ou *lane changing*.

No seu estado atual, o TRFS apresenta-se notoriamente mais apto à modelação de condições de condução em autoestrada do que à modelação de condições urbanas. A imatura interface de interação entre o veículo e rede não proporciona uma reação realista à geometria do traçado, como o abrandamento à chegada a uma interceção ou a alteração de comportamento face à existência de sinalização.

Por outro lado, por uma questão de eficiência, não são modeladas interações físicas entre veículos tais como colisões e reações, pelo que é frequente a sobreposição de veículos quando o tempo de travagem não é suficiente para evitar o embate. Naturalmente, este tipo de fenómeno tem origem na calibração prematura do programa e terá tendência a ser mitigado, mas entende-se que a adição de um sistema de deteção e simulação de colisões seria vantajoso no futuro, servindo como uma ferramenta de auxílio à identificação de problemas nos modelos implementados no programa.

Apesar do estado primitivo de desenvolvimento do TRFS, é pertinente a apresentação de alguns resultados que permitam demonstrar as suas possibilidades, pelo que se procedeu à simulação de uma situação real e análise dos resultados obtidos.

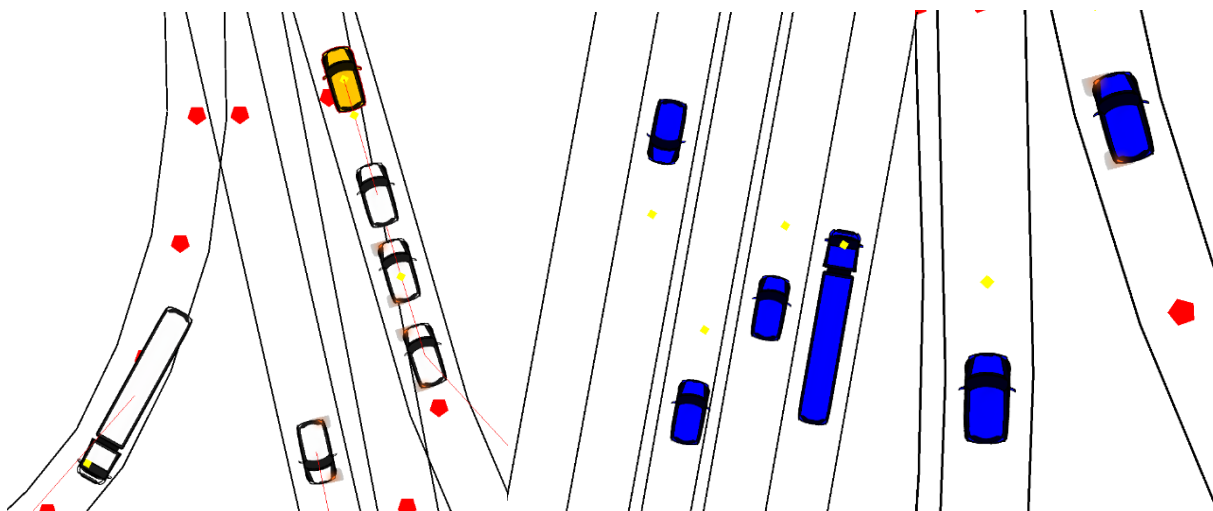


Figura 4.24 – Ampliação dos agentes autónomos do TRFS

4.5.1. RECOLHA DE DADOS

O TRFS oferece a possibilidade de *output* de dados recolhidos por contadores implementados nos arcos desejados. Em cada simulação, o programa cria um arquivo CSV que vai atualizando à passagem de cada veículo nos locais seleccionados. Em cada contagem são registados não só dados comumente obtíveis em contagens de campo, tais como o tipo de veículo (ligeiro ou pesado), a sua origem, destino e velocidade, mas também a aceleração do veículo, tempo de viagem, distancia percorrida e ainda valores difíceis de quantificar em condições reais como o tempo total a travar, acelerar ou imóvel e o tempo passado com o desejo de mudar de via.

Para esta demonstração, optou-se pela simulação do nó 13 do IC19/A37 de ligação com a N249-4. A geometria foi isolada e importada diretamente de uma rede SIG obtida de dados abertos do *OpenStreetMap*. O TRFS utiliza a informação disponível no ficheiro importado para construir a rede, obtendo-se o resultado da Figura 4.25.

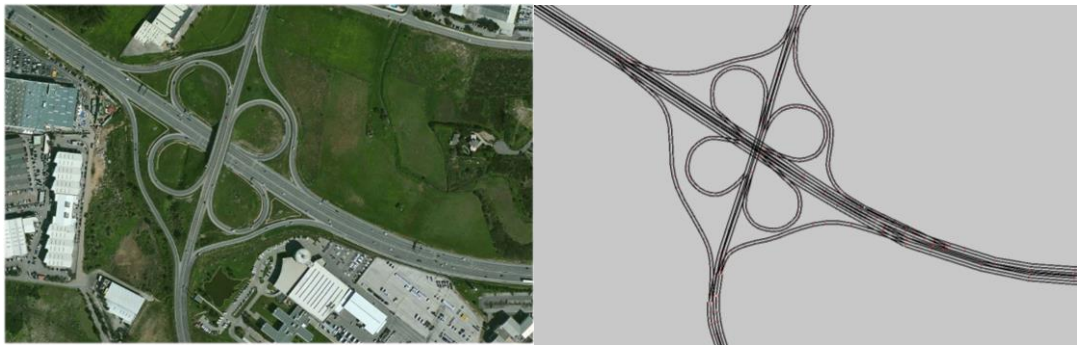


Figura 4.25 – Comparação da rede em fotografia aérea com a rede importada pelo TRFS.

De seguida, foram implementados contadores nos 14 arcos numerados na Figura 4.26. A numeração representada corresponde aos ID's dos arcos originários da rede SIG importada e permite a identificação dos locais de cada contagem no ficheiro de *output*.

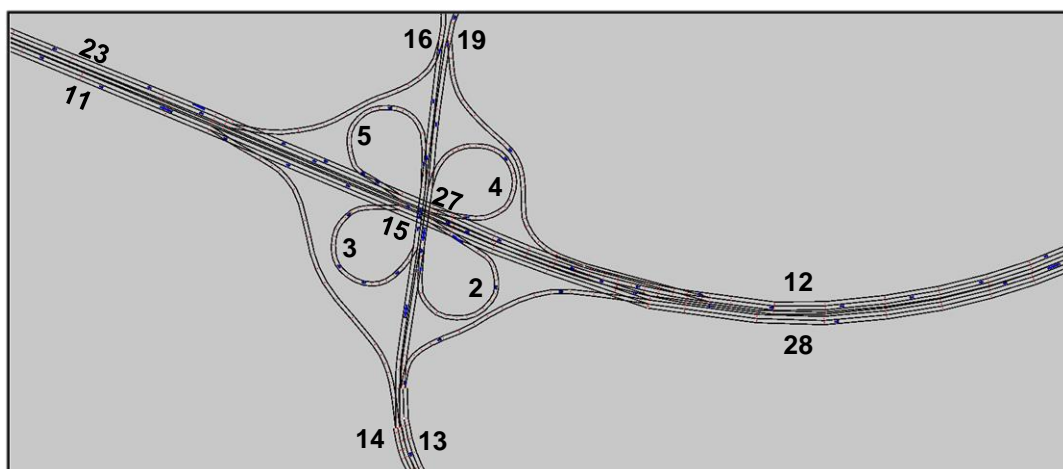


Figura 4.26 – ID's dos arcos seleccionados para a extração de dados.

Por fim, foi corrida a simulação com os seguintes condições iniciais:

Tabela 4-1 – Condições iniciais da simulação de demonstração.

ARCO	FLUXO (VEIC/H)	PESADOS	TEMPO (SEG)
11	2500	10%	1000
12	2500		
13	1060		
16	1060		

4.5.2. ANÁLISE DOS RESULTADOS

O documento de *output* contém uma entrada para cada leitura efetuada na rede, o que implica a contagem do mesmo veículo em diversos locais da rede. Em 1000 segundos de simulação (16 minutos e 40 segundos), obtiveram-se 5880 registos correspondentes a cerca de 1900 veículos.

Filtrando os resultados da simulação pelas leituras efetuadas em arcos de saída da rede, e agregando os veículos em função do seu arco de origem e destino, é possível a construção da matriz OD da Tabela 4-2.

Tabela 4-2 - Matriz OD da rede.

O/D	14	19	23	28	TOTAL
11	196	157	148	174	675
12	169	166	157	168	660
13	80	65	66	67	278
16	63	77	67	75	282
Total	508	465	438	484	1895

Similarmente, é possível obter uma matriz referente ao tempo médio de viagem (Tabela 4-3) e velocidade média por par OD (Tabela 4-4) em cada par OD, valores passíveis de comparar com dados reais obtidos em campo.

Tabela 4-3 – Tempo médio de viagem por par OD (segundos).

O/D	14	19	23	28	TOTAL
11	16.06	25.36	33.50	18.87	22.77
12	43.72	32.50	34.84	52.34	40.98
13	41.23	24.14	32.27	22.27	30.54
16	13.32	28.54	9.34	21.85	18.80
Total	28.88	28.26	30.10	31.42	29.66

As velocidades médias obtidas são excessivamente altas devido ao reduzido fluxo introduzido na rede e conseqüente falta de obstruções na circulação dos agentes. No entanto, é notável a necessidade de calibração dos parâmetros que regulam as velocidades desejadas de cada condutor, uma vez que aparentam produzir valores demasiadamente elevados.

Tabela 4-4 – Velocidade média por par OD (Km/h).

O/D	14	19	23	28
11	120.85	116.46	115.52	118.92
12	119.20	122.49	122.63	116.81
13	115.65	120.35	118.45	120.47
16	101.23	103.70	94.40	102.86

Uma vez que o TRFS permite o registo de qualquer parâmetro físico ou psicológico envolvido na modelação do comportamento dos condutores, torna-se também possível a quantificação de parâmetros dificilmente acessíveis em condições de campo, tais como o tempo médio em travagem (Tabela 4-5) e o tempo médio que os condutores passam com o desejo de mudar de via em cada par OD (Tabela 4-6).

Tabela 4-5 – Tempo médio em travagem por par OD (segundos).

O/D	14	19	23	28
11	0.00	0.33	0.29	0.43
12	0.44	0.00	0.01	0.59
13	0.44	0.21	0.18	0.00
16	0.11	0.38	0.00	0.38

Tabela 4-6 – Tempo médio com o desejo de mudar de via por par OD (segundos).

O/D	14	19	23	28
11	1.20	3.34	4.19	2.19
12	3.65	1.18	1.17	4.97
13	2.80	1.37	1.92	0.27
16	0.12	1.25	0.04	1.50

O acesso a estes dados abre a possibilidade de relacionar as características físicas da infraestrutura e as condições de escoamento do tráfego com o estado psicológico dos condutores.

Por outro lado, expandido a recolha de dados à generalidade da rede, é possível utilizar os registos da simulação para a produção de *outputs* gráficos numa aplicação SIG, que facilite a visualização de quantificadores como o fluxo (Figura 4.27) e a densidade (Figura 4.28) na rede.



Figura 4.27 – Representação do fluxo (Veic/h) na rede.

Neste caso, a uniformidade das velocidades praticadas na rede implica uma densidade variável proporcionalmente ao fluxo, pelo que o mapa resultante apresenta um grafismo semelhante.



Figura 4.28 – Representação da densidade do tráfego (Veic/Km) na rede.

5. CONCLUSÕES

As ferramentas de simulação de tráfego representam uma evolução significativa na forma como a engenharia de transportes avalia e toma decisões acerca do presente e futuro de infraestruturas de transporte, a qualquer modo ou escala.

Com o desenvolvimento de programas de simulação que operam tanto à escala macro, como à escala micro, torna-se conveniente para entidades gestoras manter um modelo que englobe toda a rede a seu cargo, criando interligações a infraestruturas de carregamento de tráfego, gestão de sinais, listagem de ocorrências ou incidentes, que sejam atualizados em tempo real. Um sistema integrado a este nível permite não só uma maior eficiência no planeamento da operação e manutenção da rede, como possibilita a previsão da influência de alterações futuras em todos os elementos da rede, com base em dados constantemente atualizados.

No entanto, do ponto de vista do realismo de simulação, existe ainda algum caminho a percorrer na modelação de determinadas condições, nomeadamente:

- Interação entre automóveis e veículos de duas rodas na mesma via;
- Interação entre automóveis e outros veículos estacionados na berma;
- Interação entre automóveis e tráfego pedonal na berma ou passeios de largura reduzida;
- Ultrapassagens em vias de dois sentidos;

Adicionalmente, com o crescente interesse no aproveitamento de *Big Data*, é expectável o aparecimento de ferramentas de simulação que incorporem diretamente os dados de tráfego gerados em tempo real por equipamentos *GPS* e dados móveis.

Em função do formato em que este tipo de dados sejam disponibilizados, poderá ocorrer uma mudança significativa na forma como os mesmos são introduzidos e validados. De modo a proteger a privacidade da população, a aplicação destes dados na afetação de tráfego de uma rede poderá ter de acontecer de forma automática com base na georreferenciação dos pontos de entrada, negando ao modelador a possibilidade de observar a sua validade.

Por outro lado, com o desenvolvimento dos veículos autónomos, a modelação da interação entre um veículo conduzido por uma inteligência artificial e outro afeto ao comportamento humano torna-se muito relevante. Com o incremento da percentagem de veículos autónomos na rede pública, são de prever sérias alterações na relação entre o fluxo e o nível de serviço perceptível na rede. Do ponto de vista da engenharia de transportes, é

importante o acompanhamento desta evolução e a modelação dos seus efeitos na infraestrutura de forma proactiva e não reativa às suas consequências.

Neste âmbito, existe um vazio nas teorias de *car-following*, evidenciando-se a necessidade de desenvolver um modelo comportamental de tráfego que aborde a relação entre condutores humanos e inteligência artificial.

Neste documento, efetuou-se uma revisão do estado da arte da simulação de tráfego, expondo as suas aplicações na comparação de geometrias a adotar em novas infraestruturas, no auxílio a propostas de alteração em infraestruturas de transporte e nos processos de tomada de decisão e divulgação pública. Neste contexto, foi desenvolvida uma revisão das restantes metodologias de análise de tráfego e do seu enquadramento nas diferentes fases do projeto de uma infraestrutura rodoviária (modelos de estimação de procura – fases de planeamento e projeto; metodologias analíticas – fases de projeto e operação; otimização de semáforos – fases de projeto e operação; simulação de tráfego – fases de projeto e operação).

Foram enumerados os diferentes tipos de simulação (macroscópica, mesoscópica, microscópica, híbrida e nanoscópica), expondo os princípios nos quais se baseia o seu procedimento de cálculo, bem como as respetivas aplicabilidades em diferentes contextos de escala e dimensão da rede (simulação macroscópica – escala regional e mais de 10000 arcos; simulação microscópica – pequena área e menos de 1000 arcos), período (simulação macroscópica – 24 horas; simulação microscópica – período de ponta) e procura (simulação macroscópica – mais de 1 milhão de veículos; simulação microscópica – menos de 200 mil veículos).

Seguidamente, foi apresentada uma revisão de dezasseis programas de simulação de tráfego atualmente disponíveis no mercado, expondo as respetivas tipologias de licenciamento e aplicabilidade às diferentes escalas de simulação, bem como uma revisão detalhada dos programas comerciais *Vissim*, *Aimsun*, *Paramics*, *Transmodeler* e *Dynasim*, comparando as suas funcionalidades no que diz respeito às geometrias de rede possíveis de modelar, aos agentes de tráfego que integram a simulação e às possibilidades de controlo e afetação de tráfego.

Foi descrita uma metodologia geral de desenvolvimento de modelos de simulação de tráfego, explorando as etapas de definição do âmbito da análise, recolha e preparação de dados, desenvolvimento do modelo, verificação de erros, calibração do modelo, análise de soluções e comparação de resultados, relatório final e documentação técnica.

Foram explorados os modelos de *car-following* implementados em três programas comerciais (*Wiedemann '74 – Vissim; Gipps – Aimsun; Fritzsche – Paramics*), apresentando o seu conceito e formulação e os modelo que deram origem às formulações atualmente usadas em programas comerciais: *GHR, CA, Helly, AP* e lógica difusa.

Finalmente, foi apresentado o desenvolvimento de um programa de simulação microscópica de tráfego, o *TRFS*, construído de raiz na linguagem de programação *Java*, que permite a implementação de modelos matemáticos de comportamento de agentes, com um foco na visualização do efeito dos seus parâmetros de calibração no tráfego, em tempo real.

O *TRFS* é desenvolvido numa perspetiva de providenciar uma plataforma que permita a implementação e teste de modelos comportamentais de tráfego de forma tão simples quanto possível, sem as preocupações adicionais de um programa comercial.

Perspetivas futuras

Apesar de se encontrar ainda num estágio inicial de desenvolvimento, considera-se que o programa apresentado representa um grande potencial de uma evolução continua, nomeadamente no meio académico, com a adição de inúmeras funcionalidades:

- Introdução de novos modelos comportamentais de *car-following*;
- Atribuição de modelos comportamentais distintos a diferentes veículos, de modo a visualizar o modo como se relacionam durante a simulação;
- Implementação de um editor de redes incorporado, que permita efetuar correções à rede importada ou criar uma rede de raiz, eliminando a dependência de um programa *SIG* externo;
- Programação de elementos de controlo de tráfego, como sinais de *STOP*, cedências de passagem e semáforos;
- Implementação de um sistema de análise de dados que permita quantificar as condições da rede ao longo ou no final da simulação, possibilitando, desta forma, a sua representação gráfica diretamente no simulador.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] VISSIM - PTV Planung Transport Verkehr, Karlsruhe [Consult. em 2014]. URL: vision-traffic.ptvgroup.com/en-uk/products/ptv-vissim/
- [2] Towards Sustainable Transportation - Em **OECD Proceedings**. Vancouver : OECD, 1996 URL: oecd.org/greengrowth/greening-transport/2396815.pdf
- [3] FHWA; CAMBRIDGE SYSTEMATICS INC; TEXAS TRANSPORTATION INSITITUTE - **Traffic Congestion and Reliability - Trends and Advanced Strategies for Congestion Mitigation**. Cambridge : Cambridge Systematics, Inc., 2005 URL: ops.fhwa.dot.gov/congestion_report/congestion_report_05.pdf
- [4] PERSULA, M. - Simulation of Traffic Systems - An Overview. **Journal of Geographic Information and Decision Analysis**. Berlin, Heidelberg. ISSN 1435-5949 , Vol. 3, p. 1–8.
- [5] DIETMAR P. F. MÖLLER - Introduction to Transportation Analysis, Modeling and Simulation. . London : Springer-Verlag London, 2014. (1. ed.) URL: <http://www.springer.com/cn/book/9781447156369> , ISBN 978-1-4471-5637-6.
- [6] **Highway Capacity Manual 2010** - . Washington, D.C. : Transportation Research Board, 2010. , ISBN 978-0-309-16077-3.
- [7] TransModeller - Caliper, Newton MA [Consult. em 2015]. URL: <http://www.caliper.com/>
- [8] Cube Dynasim - Citilabs, Lafayette [Consult. em 2015]. URL: <http://www.citilabs.com/>
- [9] Aimsun - TSS-Transport Simulation Systems, Barcelona [Consult. em 2015]. URL: <http://www.aimsun.com/>
- [10] Paramics - Quadstone Paramics, Edinburgh [Consult. em 2015]. URL: <http://www.paramics-online.com/>
- [11] DOWLING, R.; SKABARDONIS, A.; ALEXIADIS, V. - Traffic Analysis Toolbox Volume III: Guidelines for Applying Traffic Microsimulation Modeling Software. . McLean, VA : Federal Highway Administration, U.S. Department of Transportation, 2004. URL: http://ops.fhwa.dot.gov/trafficanalysistools/tat_vol3/vol3_guidelines.pdf
- [12] GAZIS, D. C.; HERMAN, R.; ROTHERY, R. W. - Nonlinear Follow-the-Leader of Traffic Flow. **Operations Research**. Catonsville. ISSN 1526-5463 , Vol. 9(4), p. 545–567. DOI:10.1287/opre.9.4.545.
- [13] GIPPS, P. G. - A Behavioural Car-Following Model for Computer Simulation. **Transportation Research Part B: Methodological**. Melbourne, Australia. , Vol. 15(2), p. 105–111. DOI:10.1016/0191-2615(81)90037-0.
- [14] KOMETANI, E.; SASAKI, T. - Dynamic Behaviour of Traffic with a Nonlinear Spacing-Speed Relation, Proceedings of The Symposium on Theory of Traffic Flow. Em **Theory of Traffic Flow, Proceedings**. Amsterdam Netherlands : Robert Herman, 1959 (pp. 105–119).
- [15] HELLY, W. - Simulation of Bottlenecks in Single-Lane Traffic Flow. Em **Theory of Traffic Flow**

- Symposium. Proceedings.** Amsterdam : Elsevier Publishing Company, 1961 URL: <http://trid.trb.org/view.aspx?id=115225> (pp. 207–238).
- [16] KIKUCHI, S.; CHAKROBORTY, P. - Car-Following Model Based on Fuzzy Inference System. **Transportation Research Record**. Washington. ISSN 0361-1981 , p. 82–91.
- [17] WIEDEMANN, R.; REITER, U. - Microscopic Traffic Simulation: The Simulation System MISSION. **ICARUS (V1052) Final Report**. Brussels. , Vol. 2, p. 1–53.
- [18] FRITZSCHE, H.-T. - A Model for Traffic Simulation. **Traffic Engineering & Control**. Stuttgart. ISSN 0041-0683 , Vol. 35(5), p. 317–321.
- [19] VASCONCELOS, L.; SECO, Á.; SILVA, A. B. - Hybrid Calibration of Microscopic Simulation Models. *Computer Based Modelling and Optimization in Transportation, Advances in Intelligent Systems and Computing*. [s.l.] : Springer International Publishing, 2014. URL: <http://www.estgv.ipv.pt/PaginasPessoais/vasconcelos/Documentos/HybridCalibrationSpringerLV2WEB.pdf> , ISBN 978-3-319-04630-3.
- [20] CARLOS, L. A. - Probabilistic Safety Analysis Using Traffic Microscopic Simulation. . Lisboa : Instituto Superior Técnico, 2014.
- [21] MACEDO, J. M.; BENTA, A.; PICADO-SANTOS, L. - Utilização Da Microsimulação No Desenvolvimento De Uma Metodologia De Avaliação Dos Níveis De Serviço Em Autoestradas. Em **Actas do 7º Congresso Rodoviário Português**. Lisboa : CRP, 2013
- [22] MACEDO, J. M.; BENTA, A. - Seleção Da Tipologia de Uma Intersecção Utilizando um Simulador Microscópico de Tráfego - Um Caso De Estudo. Em **Actas do 7º Congresso Rodoviário Português**. Lisboa : CRP, 2013
- [23] VASCONCELOS, L.; SECO, Á.; SILVA, A. B. - Comparison of Procedures to Estimate Critical Headways at Roundabouts. **PROMET - Traffic & Transportation**. Zagreb. , Vol. 25(1), p. 43–54.
- [24] CAETANO, P. - Propostas de Intervenção para Carreiras de Transportes Colectivos em Troços de Reduzida Velocidade Comercial. . Lisboa : Instituto Superior Técnico, 2011.
- [25] FERREIRA, G. - Estudo do desempenho de uma rotunda da Cidade de Aveiro usando a microsimulação de tráfego. amferreira.eu
- [26] PINTO FERREIRA, J. - Aplicação De Modelos De Tráfego (Dracula) Na Avaliação Dos Impactos Em Redes Rodoviárias Urbanas Resultantes Das Alterações Nas Infra-Estruturas. . Porto : Universidade do Porto, 2008.
- [27] DRACULA - Institute for Transport Studies, University of Leeds, Leeds [Consult. em 2015]. URL: <http://www.its.leeds.ac.uk/software/dracula/>
- [28] VILARINHO, C. A. T. - Calibração de Modelos Microscópicos de Simulação de Tráfego em Redes Urbanas. . Porto : Universidade do Porto, 2008.
- [29] VASCONCELOS, L. - Modelos de Atribuição/Simulação de Tráfego: O Impacto na Qualidade dos Resultados de Erros no Processo de Modelação. . Coimbra : Universidade de Coimbra, 2004.
- [30] Saturn - Atkins, Surrey [Consult. em 2015]. URL: <https://saturnsoftware.co.uk>

- [31] TRANSYT - TRL Software, Berkshire [Consult. em 2015]. URL: https://trlsoftware.co.uk/products/junction_signal_design/transyt
- [32] ALEXIADIS, V.; JEANNOTTE, K.; CHANDRA, A. - Traffic Analysis Toolbox Volume I: Traffic Analysis Tools Primer. . McLean, VA : Federal Highway Administration, U.S. Department of Transportation, 2004. URL: ops.fhwa.dot.gov/trafficanalysistools/tat_vol1/vol1_primer.pdf
- [33] MCNALLY, M. G. - The Four Step Model. Em **Handbook of Transport Modeling**. 2. ed. Irvine : Emerald, Inc., 2008 URL: its.uci.edu/its/publications/papers/CASA/UCI-ITS-AS-WP-07-2.pdf ISBN 978-1-61344-939-4 (2. ed.).
- [34] SPENCER, P. - **Keep The Traffic Moving!: Details on the Mathematical Model** URL: <https://www.math.toronto.edu/mathnet/plain/carcompet/model.html>
- [35] BENDER, E. A. - An Introduction to Mathematical Modeling. . New York : Dover Publications, Inc., 1978. , ISBN 978-0486411804.
- [36] BOXILL, S. A.; YU, L. - **An Evaluation of Traffic Simulation Models for Supporting ITS Development**. Houston, Texas : Center for Transportation Training and Research, Texas Southern University, 2000 URL: <http://ntl.bts.gov/lib/17000/17500/17586/PB2001102338.pdf>
- [37] DOWLING, R. - Traffic Analysis Toolbox Volume VI: Definition, Interpretation and Calculation of Traffic Analysis Tools Measures of Effectiveness. . Washington, DC : Federal Highway Administration, U.S. Department of Transportation, 2007. (2007) URL: <http://ops.fhwa.dot.gov/publications/fhwahop08054/fhwahop08054.pdf>
- [38] MOECKEL, R. *et al.* - Microsimulation for integrated urban modelling. Em **Proceedings of the 10th International Conference on Computers in Urban Planning and Urban Management**. Iguassu Falls, Brazil : CUPUM, 2007 URL: spiekermann-wegener.de/pub/pdf/RMBSKSMW_CUPUM2007.pdf
- [39] DIA, H.; PANWAI, S. - Nanoscopic traffic simulations: Enhanced models of driver behavior for ITS and telematics simulations. Em **8th International Symposium on Transport Simulation**. QUEENSLAND, AUSTRALIA : Monash University, 2008
- [40] SBAYTI, H.; RODEN, D. - **Best Practices in the Use of Micro Simulation**. Arlington, VA : AECOM, 2010 URL: http://statewideplanning.org/wp-content/uploads/259_NCHRP-08-36-90.pdf
- [41] BLOOMBERG, L.; SWENSON, M.; HALDORS, B. - Comparison of Simulation Models and the HCM. Em **Transportation Research Board, 82nd Annual Meeting**. Washington, DC : Transportation Research Board, 2003 URL: ltrc.lsu.edu/TRB_82/TRB2003-001820.pdf
- [42] BRACKSTONE, M.; MCDONALD, M. - Car-following: a historical review. Em **Transportation Research Part F: Traffic Psychology and Behaviour**. Southampton, UK : Elsevier Ltd, 1999 ISBN 4417035946 (pp. 181–196).
- [43] MICHAELS, R. M. - Perceptual factors in car following. Em **Proceedings of the 2nd International Symposium on the Theory of Road-Traffic Flow**. Paris : OECD, 1965 (pp. 44–59).
- [44] REITER, U. - IMPIRICAL STUDIES AS BASIS FOR TRAFFIC FLOW MODELS. **PROCEEDINGS OF THE SECOND INTERNATIONAL SYMPOSIUM ON HIGHWAY CAPACITY**. Sydney, Australia. , Vol. 2, p. 493–502.

- [45] REKERSBRINK, A. - MICROSCOPIC TRAFFIC SIMULATION WITH FUZZY LOGIC. Em **Strassenverkehrstechnik**. Köln, Alemanha : Forschungsgesellschaft fuer Strassen- und Verkehrswesen (FGSV), 1995 URL: <http://trid.trb.org/view.aspx?id=494082> (Vol. 39, pp. 68–74).
- [46] HIGGS, B. J. - Application of naturalistic truck driving data to analyze and improve car following models. . Blacksburg, Virginia : Virginia Polytechnic Institute and State University, 2011. [Consult. 24 abr. 2014]. URL: <http://scholar.lib.vt.edu/theses/available/etd-12122011-152121>
- [47] PANWAI, S.; DIA, H. - Comparative evaluation of microscopic car-following behavior. **IEEE Transactions on Intelligent Transportation Systems**. ISSN 15249050 , Vol. 6(3), p. 314–325. DOI:10.1109/TITS.2005.853705.
- [48] SUN, D. J. - A lane-changing model for urban arterial streets. . Gainesville, FL : University of Florida, 2009.
- [49] BARCELÓ, J. - Fundamentals of Traffic Simulation. International Series in Operations Research & Management Science. . New York, NY : Springer New York, 2010. (Vol. 145) [Consult. 9 nov. 2013]. URL: <http://link.springer.com/10.1007/978-1-4419-6142-6> , ISBN 978-1-4419-6141-9.
- [50] SPARMANN, U. - Spurwechselforgänge auf zweispurigen BAB-Richtungsfahrbahnen. . Karlsruhe : Bundesminister für Verkehr, 1978. , ISBN 3-86509-537-2.
- [51] JOY, K. I. - **Chaikin's Algorithms for Curves**. Davis, California : University of California, 1996 URL: <http://www.idav.ucdavis.edu/education/CAGDNotes/Chaikins-Algorithm.pdf>
- [52] SAFFARZADEH, M.; REZAEI-ARJROODY, A. - Cost- Benefit Analysis of Electronic Toll Collection System in Iranian Freeways (Case Study : Tehran-Qom Freeway). **PIARC International Seminar on Intelligent Transport System (ITS) in Road Network Operations**. Kuala Lumpur, Malaysia.
- [53] DOWLING, R.; HOLLAND, J.; HUANG, A. - **Guidelines for Applying Traffic Microsimulation Modeling Software**. Oakland, California : California Department of Transportation, 2002
- [54] MENNENI, S.; SUN, C. - **Analysis of Wiedemann 74 and 99 Driving Behavior Parameters**. Columbia, MO : University of Missouri Columbia, [s.d.]

APÊNDICE A. MODELAÇÃO DE REDES EM VISSIM

A.1. ENQUADRAMENTO

O *Vissim*, programa comercial desenvolvido pela *PTV*, consiste num simulador de tráfego à escala microscópica baseado em modelos comportamentais de *car-following* e *lane-changing*, sendo frequentemente utilizado para análise e otimização de fluxos de tráfego. Ao oferecer a possibilidade de análise simultânea de condições urbanas e em autoestrada, proporciona a integração de transporte público e tráfego pedonal. Mais particularmente, admite a introdução de peões, ciclistas, motociclos, veículos ligeiros e pesados, autocarros, metro ligeiro e comboio, incluindo várias opções de personalização e calibração dos dados.

Alguns exemplos práticos de aplicação do *Vissim* são:

- Estudo de corredores de autoestrada para identificação de indicadores de desempenho em supressões de via;
- Estudo da implementação de *ramp metering* em autoestradas;
- Desenvolvimento de sistemas de gestão de tráfego em operações de manutenção na via;
- Estudo de corredores arteriais com interseções sinalizadas e não sinalizadas;
- Otimização da temporização de semáforos, com ou sem recurso a ferramentas externas;
- Estudo de esquemas de prioridades em infraestruturas com exigências multimodais, tais como operações de transporte público, passagens de peões ou vias reservadas.
- Avaliação do impacto da implementação de medidas de acalmia de tráfego;
- Apresentação *3D* de soluções alternativas para exibição a terceiros.

Neste apêndice, apresenta-se uma revisão das funcionalidades do *Vissim* e resumam-se as etapas necessárias à criação de um modelo de simulação através deste programa. Seguidamente, descreve-se a execução de um modelo de *VAO* na via de acesso à praça de portagens da Ponte 25 de Abril, efetuando-se a análise de diferentes alternativas, com vista à demonstração das diversas funcionalidades do *software*.



Figura A-1 – Logotipo *Vissim*

A.2. DADOS DE BASE PARA A SIMULAÇÃO

O potencial do *Vissim* reside na capacidade de simular o comportamento humano à escala individual, sendo um modelo de natureza estocástica, ou seja, baseado em acontecimentos aleatórios. No entanto, o comportamento de condutores e peões pode variar intensamente com fatores físicos, nomeadamente a capacidade das infraestruturas, o caráter rural ou urbano do local e o tipo e condições do pavimento, assim como com fatores inerentes ao contexto cultural onde a rede se insere, como o grau de agressividade ou civismo dos automobilistas face às várias situações com que se deparam no tráfego.

Por conseguinte, com vista a criar um modelo eficiente, é importante calibrar o funcionamento do programa, para que este se aproxime o mais possível da realidade em que a rede se insere.

No *Vissim*, os acontecimentos aleatórios são delineados por **Funções e Distribuições** (*Functions and Distributions*): que modelam fatores como a distribuição da velocidade, peso e potência dos veículos; **Tipos e Categorias de Veículos** (*Vehicle Type, Class and Category*): que definem os parâmetros associados a cada tipo de veículo (rodoviários e ferroviários) introduzido pelas distribuições acima referidas; **Parâmetros de Comportamento** (*Driving Behavior*): que, com base no modelo de *Wiedemann*, descrevem a atitude dos condutores em situações de condução livre (*free driving*), aproximação (*approaching*), seguimento (*following*) e travagem (*braking*); **Tipos de Arcos** (*Links*): nos quais se aplicam os aspetos supracitados à infraestrutura em estudo.

Neste relatório não se exploram todos os parâmetros que o programa admite para a afetação do modelo, pois a informação necessária para efetuar uma calibração relevante não está disponível. No âmbito desta exposição, os valores pré-definidos no *software* foram considerados adequados e suficientes já que, de acordo com o manual, se baseiam em padrões de tráfego europeus.

A.3. REDE

Uma rede no *Vissim* consiste em elementos estáticos, que se mantêm inalteráveis durante a simulação, e elementos dinâmicos, contendo informação acerca do tráfego a introduzir no sistema.

Os elementos estáticos representam a infraestrutura rodoviária, incluindo arcos, conectores, paragens de transporte público, semáforos e restante sinalização vertical.

À semelhança de outros *softwares* que trabalham com redes viárias, o elemento básico é o arco (*link*) que representa uma faixa de rodagem, com uma ou mais vias no mesmo sentido. Posteriormente, os arcos são ligados através de conectores (*connectors*), que permitem a continuidade do tráfego entre dois arcos. Por outro lado, dois arcos A e B que se cruzem, mas não incluem um conector na sua intersecção, indicam uma passagem superior ou inferior, não existindo fluxo de A para B.

Por elementos dinâmicos entende-se:

- Volumes de tráfego (*traffic volumes*) em cada arco de entrada na rede (incluindo as percentagens de cada tipo de veículo);
- Decisões de percursos a seguir dentro da rede (*routes*) para uma dada percentagem do tráfego contido em determinado arco, que podem ser definidas para diferentes períodos e tipos de veículo;
- Regras de prioridade (*right-of-way*) e zonas de conflito (*conflict areas*) para intersecções com e sem sinalização (no primeiro caso, atuando quando dois veículos se encontram dentro do cruzamento, fora da influência da sinalização);

Para introduzir novos elementos na rede, é necessário seleccionar o elemento desejado na barra de ferramentas:

Tabela A-1 – Barra de Ferramentas do Vissim

SIMB.	DESCRIÇÃO	SIMB.	DESCRIÇÃO
	Arcos e Conectores		Sinais de STOP
	MARCAÇÕES NO PAVIMENTO		SEMÁFOROS
	PERCURSOS		DETETORES
	LIMITES DE VELOCIDADE		PARAGENS DE PUT
	ZONAS DE VELOCIDADE REDUZIDA		LINHAS DE PUT
	PRIORIDADES		PARQUES DE ESTACIONAMENTO
	ZONAS DE CONFLITO		Nós

A.3.1. PLANO DE FUNDO

Tratando-se do estudo de um caso já existente, uma forma simples de iniciar a construção do modelo é utilizar um plano de fundo, na forma de fotografia aérea ou *CAD*, que sirva de base para o desenho dos arcos no *Vissim*. É possível modificar a posição da imagem, assim como alterar a sua escala, através da marcação de uma distância conhecida. A hipótese de

inserir fundos adicionais permite caracterizar em pormenor um determinado local ou até expandir a rede inicialmente analisada.

Adicionalmente, é boa prática alterar ligeiramente a imagem original de satélite, de modo a que apresente de forma clara a geometria longitudinal, o sentido dos arcos, o número de vias e a localização de semáforos, *STOPS* e de outros elementos pertinentes à representação da rede.

A.3.2. ARCOS E CONETORES

É oportuno referir certas linhas gerais a considerar, previamente à criação de um arco:

- Um arco não pode ter mais do que um perfil transversal, sendo necessário criar um novo arco quando se pretende modelar, a título de exemplo, uma supressão de via.
- Um arco funciona num só sentido, sendo necessário repetir a geometria no sentido oposto.

A.3.2.1. MODELAÇÃO DE ARCOS

Para adicionar um arco, pressiona-se o botão direito no local desejado e arrasta-se o rato até ao ponto final pretendido (ou seja, no sentido do tráfego). Para arcos com geometria curva, é possível simular a curvatura pressionando o botão direito do rato em cima do arco tantas vezes quanto o número de vértices necessários à definição da curva. Os pontos arrastam-se com o botão esquerdo do rato.

Após a criação de um arco, abre-se uma janela com as suas propriedades, onde é possível definir o nome do arco, o número de vias e as suas dimensões e ordenar a criação do sentido oposto, seleccionando-se o número de vias respetivas. É então criado um arco de geometria idêntica ao original, que deve posteriormente ser adaptado à geometria real.

Finalmente, é possível definir o tipo de comportamento dos condutores que se pretende para o arco.

A.3.2.2. INTRODUÇÃO DE CONETORES

Um conector é criado arrastando o botão direito do rato de um arco para o outro no sentido do tráfego. O *Vissim* desenha automaticamente a curvatura do conector baseando-se no número de pontos definido pelo utilizador.

À semelhança do que acontece na modelação de arcos, é possível definir o nome do conector e o seu comportamento.

Na janela de propriedades é necessário selecionar as vias a unir pelo conector, entre os dois arcos. O programa identifica as vias da direita para a esquerda, pelo que a designação *Lane 1* corresponderá sempre à via mais à direita. Note-se que o número de vias do arco a

montante deve corresponder, obrigatoriamente, ao número de vias a jusante. Por conseguinte, nos casos em que se dê a abertura de uma nova via é necessário adicionar um segundo conetor para esta.

Além disso, é possível escolher o tipo de viragem que o conetor permite, limitar o tipo de veículos que circulam em cada via, opção também disponível nos arcos e modelar o comportamento dos condutores na mudança de vias.

- **Emergency Stop** – corresponde à última posição no conetor em que o condutor ainda tentará mudar de via. A distância é calculada a montante do início do elemento. A este valor são somados 5 metros por cada via a atravessar e ainda 2,5 metros, nos casos com um número ímpar de vias.

Consequentemente, um veículo que pretenda entrar no conetor e necessite de atravessar da via 3 para a via 1 (do arco precedente), poderá fazê-lo até 15 metros antes da viragem.

$$5m_{\text{definido pelo utilizador}} + 2_{\text{vias}} \times 5m_{\text{por via}} \quad (A - 1)$$

- **Lane Change** – corresponde à distância antes do início do conetor a que um veículo inicia a tentativa de mudança de via(s). A caixa *per lane* indica se o valor introduzido é por via ou se corresponde à globalidade das vias no conetor.

A.3.2.3. ALTIMETRIA

Adicionalmente, é possível acrescentar uma componente altimétrica ao modelo, relevante não só pela importante questão gráfica, mas também na perspetiva de aproximar o modelo à realidade, uma vez que o *Vissim* pondera a inclinação dos arcos para modelar a aceleração (positiva e negativa), consumos e emissões dos veículos.

Os dados altimétricos são inseridos no menu **Display** da janela de propriedades do arco pretendido (selecionando-o através de duplo clique), onde é possível inserir a respetiva cota inicial e final. Uma vez que a representação é feita somente com base em dois pontos, é essencial dividir (*Split*) os arcos em tantos trainéis quanto necessário para uma representação fiel da realidade.

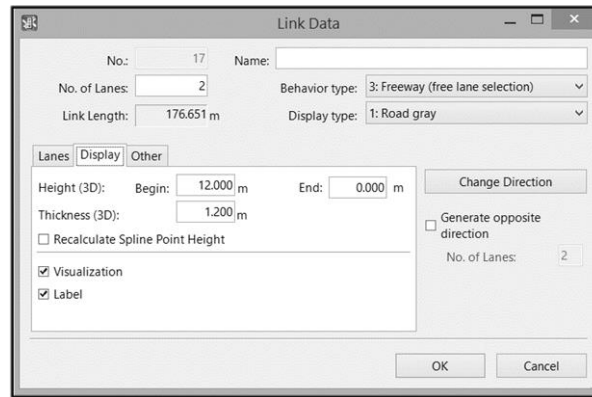


Figura A-2 – Altimetria: Janela de Propriedades

Tendo em conta a morosidade associada, este processo deve ser aplicado apenas aos arcos cuja inclinação se mostre suficientemente relevante para o escoamento do tráfego (e nos arcos vizinhos, de modo a manter a continuidade da rede).

O resultado será um modelo a três dimensões, significativamente mais realista e apelativo.



Figura A-3 – Altimetria: Visualização 3D da atribuição de altimetria

A.3.3. REPRESENTAÇÃO DA REDE

Por ora, importa referir três funcionalidades que podem facilitar tanto o traçado como o aspeto final do modelo.

A.3.3.1. MODO 3D

É possível escolher entre dois modos de visualização, *2D* e *3D*. A rede apenas pode ser alterada no modo *2D*, enquanto a adição de elementos estéticos é efetuada no modo *3D*.

A.3.3.2. MODO DE VISUALIZAÇÃO DE EIXO DOS ARCOS

Adicionalmente, é possível alternar entre o modo de visualização das faixas de rodagem ou simplesmente do eixo da via. Este último permite, por exemplo, averiguar a existência de elementos em excesso ou confirmar o seu correto posicionamento nos arcos pretendidos.

A.3.3.3. MARCAÇÕES NA VIA

As marcas de separação das vias podem ser impostas no menu **View – Options**, onde se encontra a opção **Lane Marking**. De um modo geral, obtém-se o melhor resultado aplicando uma marcação com 1 *pixel* de espessura.

Por outro lado, é possível inserir uma série de elementos, tais como indicações de direção e passadeiras através da ferramenta **Create pavement marker**, disponível na barra de ferramentas.

A.3.3.4. ESPESSURA DOS ARCOS

É possível atribuir uma espessura à visualização 3D dos arcos a partir do menu **Display** da janela de propriedades dos mesmos. Este parâmetro serve simplesmente um propósito estético, não exercendo qualquer influência no comportamento do *software*.



Figura A-4 – Representação da espessura dos arcos

A.3.4. LIMITES DE VELOCIDADE

A velocidade do tráfego é um parâmetro fundamental para o funcionamento da rede. No *Vissim*, cada tipo de veículo tem uma distribuição de velocidade associada para a generalidade da rede (explorado no capítulo A.4.1.3 - Composição). No entanto, por norma, não existe apenas um limite de velocidade ou uma única velocidade praticada em todas as vias, pelo que será necessário introduzir estas condições individualmente.

A.3.4.1. IMPOSTOS PELA SINALIZAÇÃO

A forma mais simples e intuitiva de modelar as velocidades do tráfego consiste na introdução de velocidades desejadas. Estas variáveis atuam na rede não como limitadoras da velocidade dos veículos, mas sim modelando uma distribuição de velocidade com um valor médio próximo daquele que se pretende para determinada via, o que leva à passagem de veículos acima e abaixo da velocidade limite, tal como se observaria numa situação real.

A velocidade desejada é adicionada selecionando o respetivo elemento na barra de ferramentas e pressionando o botão direito do rato em cima da via que se pretende caracterizar. Estes elementos são independentes em cada uma das vias (*lane*) que

compõem um arco (*link*), pelo que é possível atribuir-lhes velocidades distintas, aspeto particularmente útil em zonas urbanas.

Sucessivamente, adicionam-se as velocidades pretendidas consoante o tipo de veículo e repete-se o processo para cada via.

Ainda que as distribuições de velocidade possam ser definidas em **Base Data – Distributions – Desired Speed**, o *Vissim* oferece uma gama de velocidades por defeito, sendo que cada uma representa uma distribuição com um valor máximo e mínimo (entre 5 Km/h e 140 Km/h).

A.3.4.2. IMPOSTOS PELA INFRAESTRUTURA

Sabendo que o *Vissim* não incorpora o efeito da geometria do traçado na velocidade do tráfego, torna-se indispensável adicionar limitações que simulem abrandamentos naturalmente associados a curvas de raio pequeno ou ao mau estado dos pavimentos.

Tal como nas limitações impostas pela sinalização, as reduções de velocidade relacionadas com a infraestrutura devem ser inseridas para cada via de determinado arco.

Esta função executa-se arrastando o botão direito do rato em cima da via, ao longo da extensão pretendida para o abrandamento. Os veículos reduzirão a velocidade previamente à entrada na zona de velocidade limitada.

A.3.5. PRIORIDADES

Globalmente, o comportamento dos veículos numa dada interseção pode ser modelado recorrendo a **Conflict Areas**. O *Vissim* possui ainda a ferramenta **Priority Rules**, que permite seleccionar diferentes prioridades para cada tipo de veículo e que apenas deverá ser utilizada em casos particulares que as zonas de conflito não satisfaçam.

A.3.5.1. ZONAS DE CONFLITO

As zonas de conflito (**Conflict Areas**) são automaticamente definidas pelo *Vissim* e ocorrem em todas as interseções existentes no modelo. Logo, cabe ao utilizador seleccionar o tipo de comportamento esperado em cada situação específica. Esta alteração efetua-se pressionando o botão direito sobre a zona de conflito. O *Vissim* percorre as várias opções ilustrando as prioridades através de um código de cores, como se mostra na Figura A-5.

Por definição, uma dada interseção encontra-se na situação A, em que os arcos a amarelo pretendem representar a circulação de veículos que têm em conta aqueles provenientes da outra via. Apesar de resultar em paragens em ambas as vias, esta opção permite o entendimento e cooperação entre os vários elementos. A opção B nega a prioridade nas

duas vias, o que resulta em paragens e num mau funcionamento global da interseção. Não obstante, esta opção pode revelar-se útil em casos específicos, nomeadamente quando associada a sinalização luminosa.

Finalmente, com as opções C e D, é possível atribuir a prioridade a uma das vias. Esta é a situação habitualmente desejável para a generalidade das interseções, dando prioridade ao percurso representado a verde sobre o percurso representado a vermelho.

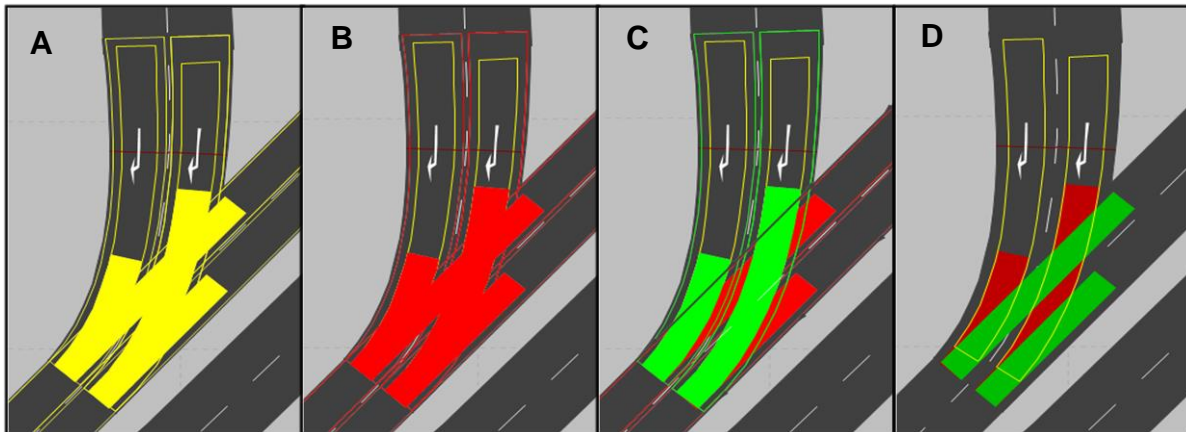


Figura A-5 – Cenários possíveis em zonas de conflito

A.3.5.2. SINAIS DE STOP

Neste programa, os sinais de STOP (**Stop Signs**) significam obrigatoriedade de paragem para os veículos, no local onde estão colocados, sem qualquer efeito na prioridade face à via a intersectar. Por conseguinte, a sua aplicação não dispensa a modelação de zonas de conflito.

Quando associados a uma distribuição de tempo, estes sinais podem igualmente ser utilizados para modelar outras paragens no percurso, designadamente em situações de portagem ou *ramp metering*.

A.3.6. SEMÁFOROS

Previamente à introdução de semáforos na infraestrutura, o *Vissim* exige a definição dos conjuntos de sinais que vão estar em funcionamento, para que seja possível modelar a duração dos períodos de sinalização verde, amarela e vermelha, de forma independente em cada interseção.

A cada interseção presente na rede corresponde um **Signal Controller (SC)**, representando o computador central responsável pela coordenação de inúmeros **Signal Groups (SG)**, ou conjuntos de semáforos sincronizados, que lhe estão associados. A Figura A-6 pretende

ilustrar a situação mais vulgar de uma interseção em que um **SC** gere dois **SG**, um por cada direção. Assim, os semáforos instalados nas vias A, B e C funcionam com os mesmos intervalos de tempo, pelo que lhes corresponde apenas um **SG** (SG2). De forma análoga, os semáforos colocados nas via D e E também se encontram sincronizados, pertencendo ao mesmo **SG** (SG1). Notavelmente, é possível utilizar o mesmo **SC** em duas interseções distintas, desde que os semáforos presentes em ambas funcionem com intervalos idênticos.

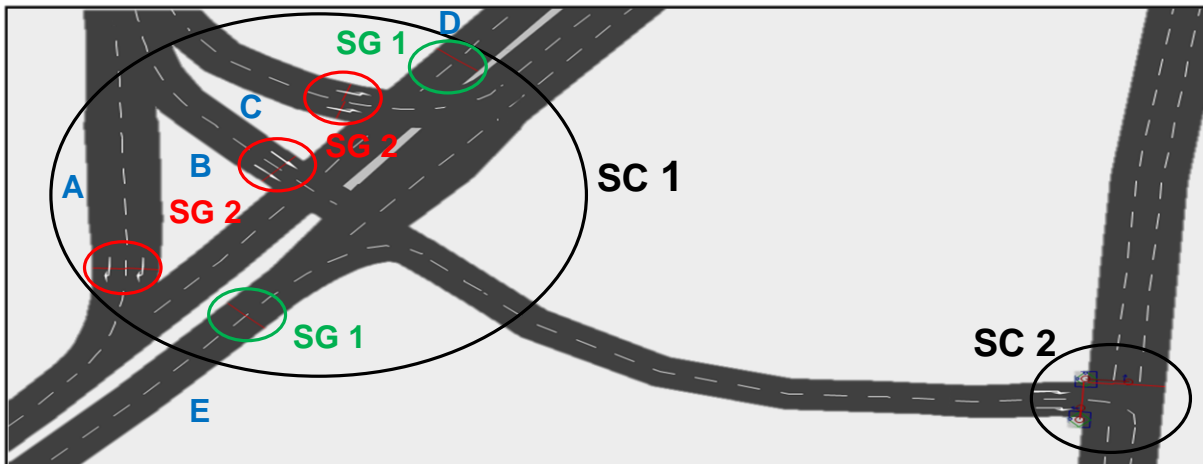


Figura A-6 – Esquematização de Grupos de Semáforos

Primeiramente, no menu **Signal Groups**, adicionam-se tantos grupos quanto necessário para a interseção em estudo (geralmente, dois).

De seguida, no separador **Signal Programming**, calibram-se os tempos pretendidos em cada **Signal Group**, assim como a forma de interação que caracterizará cada grupo. As alterações são efetuadas arrastando as barras coloridas para a extensão desejada, em função da escala de tempo representada no topo da janela.

Como regras gerais, os tempos de verde não deverão cruzar-se, e é essencial que o tempo de vermelho de um SG se prolongue até ao início do tempo de vermelho do SG seguinte.

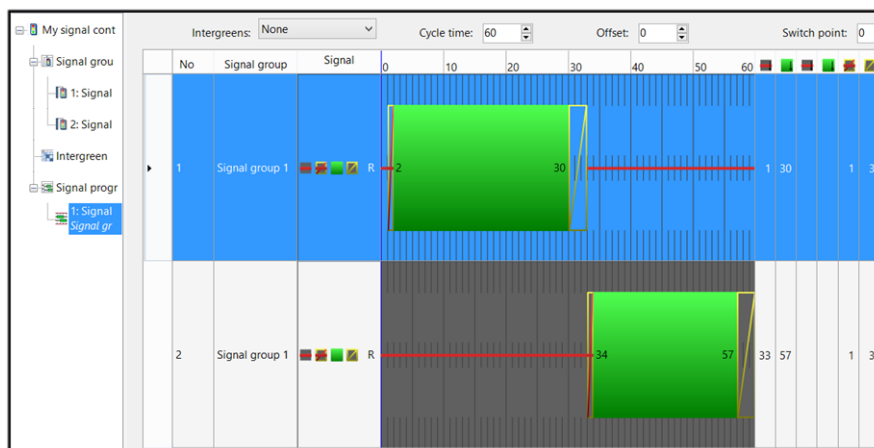


Figura A-7 – Temporização dos Sinais

Por fim, procede-se à instalação dos semáforos na rede pressionando o botão direito do rato sobre as vias pretendidas. Em cada clique, é aberta uma janela na qual se introduz o SC e SG a que o semáforo pertence e, caso necessário, os veículos aos quais o sinal se aplica.

Tal como para grande parte dos elementos do *Vissim*, é necessário repetir a introdução dos sinais em cada via (*lane*) do mesmo arco (*link*). Quando se pretende colocar elementos com as mesmas características em mais do que uma via, é possível copiar o primeiro elemento criado pressionando a tecla **CTRL** e arrastando-o para a via seguinte.

Finalmente, os elementos de sinalização podem ser adicionados no modo 3D, valorizando eventuais apresentações a ser desenvolvidas. A colocação de objetos 3D será posteriormente explorada na secção A.3.7. Contudo, importa aqui referir que a coordenação entre os tempos das luzes da sinalização em modo tridimensional e os movimentos de arranque e paragem do tráfego exige, aquando da adição dos objetos, que se defina o **SC** e **SG** a que cada cabeça de sinal se refere.

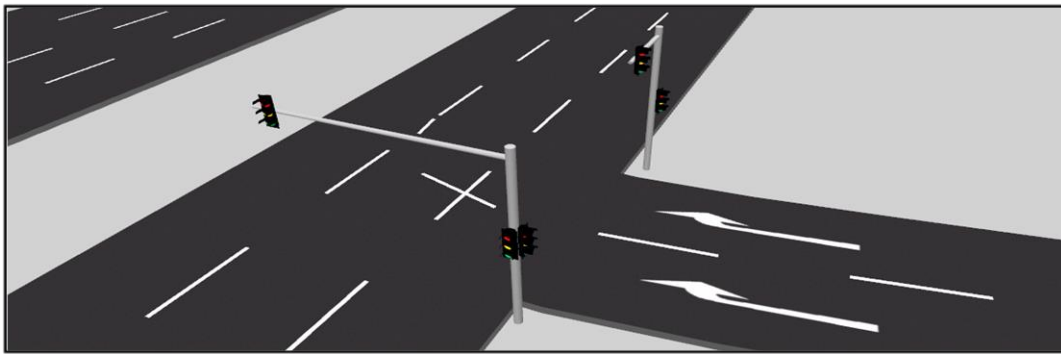


Figura A-8 – Representação 3D dos Semáforos

A.3.7. OBJETOS 3D

Independentemente do local onde sejam colocados, os objetos tridimensionais não têm influência sobre o comportamento do tráfego, servindo somente uma função estética, designadamente para fins de apresentação.

A introdução de objetos é feita no modo 3D, pressionando o botão direito do rato no local pretendido. O programa permite selecionar o objeto a introduzir, a cota da sua base e ainda a orientação do modelo.

À exceção do que concerne à posição, não é exequível editar objetos no *Vissim*, pelo que é indispensável utilizar outro *software* de modo a efetuar as alterações pretendidas. Nestes casos, não é imperioso remover o objeto anteriormente inserido, bastando substituir o modelo antigo pelo novo, mantendo o mesmo nome e diretório.

O *Vissim* inclui originalmente uma gama de objetos à disposição do utilizador. Embora se encontrem em formato 3D, pouco corrente e dificultando a edição dos modelos originais, é possível importar modelos criados em *Google SketchUp*, um programa de utilização grátis que detém uma larga base de dados de objetos criados por outros utilizadores. Este facto torna bastante prática a aplicação e personalização de modelos como praças de portagem, valorizando a apresentação que resulta da simulação efetuada.



Figura A-9 – Árvores no modo 3D

A.4. TRÁFEGO

A.4.1. VEÍCULOS

No que diz respeito ao tráfego que irá percorrer a rede, o elemento fundamental é a categorização em tipos de veículos (**Vehicle Types**). Esta variável define veículos ligeiros, pesados, BUS e quaisquer variações desses elementos que se pretendam introduzir na rede. A título de exemplo, no estudo de uma praça de portagem, seria necessário criar um tipo de veículo ligeiro para cada modo de pagamento e, seguidamente, repetir o processo para os veículos pesados.

Por sua vez, os diversos tipos de veículos agrupam-se em classes (**Vehicle Classes**), que permitem, nomeadamente, agregar todos os tipos de veículos ligeiros num único grupo, possibilitando a imposição de restrições na rede com maior eficiência. De facto, qualquer restrição em vias ou sinais é criada em função das classes de veículos, pelo que será sempre necessário defini-las, ainda que contenham apenas um tipo de veículo.

Finalmente, definem-se as composições do tráfego (**Vehicle Compositions**) que representam a base fundamental para o *input* de fluxos na rede, uma vez que definem a percentagem de cada tipo de veículo que entra num determinado arco (**link**).

Esquemáticamente, estes três campos organizam-se da seguinte forma:

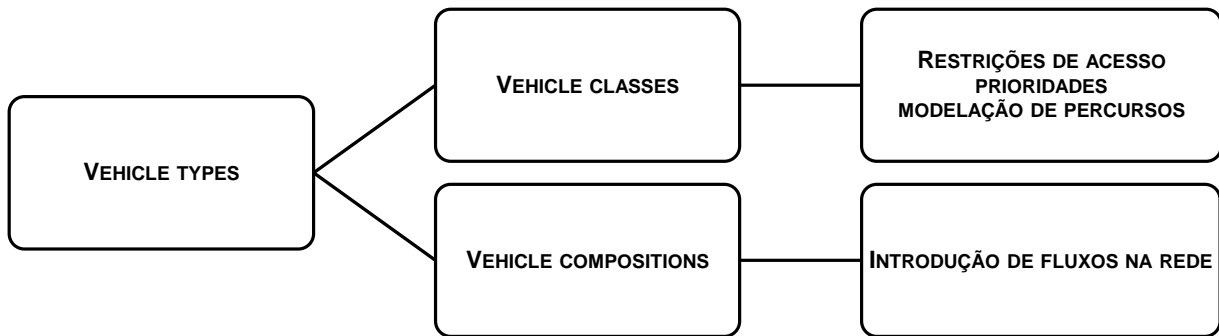


Figura A-10 – Diagrama de Vehicle Types no Vissim.

A.4.1.1. TIPOS DE VEÍCULOS

O Vissim categoriza automaticamente os veículos em *Car* (ligeiros), *HGV* (pesados), *Bus* (autocarros), *Tram* (metro ligeiro), *Pedestrian* (peões) e *Bike* (velocípedes). Outros tipos podem ser definidos no menu **Base Data – Vehicle Types**, criando novas entradas na lista ou duplicando um tipo já existente.

Adicionalmente, é possível definir a ocupação de cada tipo de veículo (ferramenta útil para o estudo de vias de alta ocupação), a distribuição de cores (que permite identificar os tipos de veículos no decorrer da simulação) e ainda a distribuição de modelos 3D a utilizar (função que influencia somente o efeito estético da apresentação).

A.4.1.2. CLASSES DE VEÍCULOS

Como já referido, estas classes reúnem grupos de veículos com características em comum. Na coluna **VehTypes**, selecionam-se os tipos pretendidos para cada classe, que podem ser observados no quadro à direita, tal como mostra a Figura A-11. É também possível escolher uma distribuição de cores que vá ao encontro da opção tomada para os tipos de veículos.

A Figura A-11 demonstra ainda dois tipos de veículos ligeiros que diferem na ocupação, porém partilham o modo de pagamento de portagem (VC), pelo que são agrupados na classe *Car VC*, facilitando futuras referências a ambos os tipos de veículo.

Count	No	Name	VehTypes	UseVehTypeColor	Color
2	20	HGV	200	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
3	30	Bus	300	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
4	40	Tram	400	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
5	50	Pedestrian	510,520	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
6	60	Bike	600	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
7	70	Bus	300	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
8	80	Car VV	620,690	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
9	110	Car VC	630,700	<input checked="" type="checkbox"/>	(255, 0, 0, 0)
10	120	Car VM	100,640,710	<input checked="" type="checkbox"/>	(255, 0, 0, 0)

Count	No	Name	Category	Model2D3DDistr	ColorDistr1	OccupDistr	Capacity
1	630	Car VC 1	Car	10: Car	3: Via Car	1: 1 Ocupan	0
2	700	Car VC 2	Car	10: Car	206: Via Car	2: 2 Ocupan	0

Figura A-11 – Classes de Veículos

A.4.1.3. COMPOSIÇÃO DO TRÁFEGO (VEHICLE COMPOSITIONS)

Neste menu, podem ser definidas as percentagens de cada tipo de veículo com origem em determinado local. Na metade esquerda da Figura A-12, realçam-se quatro composições que correspondem a locais de origem de fluxos distintos. À direita, adicionam-se os tipos de veículos necessários e atribui-se o fluxo relativo desejado para cada um (**RelFlow**). É de notar que estes valores podem ser inseridos como montantes exatos e não, obrigatoriamente, como percentagem. De seguida, o *Vissim* calcula o valor adequado em função da soma dos dígitos introduzidos pelo utilizador.

Vehicle Compositions / Relative Flows						
Count: 5	No	Name	Count: 3	VehType	DesSpeedDistr	RelFlow
1	1	Default	1	100: Car	90: 90 km/h	90.900
2	2	NS	2	200: HGV	80: 80 km/h	8.400
3	3	CL	3	300: Bus	80: 80 km/h	0.700
4	6	AI				
5	8	SN				

Figura A-12 – Composições de Veículos

A.4.2. INTERVALOS DE TEMPO

Previamente à introdução dos fluxos na rede, poderá ser necessário definir períodos de tempo com diferentes características, nomeadamente nos casos em que se pretenda simular a evolução do estado da rede ao longo de diferentes intervalos da hora de ponta.

Para tal, a partir do menu **Base Data – Time Intervals**, adicionam-se as entradas necessárias e preenchem-se os períodos de início e fim de cada intervalo, em segundos.

Time Intervals List			Area behavior types			
Count: 3	Start	End				
1	0.0	900.0				
2	900.0	1800.0				
3	1800.0	MAX				
	Add...					
	Delete					

Figura A-13 – Intervalos de Tempo

A.4.3. INTRODUÇÃO DE FLUXOS NA REDE

Após os passos supracitados, será possível introduzir os fluxos na infraestrutura, pressionado o botão direito do rato sobre o arco (**link**) de origem. O **Vehicle Input** corresponde sempre a todas as vias do arco e fica representado no seu início de forma constante, independentemente do local onde se coloque o rato.

De seguida, à lista resultante da introdução do elemento de *input*, associa-se a composição de veículos (*Vehicle Compositions*) desejada, assim como o volume de tráfego pretendido para cada intervalo de tempo.

Count	No	Name	Link	Volume(0)	VehComp(0)
1	2	AL	64	1010.0	6: AI
2	3	NS	5: TAB SN 2	3100.0	2: NS
3	4	CL	60	2902.0	3: CL
4	5	SN	73	3750.0	8: SN

Figura A-14 – Volumes de Tráfego

A.4.4. MODELAÇÃO DE PERCURSOS

Para representar convenientemente os fluxos no interior da rede, é igualmente recomendável modelar as mudanças de direção efetuadas pelos veículos, pressionando o botão direito do rato sobre o arco de origem dos fluxos e, de seguida, sobre os arcos de destino. Posteriormente, define-se a percentagem do tráfego que segue para cada um dos destinos escolhidos.

É útil especificar as classes de veículos afetadas pela *route* inserida, particularmente em casos mais complexos. De modo semelhante ao que sucede com os *Vehicle Inputs*, o fluxo relativo associado a cada destino não tem de ser introduzido em percentagem, uma vez que o *Vissim* executa o cálculo com base no somatório dos valores introduzidos.

É fundamental a introdução de *Vehicle Routes* em todas as interseções que contenham mais do que uma hipótese de destino. Adicionalmente, as *routes* são de enorme utilidade para modelar casos complexos como parques de estacionamento e praças de portagem. Na Figura A-15, à esquerda, observa-se um parque de estacionamento onde é necessário introduzir uma *route* que capte a totalidade do tráfego (fora da influência de outra *route*) e o conduza para a saída, evitando que veículos entrem num *loop* em torno do parque. Por outro lado, à direita, observa-se uma praça de portagem onde se torna indispensável encaminhar todo o tráfego com um determinado modo de pagamento para as cabines correspondentes.

Este processo pode tornar-se bastante moroso em estradas com um grande número de vias e diversos tipos de veículos exigindo, naturalmente, uma comparação constante com a situação real.

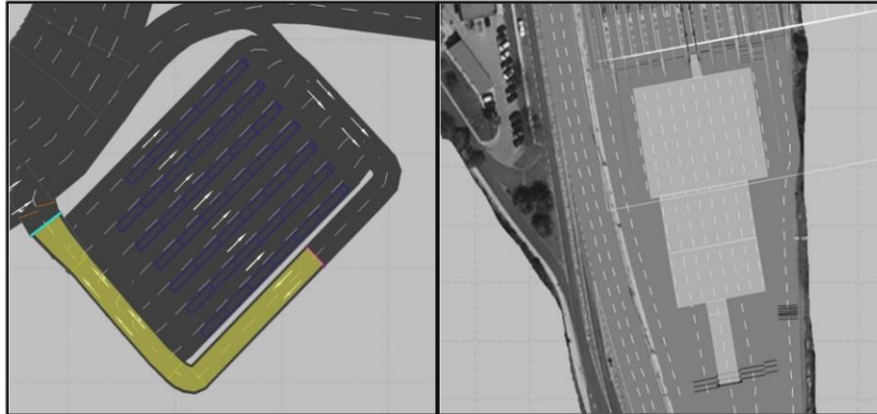


Figura A-15 - Parque de Estacionamento e Praça de Portagem

A.5. SIMULAÇÃO

A.5.1. DEFINIÇÃO DOS PARÂMETROS PARA A SIMULAÇÃO

Previamente ao início da simulação, é essencial rever os parâmetros utilizados pelo *Vissim* para computação dos resultados:

Period – Corresponde ao tempo de simulação, em segundos. Por definição, o programa estabelece um período de 3600 segundos (1 hora). Todavia, em grande parte das situações, nomeadamente quando se definem volumes distintos para diferentes intervalos de tempo, é essencial recorrer a um período de simulação superior. Adicionalmente, redes com uma dimensão considerável poderão demorar mais de 1 hora a atingir convergência, isto é, um ponto em que os níveis de saturação não se alteram para o mesmo volume de tráfego.

Start Time e Start Date – Definem a hora e data de início da simulação. Ao corresponder a uma hora e data reais, facilitam o mapeamento posterior dos resultados.

Simulation Resolution – Representa a resolução dos passos de simulação. Enquanto valores diminutos poderão conduzir à perda de eventos relevantes para a evolução do estado da rede, valores muito elevados consumirão excessivamente os recursos de processamento, diminuindo a velocidade da simulação. A *PTV* recomenda um valor de 10 *time steps* por segundo de simulação.

Random Seed – É uma constante introduzida nos algoritmos de introdução de veículos na rede, com o intuito de permitir a obtenção de resultados distintos em diversas simulações. Desta forma, ao alterar a *semente* (**Seed**) e calcular a média de valores finais em cada simulação, obter-se-ão resultados mais fiéis à realidade.

Nr. of Runs e Random Seed Increment – Estes parâmetros permitem programar o número de simulações desejadas e incrementar a *semente* a cada simulação, evitando assim a obrigatoriedade de alterar os valores e recomeçar a simulação manualmente.

Simulation Speed – É possível estabelecer uma velocidade máxima para a simulação. No entanto, na maioria das situações, esta variável será limitada pela velocidade do *hardware* do computador, tornando-a uma opção pouco relevante. A velocidade da simulação pode também ser modificada no decorrer da mesma.

Break at – Este parâmetro interrompe a simulação ao termo de um determinado período de tempo (em segundos), definido pelo utilizador.

A.5.2. APRESENTAÇÃO

Para que o *Vissim* grave um vídeo da simulação é necessário que a visualização da rede esteja em modo 3D e que a opção **Record AVIs** esteja escolhida no menu **Presentation**.

O modo mais prático para construir apresentações é criando uma **Storyboard**. Esta consiste numa série de posições ou pontos de vista que o programa vai percorrendo em intervalos de tempo definidos pelo utilizador, as **Camera Positions**.

As posições são gravadas partindo da vista desejada em modo 3D, e preenchendo o campo apontado na Figura A-16 com o nome desejado.

De seguida, no menu **Presentations – Storyboards**, adicionam-se as posições pretendidas para a apresentação e preenche-se os valores pretendidos para o tempo de início (**StartTime**) e o tempo que o programa se mantém em cada posição (**DwellTime**).

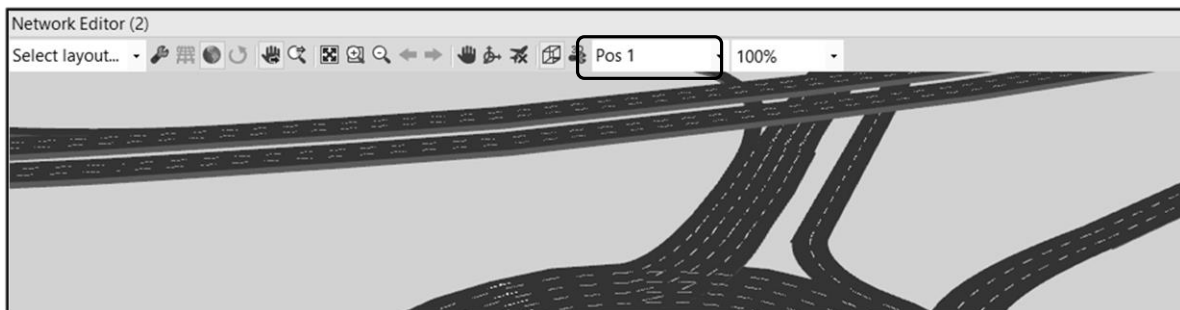


Figura A-16 – Criação de Camera Positions

O *Vissim* guarda o vídeo na pasta onde está gravado o ficheiro **inpx**, correspondente ao ficheiro que contém o modelo.

As gravações efetuadas sem um codificador/compressor poderão tornar-se demasiadamente grandes, podendo inclusivamente diminuir a velocidade da gravação, pelo que é de grande utilidade a utilização de um codificador externo como, por exemplo, o **Xvid MPEG-4 Codec**. Esta opção pode ser definida no menu **Edit – User Preferences – GUI – Graphics – AVI Recording – Compression**.

A.6. OBTENÇÃO DE RESULTADOS

Finalmente, está-se em condições de avaliar os resultados da simulação. Neste manual, apresenta-se o modo de obtenção de leituras de tempos de percurso e filas de espera, uma vez que necessitam de ser definidos na rede, mas o *Vissim* fornece automaticamente outros dados que podem ser consultados no menu **Evaluation**. Os resultados são apresentados em listas que poderão ser exportadas para uma base de dados em *Access* ou simplesmente para um bloco de notas. É ainda possível sublinhar os dados pretendidos e copia-los diretamente para o *Excel*.

A.6.1. TEMPO DE PERCURSO

Os tempos de percurso são medidos mediante a colocação de um ponto de início e fim para a medição. Para tal, pressiona-se o botão direito do rato em cima do ponto de início e repete-se para o ponto de fim.

Os resultados são apresentados no menu **Evaluation – Result Lists – Vehicle Travel Times**. Apresentam a quantidade de veículos de cada tipo que passaram nas secções em estudo (**Vehs**) e o tempo, em segundos, da viagem de cada tipo (**TravTm**).

A.6.2. FILAS DE ESPERA

As filas de espera são obtidas colocando um *counter* nos locais pretendidos. As medições são feitas para montante do elemento colocado na via e correspondem às filas que se desenvolvem continuamente nesse sentido. Os resultados podem ser consultados no menu **Evaluation – Result Lists – Queue Lengths**. É apresentado o comprimento da fila para o *timestep* atual, o comprimento máximo obtido ao longo da simulação e ainda o número total de paragens efetuadas pelos veículos.

A.7. ESTUDO DE CASO – VIA DE ALTA OCUPAÇÃO NO ACESSO À PRAÇA DE PORTAGENS DA PONTE 25 DE ABRIL

A.7.1. ENQUADRAMENTO

Com o intuito de reduzir o número de veículos em circulação e, conseqüentemente, os níveis de congestionamento e poluição atmosférica na Área Metropolitana de Lisboa, o Instituto de Infraestruturas Rodoviárias, atualmente Instituto da Mobilidade e dos Transportes, efetuou um estudo para analisar a viabilidade de implementação de Vias de

Alta Ocupação⁵ (VAO) nas infraestruturas rodoviárias de acesso à cidade, enquadrado no Plano de Melhoria da Qualidade do Ar na Região de Lisboa e Vale do Tejo.

O estudo envolveu a análise de quatro corredores: Alverca - Lisboa (A1), Fogueteiro - Lisboa (A2), Cascais - Lisboa (A5) e Sintra – Lisboa (IC19). A comparação das vantagens e desvantagens de cada corredor culminou na seleção do corredor Fogueteiro – Lisboa para o desenvolvimento de um projeto-piloto para a implementação de uma VAO.

A.7.2. OBJETIVOS

O objetivo desta secção é a demonstração das funcionalidades de um programa comercial de simulação microscópica de tráfego aplicadas a um caso complexo e real.

Este exemplo não representa um estudo exaustivo e não tem o objetivo de representar um modelo fiel da situação atual, mas sim uma plataforma de demonstração da modelação de uma infraestrutura complexa como uma praça de portagens. Analogamente, as alternativas apresentadas são apenas cenários possíveis para efeitos de demonstração e não representam soluções ideais para a implementação de uma VAO.



Figura A- 17 – Modelo da praça de portagens da Ponte 25 de Abril

A.7.3. RECOLHA DE DADOS

A.7.3.1. MODOS DE PAGAMENTO

Os dados recolhidos revelam uma distribuição de cerca de 49% de utilizadores de Via Verde e 25% de utilizadores de Via Card, com os restantes 26% correspondendo a pagamento manual.

⁵ Uma Via de Alta Ocupação (*High Occupancy Vehicle lane, HOV*) consiste numa via dedicada a veículos com mais do que um ocupante e, eventualmente, dando prioridade a autocarros, táxis com passageiros, veículos prioritários ou ainda veículos elétricos.

Tabela A-2 – Percentagem de utilizadores por modo de pagamento (Fonte: InIR, IP)

MODO	%	VEÍCULOS
VIA VERDE	0.49	3159
VIA CARD	0.25	1612
VIA MANUAL	0.26	1676

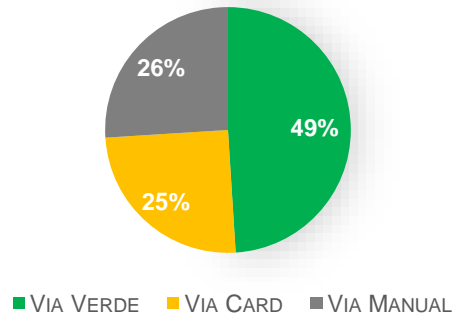


Figura A-18 - Percentagem de utilizadores por modo de pagamento

A.7.3.2. TAXAS DE OCUPAÇÃO

Relativamente à taxa de ocupação, 70% dos veículos viaja com apenas um passageiro e 25% com dois passageiros, pelo que a VAO terá maior interesse para esta gama.

Tabela A-3 – Percentagem de utilizadores por taxa de ocupação (Fonte: InIR, IP)

OCUPAÇÃO	1	2	2+
%	0.70	0.25	0.05
VEÍCULOS	4578	1612	322

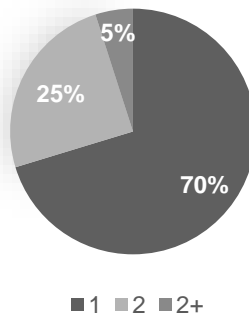


Figura A-19 – Percentagem de utilizadores por taxa de ocupação

A.7.3.3. FLUXOS NA REDE

Os dados relativos aos fluxos na rede foram obtidos com base em contagens efetuadas para o período das 7h-11h. Uma vez que, por uma questão de simplicidade, não irá efetuar uma afetação dinâmica do tráfego, considerou-se o fluxo médio horário do período da manhã para os pares OD da Tabela A-4.

Tabela A-4 – Fluxos de entrada na rede. (Fonte: InIR, IP)

VEIC./HORA	CAPARICA - LISBOA	FOGUETEIRO - LISBOA	SETÚBAL - ALMADA	ALMADA - LISBOA	SETÚBAL - CAPARICA
LIGEIOS	1451	3007	628	1010	116
PESADOS	30	88	16	21	16
Bus	5	16	10	36	0

Em função do modo de pagamento e taxa de ocupação apresentados anteriormente, é necessário a separação dos tipos de veículos a ser modelados (ligeiros, pesados e BUS) em classes que permitam a sua afetação quanto aos pórticos de portagem que podem utilizar e quanto à sua elegibilidade para utilização da VAO.

O resultado é o aparecimento de dez classes de veículos, cujos fluxos serão introduzidos no *Vissim* de acordo com a Tabela A-5.

Importa notar que, neste modelo, apenas os veículos ligeiros são elegíveis para utilização da VAO, pelo apenas estes são separados de acordo com a taxa de ocupação. Por outro lado, por uma questão de simplicidade, considera-se que todos os veículos da classe BUS utilizam o modo de pagamento Via Verde.

Tabela A-5 – Fluxos de entrada na rede por tipo de veículo, modo de pagamento e taxa de ocupação

	CL	FL	SA	AL	SC	TOTAL
CAR VV	505	1046	218	351	40	2161
CAR VC	257	534	111	179	21	1102
CAR VM	268	555	116	186	21	1147
CAR VV 2+	206	427	89	144	16	883
CAR VC 2+	105	218	45	73	8	450
CAR VM 2+	109	227	47	76	9	468
HGV VV	15	43	8	10	8	83
HGV VC	8	22	4	5	4	43
HGV VM	8	23	4	5	4	44
BUS	5	16	10	36	0	67
TOTAL	1485	3111	653	1067	132	6448

A.7.4. MODELAÇÃO DA REDE

A modelação de uma praça de portagens não é uma funcionalidade para a qual o *Vissim* esteja preparado de origem, pelo que é necessário adaptar as ferramentas existentes para a construção de um modelo que simule um comportamento semelhante ao pretendido.

O alargamento do número de vias na chegada ao pórtico é modelado colocando arcos sucessivos com um número de vias crescente. O incremento é de duas vias em cada arco,

de modo que se mantenha o alinhamento das vias existentes. A ligação entre um novo arco e o anterior é feito com conectores entre as vias das extremidades.

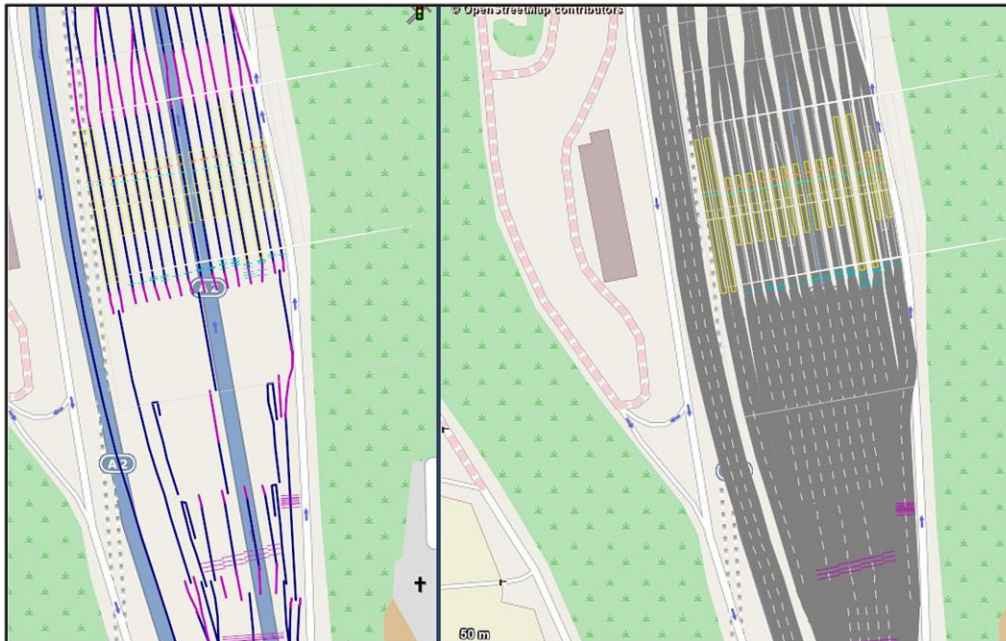


Figura A-20 – Alargamento do número de vias

A modelação de traços contínuos é feita colocando arcos paralelos, sob pena de perder o efeito de mudanças de via ilegais.

Na Figura A-20 – Alargamento do número de vias, à esquerda, é possível verificar a rede em modo de estrutura, onde apenas são representados os eixos dos arcos (em azul) e dos conectores (em cor-de-rosa). À direita, está representado o efeito final obtido.

O alargamento do número de vias resulta em diversos espaços brancos na rede, o que é contrariado colocando arcos (visíveis na Figura A-20 – Alargamento do número de vias pelo seu formato de gancho) sem ligação à restante infraestrutura.

Os pórticos de portagem são modelados utilizando zonas de redução de velocidade para 10 km/h nos pórticos que exigem a paragem dos veículos, acompanhadas de um sinal de *STOP* com um temporizador que obriga a paragem dos condutores por um período médio de 7 segundos, com base em Saffarzadeh (2006) [52].

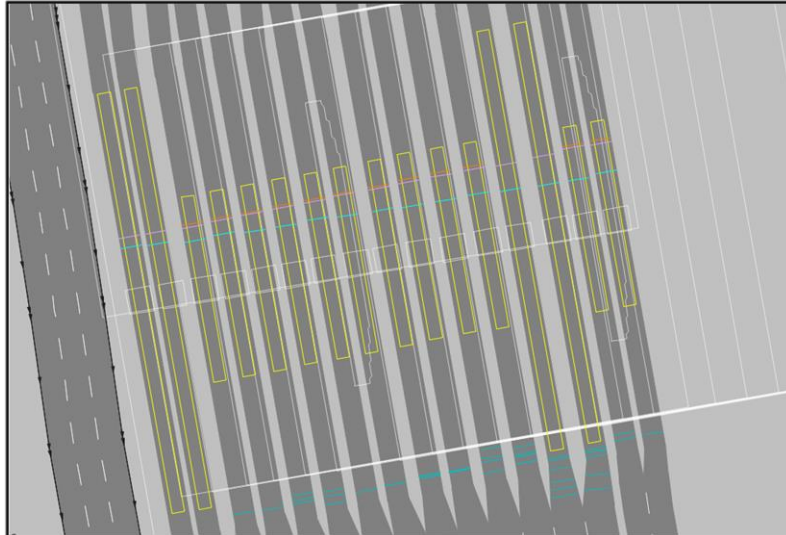


Figura A-21 – Modelação dos pódios de portagem

Após a introdução dos elementos fundamentais da infraestrutura, procede-se à afetação do tráfego que passa em cada via para os pódios possíveis de serem utilizados em função de cada modo de pagamento. Neste processo, o fator mais importante é a sensibilidade do modelador na atribuição das probabilidades de escolha de cada destino em cada via. Isto é, ainda que a rede o permita, um veículo que circule na via mais à esquerda não deverá escolher um dos pódios mais à direita e vice-versa.

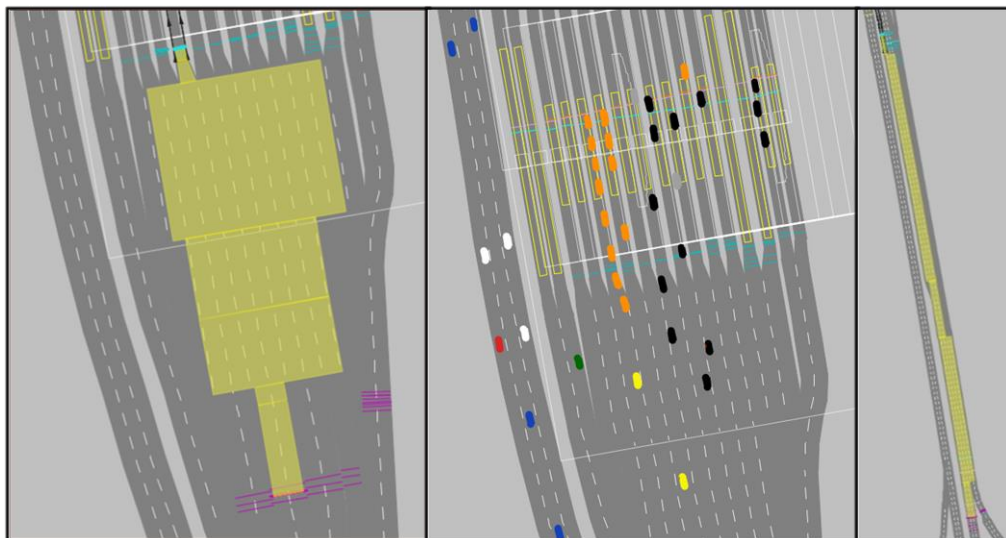


Figura A-22 – Afetação do tráfego aos pódios de portagem

Desta forma são também modelados os entrecruzamentos que se verificam ao longo do acesso à praça de portagens como, por exemplo, o cruzamento do tráfego proveniente de

Almada para a via mais á esquerda, com o objetivo de se posicionar para o acesso à Via Verde.

Finalmente, é possível a introdução de elementos estéticos como um modelo tridimensional da praça de portagem, tornando possível determinar o modo de pagamento aplicável a cada pórtico e providenciando um aspeto mais realista à simulação.



Figura A-23 – Representação 3D da praça de portagem

Neste estudo, pretendem-se apenas demonstrar as funcionalidades do *Vissim*, pelo que não é feita a calibração do modelo. No entanto, este seria o momento oportuno para testar o comportamento dos veículos na rede face à situação real e ajustar, não só os parâmetros de calibração referentes aos modelos de comportamento, mas também os percursos modelados para a distribuição dos veículos pelos pórticos.

A.7.5. SITUAÇÃO ATUAL

Com a rede modelada e apresentando um comportamento satisfatório, é possível proceder-se à simulação da situação atual.

O modelo é corrido durante 4600 segundos (1h e 17min), tempo a partir do qual, neste caso, não se registaram mais alterações no desempenho da rede.

De modo a avaliar o desempenho da rede, foram medidos os tempos de percurso e comprimento entre os pares OD representados na Figura A-24 e o comprimento das filas de espera em cada pórtico.

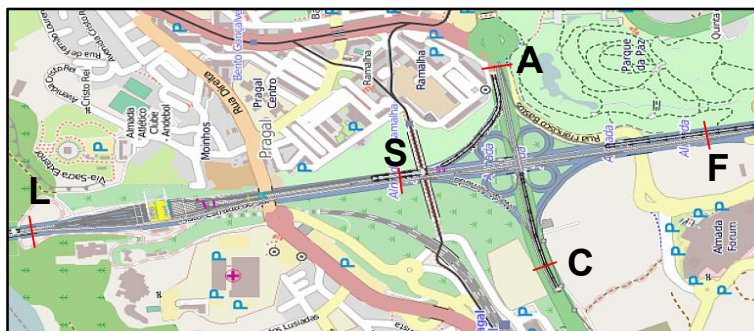


Figura A-24 – Representação dos pares OD

Obtiveram-se os valores representados na Tabela A-6, que demonstram um maior tempo de percurso para o tráfego proveniente da Costa da Caparica, devido às complicações sentidas no nó de Almada.

Tabela A-6 – Tempos de percurso na situação atual

PERCURSO	TEMPO DE PERCURSO (MIN)
CL	14.36
FL	9.74
AL	7.57
SL	5.72

Relativamente aos pórticos de portagem, verificam-se filas de espera que rondam os 390m em todos os pórticos que obrigam à paragem dos veículos, com exceção dos pórticos 15 e 16, posicionados na extremidade direita, uma vez que durante o período de ponta de manhã são reservados a veículos pesados. Como esperado, não se registam filas de espera nos pórticos referentes ao modo de pagamento Via Verde, pois não obrigam à paragem dos veículos.

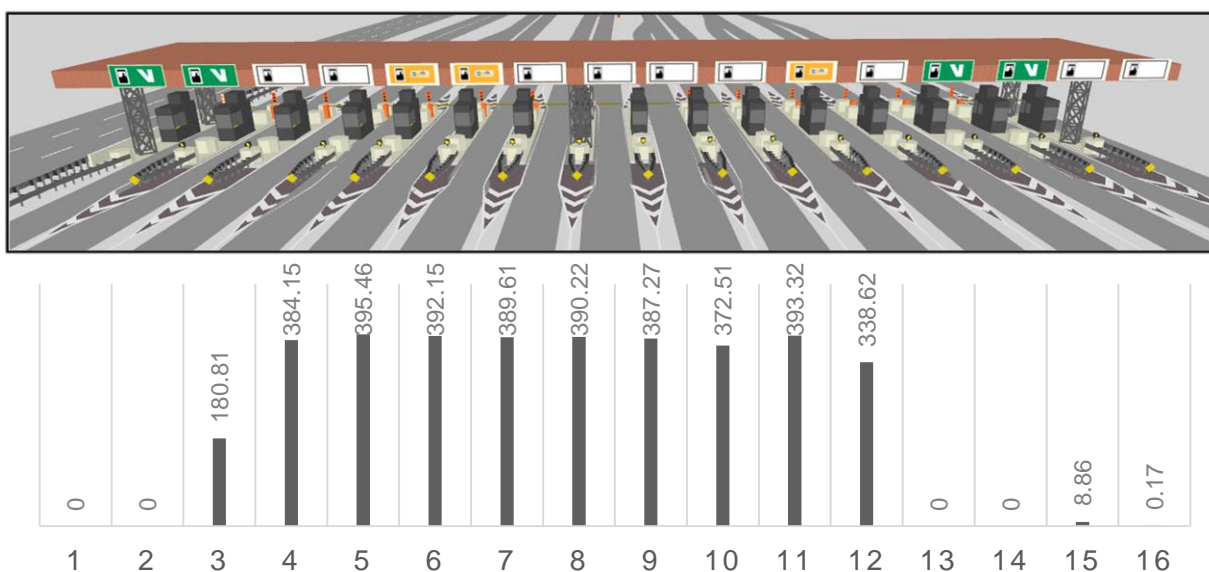


Figura A-25 – Comprimentos das filas de espera na situação atual

Com a situação atual modelada, torna-se possível a implementação de alterações na rede e a avaliação do seu efeito no desempenho global do modelo.

Nesta fase, seria oportuno comparar os indicadores de desempenho obtidos com os valores medidos no terreno e, em caso de discordância, investigar e corrigir os fatores causadores da discordância. No entanto, como referido anteriormente, este estudo não prevê a realização de trabalho de campo, pelo que o processo de calibração do modelo é ignorado.

A.7.6. IMPLANTAÇÃO DA VAO

Para a implementação da Via de Alta Ocupação, optou-se pela terceira via da A2, uma vez que não é diretamente afetada pela convergência no nó de Almada, nem é afetada pela generalidade das mudanças de via que ocorrem no acesso às portagens.

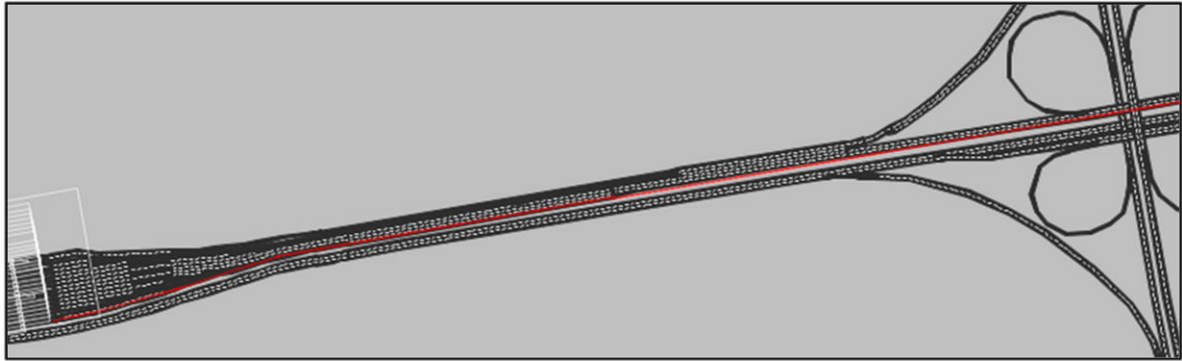


Figura A-26 – Planta da implantação da VAO

Numa primeira iteração, considerou-se oportuna a atribuição de três pórticos com acesso exclusivo ao tráfego proveniente da VAO, um para cada modalidade de pagamento, com o objetivo de a tornar atraente para o maior número de utilizadores possível.

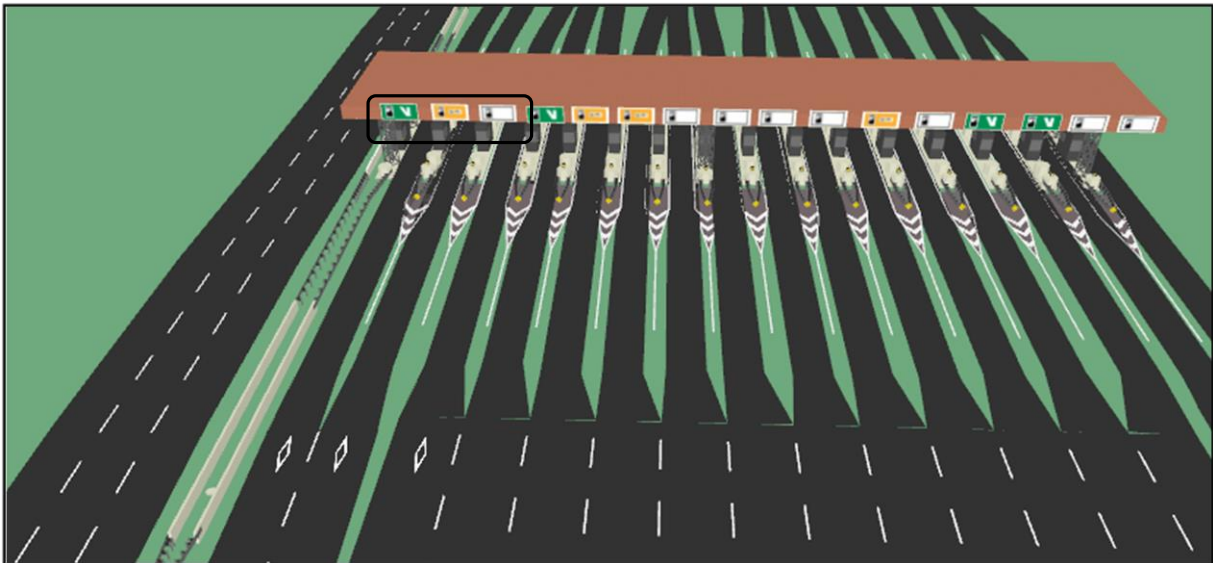


Figura A-27 – Proposta inicial de disposição dos pórticos de portagem

No entanto, iniciando a simulação, verificou-se imediatamente o congestionamento da VAO por parte dos veículos que utilizam os modos de pagamento que obrigam à paragem no pórtico, como representado na Figura A-28.

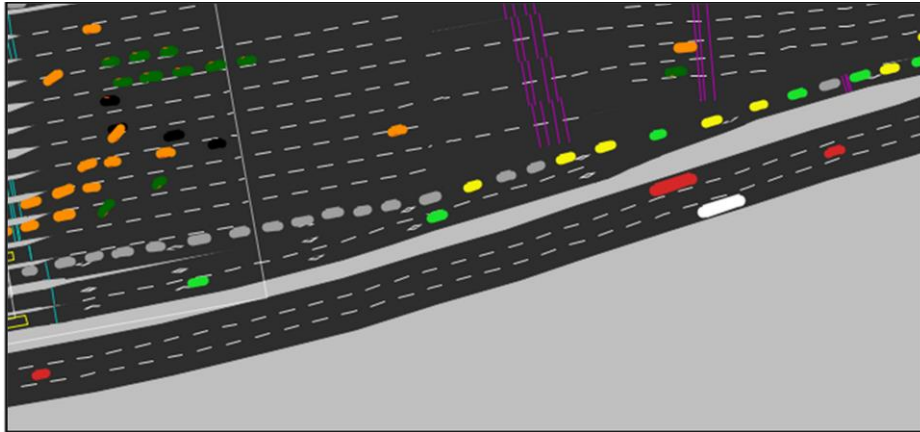


Figura A-28 – Congestionamento na VAO resultado da primeira disposição de pórticos

De modo a contrariar este efeito, foi necessário reorganizar a disposição dos pórticos de portagem, suprimindo um pórtico de pagamento manual, e acrescentando um pórtico com pagamento por Via Verde.

Neste novo cenário, considera-se que a VAO é apenas atraente a quem utiliza Via Verde, o que limita significativamente a ocorrência de filas de espera, uma vez que este modo de pagamento não obriga à paragem dos veículos.

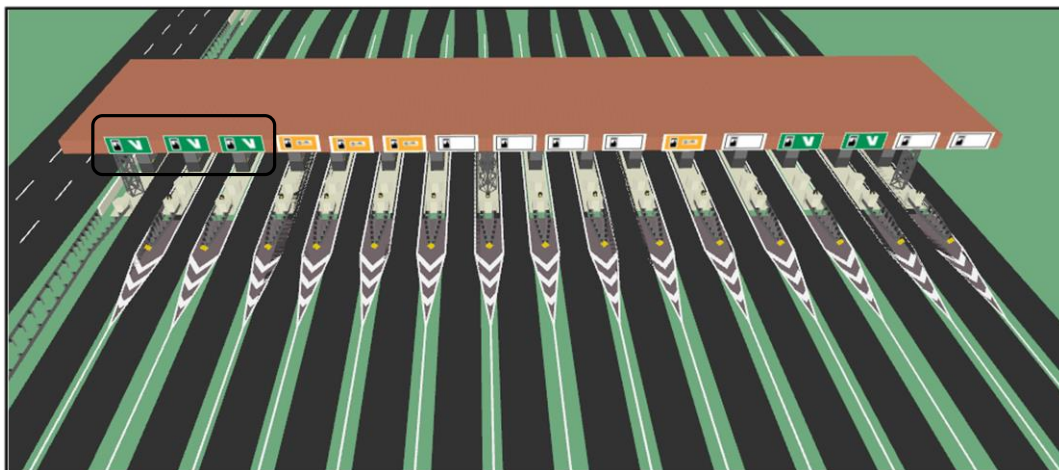


Figura A-29 – Proposta final de disposição de pórticos de portagem

A.7.7. SIMULAÇÃO DE ALTERNATIVAS

A.7.7.1. ALTERNATIVA 1

Com a nova infraestrutura, correu-se a simulação mantendo a mesma distribuição de taxas de ocupação dos veículos da situação atual e verificou-se uma descida global nos indicadores de desempenho da rede.

Os tempos de viagem aumentaram severamente para o tráfego proveniente da Costa da Caparica e Fogueteiro, resultante do desvio do tráfego para a primeira e segunda via da A2.

Tabela A-7 – Tempos de percurso na alternativa 1

PERCURSO	ATUAL	ALTERNATIVA 1
CL	14.36	22.55
FL	9.74	16.50
AL	7.57	6.49
SL	5.72	7.89

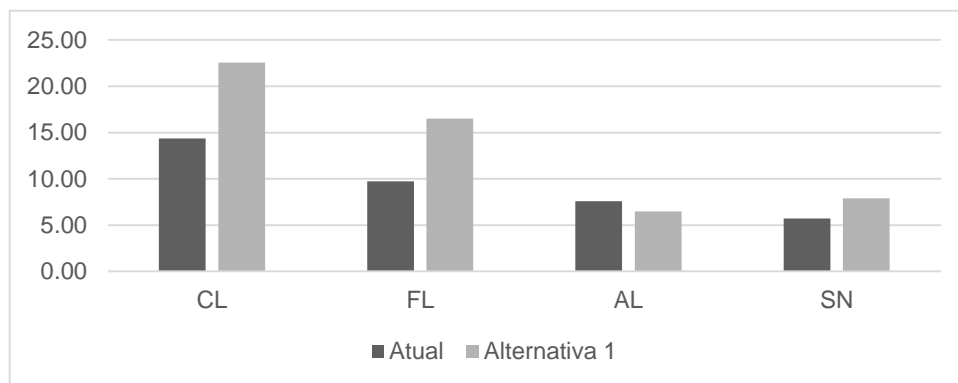


Figura A-30 – Tempos de percurso na alternativa 1

Também nos pórticos se verifica um aumento das filas de espera, uma vez que uma maior percentagem do tráfego é agora canalizado para a zona central.

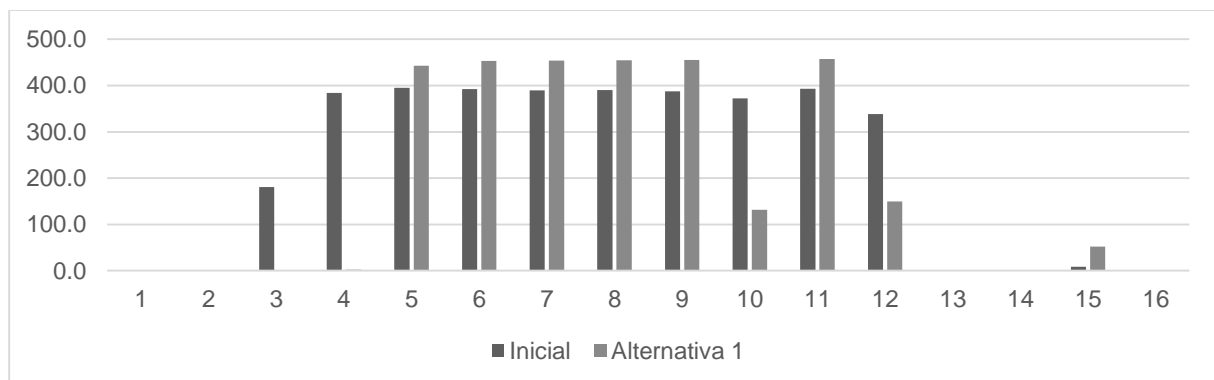


Figura A-31 – Comprimento das filas de espera na alternativa 1

A.7.7.2. ALTERNATIVA 2

Uma vez que o objetivo da VAO é promover a redução do número de veículos na rede, atraindo a partilha do veículo nos movimentos pendulares, é importante efetuar a simulação de um cenário de aumento do número de veículos com dois ou mais passageiros.

Assim, considerou-se um aumento de 6% na procura da VAO, um valor conservador com base no aumento previsto em estudos realizados para o INiR.

Neste novo cenário, verifica-se, à semelhança do cenário anterior, um aumento dos tempos de viagem do tráfego proveniente da Costa da Caparica e Fogueteiro. No entanto, é um aumento ligeiro face ao resultado anterior e que resulta na melhoria nos tempos de viagem globais da rede.

Tabela A-8 – Tempos de percurso na alternativa 2

PERCURSO	ATUAL	ALTERNATIVA 2
CL	14.36	15.40
FL	9.74	10.10
AL	7.57	5.26
SL	5.72	5.64

Nos pórticos de portagem, verifica-se um aumento ligeiro das filas de espera na generalidade das opções. No entanto, é de notar o aparecimento de limitações no acesso aos pórticos 4 e 6, resultante e o agravamento da fila de espera do pórtico 3.

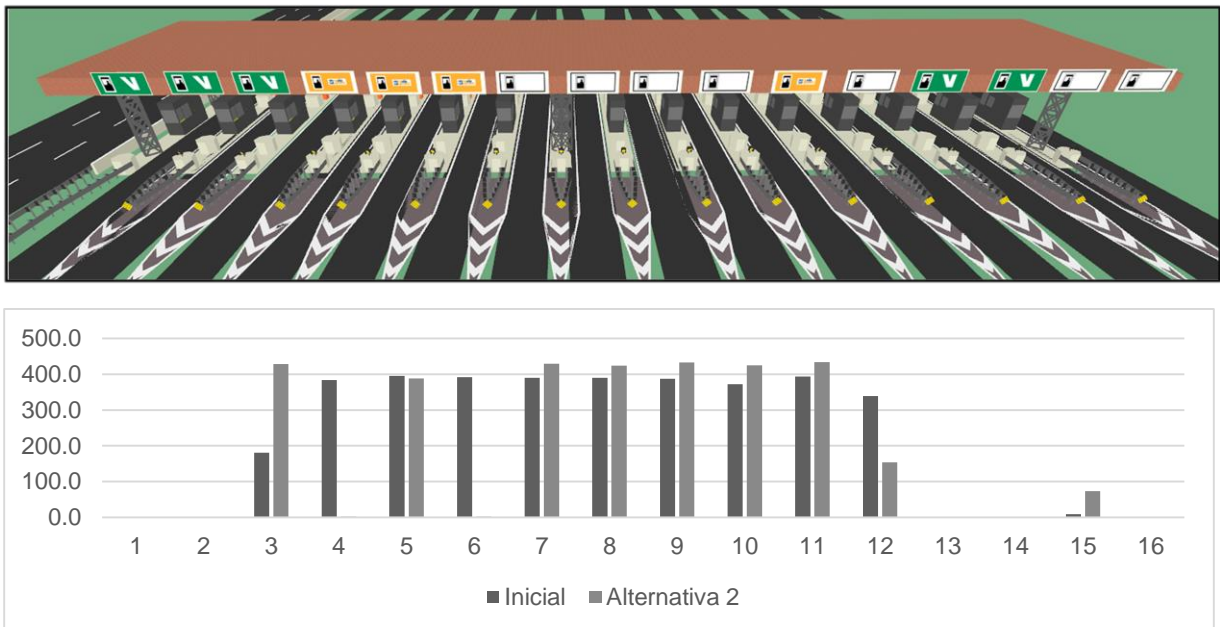


Figura A-32 – Comprimento das filas de espera na alternativa 2.

A.7.8. RESULTADOS

Comparando os indicadores de desempenho para a globalidade do sistema entre a situação atual e as duas alternativas estudadas, verifica-se, por um lado, a redução das filas de espera e o aumento dos tempos de viagem na alternativa 1, e uma melhoria de ambos os indicadores na alternativa 2.

Tabela A-9 – Evolução das condições da rede

	ATUAL	ALTERNATIVA 1	ALTERNATIVA 2
TEMPO DE PERCURSO (MIN)	9.35	13.36	9.10
FILAS DE ESPERA (M)	227.07	190.83	199.53

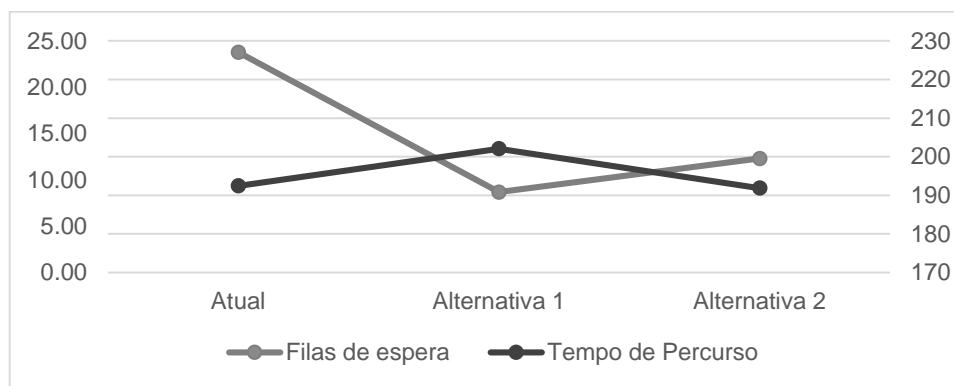


Figura A-33 – Evolução das condições da rede

Como referido anteriormente, este modelo não teve como objetivo uma análise fiel da praça de portagens da ponte 25 de Abril. Primeiramente, não se procedeu à sua calibração para que possa representar com realismo as condições que se verificam no terreno. Por outro lado, ficam por analisar diversas alternativas como uma reorganização geral dos pórticos para acomodar o tráfego proveniente da VAO que pretenda utilizar o modo de pagamento manual ou a reestruturação das permissões de mudança de via ao longo do acesso à praça de portagem de modo a prevenir os efeitos nefastos das manobras de entrecruzamento do tráfego proveniente de Almada que procura a VAO.

No entanto, é notável a vantagem do uso da simulação na avaliação do efeito de alterações numa infraestrutura. Um modelo devidamente calibrado permite a deteção de fenómenos que seriam impossíveis de verificar recorrendo a metodologias analíticas, tornando-se uma ferramenta imprescindível no estudo de infraestruturas com o nível de complexidade aqui apresentado.

A.8. ANEXOS

A.8.1. FLUXOS DE INPUT (VISSIM)

Tabela A-10 – Fluxos de input: Período das 7h às 11h. (Fonte: InIR, IP)

VEIC/HORA	PERCENTAGEM	CL	FL	SA	AL	SC
Ligeiros	90.9	5802	12028	2510	4041	462
Pesados	8.4	121	352	62	82	64
BUS	0.7	18	64	41	143	0

Tabela A-11 – Fluxos de input: Distribuição por classes na situação atual.

	CL	FL	SA	AL	SC	SN
Car VV	505	1046	218	351	40	1305
Car VC	257	534	111	179	21	666
Car VM	268	555	116	186	21	692
Car VV 2+	421	872	182	293	33	1088
Car VC 2+	105	218	45	73	8	272
Car VM 2+	109	227	47	76	9	283
HGV VV	15	43	8	10	8	59
HGV VC	8	22	4	5	4	30
HGV VM	8	23	4	5	4	31
Bus	5	16	10	36	0	26

Tabela A-12 – Fluxos de input: Distribuição por classes na alternativa 1.

	CL	FL	SA	AL	SC	SN
Car VV	505	1046	218	351	40	1305
Car VC	257	534	111	179	21	666
Car VM	268	555	116	186	21	692
Car VV 2+	206	427	89	144	16	533
Car VC 2+	105	218	45	73	8	272
Car VM 2+	109	227	47	76	9	283
HGV VV	15	43	8	10	8	59
HGV VC	8	22	4	5	4	30
HGV VM	8	23	4	5	4	31
Bus	5	16	10	36	0	26

A.8.2. OUTPUT DE FILAS DE ESPERA*Tabela A-13 – Output da simulação: Filas de espera.*

PORTAGEM	INICIAL	CENÁRIO 1	CENÁRIO 2
1.0	0.0	0.0	0.0
2.0	0.0	0.0	0.0
3.0	180.8	0.0	428.0
4.0	384.2	2.7	2.1
5.0	395.5	442.7	388.1
6.0	392.2	453.0	1.8
7.0	389.6	454.1	429.1
8.0	390.2	454.7	424.0
9.0	387.3	455.7	433.1
10.0	372.5	131.7	424.8
11.0	393.3	457.2	433.4
12.0	338.6	149.4	154.1
13.0	0.0	0.0	0.0
14.0	0.0	0.0	0.0
15.0	8.9	52.2	73.7
16.0	0.2	0.1	0.2

A.8.3. OUTPUT DE TEMPOS DE VIAGEM*Tabela A-14 – Output da simulação: Tempos de viagem.*

PERCURSO	ATUAL	ALTERNATIVA 1	ALTERNATIVA 2
CL	14.36	22.55	15.40
FL	9.74	16.50	10.08
AL	7.57	6.49	5.258
SN	5.72	7.89	5.641

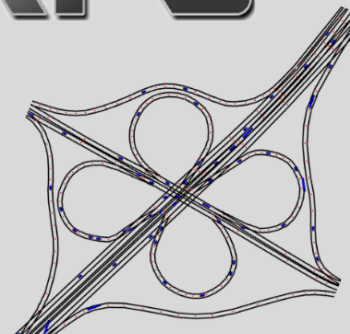
ANEXO I. POSTER TRFS

Apresentado no âmbito do 9º Encontro Transportes em Revista.

DESENVOLVIMENTO DE UM PROTÓTIPO PARA SIMULAÇÃO MICROSCÓPICA DE TRÁFEGO

TRFS

O TRFS (*Traffic Simulator*) é um protótipo desenvolvido no âmbito de uma dissertação de Mestrado em Engenharia Civil, com o objetivo de se tornar uma plataforma para o estudo e implementação de modelos comportamentais de tráfego num ambiente de simulação microscópica.

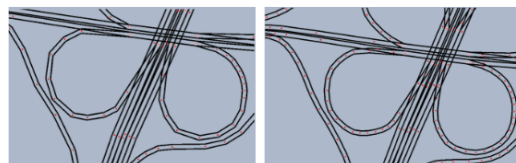


LEITURA DE REDES SIG

Importação direta de redes desenvolvidas em SIG, permitindo a alteração do ambiente de simulação em tempo reduzido.

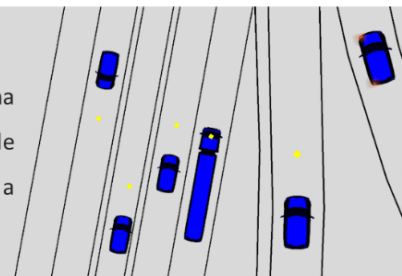


Interpretação automática da geometria da rede e suavização do traçado (Chaikin).



AGENTES AUTÓNOMOS

O recurso a modelos de *car following* (Wiedemann '74) e *lane changing* na modelação do comportamento dos veículos, associados a algoritmos de deteção de proximidade, *path following* e *path finding* (DFS), permite a simulação de agentes que circulam na rede de forma totalmente autónoma.



CONTROLO DA SIMULAÇÃO

O TRFS providencia uma interface gráfica com acesso a todos os parâmetros que influenciam a simulação, permitindo ao utilizador alterar e visualizar os seus efeitos, em tempo real.

