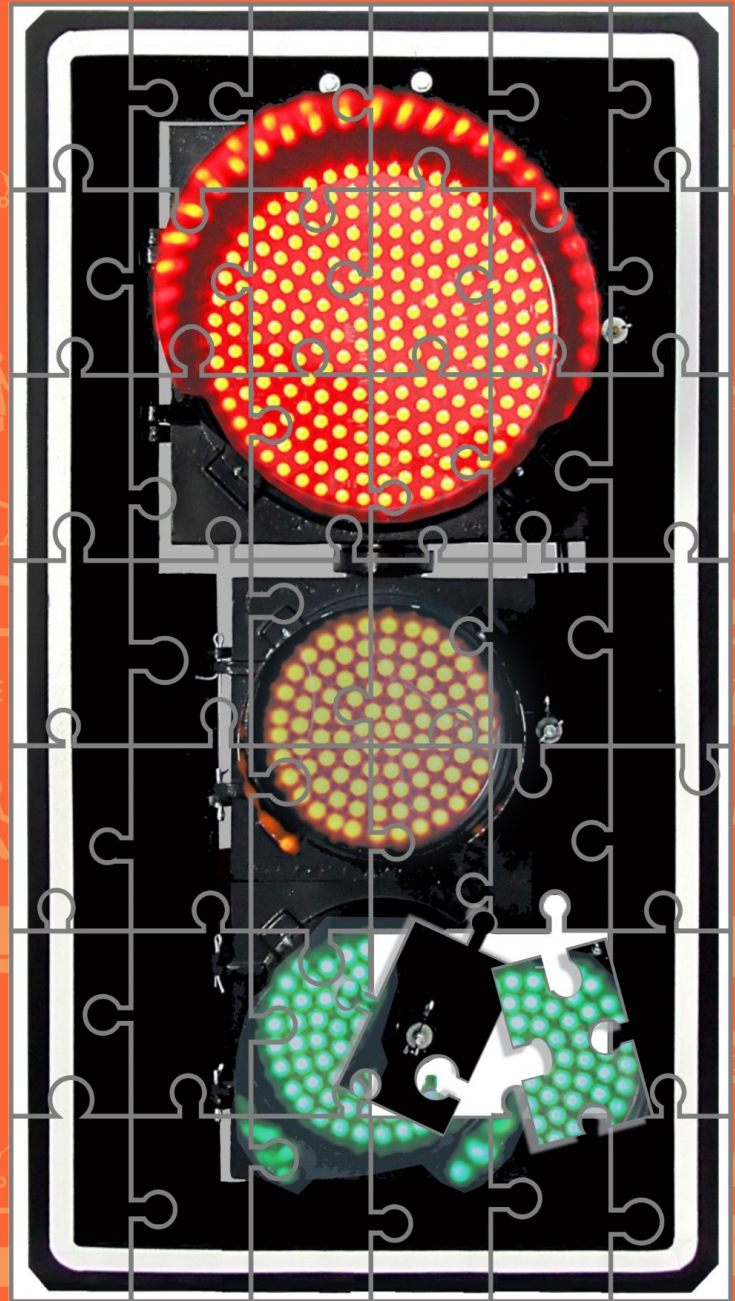




Companhia de Engenharia de Tráfego



Nota Técnica 281

Ming, Sun H.
Caruso, Wladimir S.
Duarte, Tadeu L.
Fernandes, Marcelo A.
Lopes, Denise L.
Santos, Alexandre F.
Tarricone, Nilvio A.
Toledo, Wilson V.
Ueta, Paulo S.

Uma Pesquisa Sobre
Sistemas Adaptativos
de Controle Semafórico
Parte I

Fevereiro/23

Sumário

1. INTRODUÇÃO E OBJETIVO.....	5
1.1 Custo do congestionamento.....	5
1.2 Sistemas adaptativos	6
1.3 Abrangência da pesquisa.....	7
1.4 Glossário	9
2. CLASSIFICAÇÃO DE SISTEMAS SEMAFÓRICOS.....	13
2.1 Sistemas de tempo fixo	13
2.1.1 Sistemas de seleção dinâmica de planos	14
2.1.2 Geração dinâmica de planos.....	14
2.2 Sistemas adaptativos	15
2.2.1 Classificação de sistemas adaptativos quanto à arquitetura ou origem do controle semafórico.....	15
2.2.2 Classificação de sistemas adaptativos quanto ao momento da implementação da temporização semafórica	21
2.2.3 Classificação de sistemas adaptativos quanto à sequência de estágios	22
2.2.4 Classificação de sistemas adaptativos quanto à estrutura	22
2.2.5 Classificação de sistemas adaptativos quanto à estratégia de otimização	22
3. SISTEMAS DE DETECÇÃO.....	23
4. PROTOCOLOS DE COMUNICAÇÃO	27
4.1 Protocolo aberto	27
4.2 Principais protocolos utilizados.....	27
4.3 Conceitos básicos	28
4.3.1 Modelo OSI	29
4.3.2 Estrutura do NTCIP	31
4.3.3 MIB (<i>Management Information Base</i>)	33
4.3.4 AENOR (<i>Asociación Española de Normalización y Certificación</i>)	33
4.3.5 OCIT (<i>Open Communication Interface for Road Traffic Control System</i>)	33
4.3.6 NTCIP (<i>National Transportation Communications for ITS Protocol</i>)	41
4.3.7 UTMC (<i>Urban Traffic Management and Control</i>)	41
4.3.8 SCATS (<i>Sidney Coordinated Adaptive Traffic System</i>)	41
5. SCOOT: <i>Split Cycle Offset Optimization Technique</i>.....	42
5.1 Evolução do SCOOT.....	45
5.2 Detecção	48
5.3 Modelo de tráfego	49
5.4 Otimizador de <i>split</i>	51
5.5 Otimizador de <i>offset</i>	53
5.6 Otimizador do tempo de ciclo	55
5.7 ASTRID e INGRID	56
5.8 Congestionamento	56
5.9 Pontos fracos e desvantagens do SCOOT.....	62

5.9.1	Em termos de desempenho	62
5.9.2	Em termos operacionais	63
6.	SCATS: Sydney Coordinated Adaptive Traffic System	64
6.1	Arquitetura do sistema	64
6.2	Comunicação	65
6.3	Funcionamento do sistema	65
6.4	Detecção	66
6.5	Grau de saturação	66
6.5.1	Grau de saturação conforme definição usual	66
6.5.2	Grau de saturação do SCATS	67
6.6	<i>Car Equivalent Flow (VK)</i>	69
6.7	Tempo de ciclo	69
6.7.1	SCATS 6 (s.d.) [42]	69
6.7.2	Day et al (2010) [45]:	69
6.7.3	Smardi (2001) [41]:	69
6.7.4	Fellendorf et al (1997) [46]:	69
6.7.5	Stevanovic et al (2009) [47]:	70
6.7.6	McCann (2014) [48]:	71
6.8	<i>Split</i>	72
6.8.1	SCATS 6 (s.d.) [42]:	72
6.8.2	Day et al (2010) [45]:	72
6.8.3	Smardi (2001) [41]:	72
6.8.4	Fellendorf et al (1997) [46]:	72
6.8.5	Stevanovic et al (2009) [47]:	72
6.8.6	McCann (2014) [48]:	73
6.9	<i>Offset</i>	74
6.9.1	SCATS 6 (s.d.) [42]:	74
6.9.2	Day et al (2010) [45]:	74
6.9.3	Smardi (2001) [41]:	74
6.9.4	Stevanovic et al (2009) [47]:	75
6.9.5	McCann (2014) [48]:	75
6.9.6	Considerações sobre <i>offset</i>	75
6.10	Congestionamento	79
7.	ITACA: Intelligent Traffic Adaptive Control Area	80
8.	TUC: Traffic-Responsive Urban Control	85
8.1	Estratégia do TUC	85
8.2	Arquitetura do TUC	87
8.3	Comunicação e protocolo	87
8.4	Detecção	87
8.5	Abordagem “ <i>store-and-forward</i> ”	88
8.6	Otimização de <i>split</i>	89

8.7	Otimização de tempo de ciclo	92
8.8	Otimização de <i>offset</i>	93
8.9	Considerações	94
9.	CONTREAL: Controle de Tráfego por Área em Tempo Real	97
10.	SMART GREEN	99
10.1	<i>Smart Intersection</i>	99
10.2	<i>Smart Corridor</i>	100
10.3	<i>Smart Green</i>	100
11.	INES - <i>Intelligent Network Control</i>	101
11.1	Introdução.....	101
11.2	Análise do vídeo (Schlothauer & Wauer, 2022a) [69].....	102
11.2.1	Cenário e capacidade.....	102
11.2.2	Arquitetura do INES.....	104
11.2.3	Módulos do INES.....	106
11.2.4	Procurando o cenário ótimo.....	107
11.2.5	Calculando o grau de saturação (<i>Saturation</i>).....	109
11.2.6	Função de Utilidade (<i>Utility Function</i>)	110
11.2.7	Atraso (<i>Loss Time</i>).....	113
11.2.8	Detectores estratégicos	113
11.2.9	Redistribuição do tempo de verde e períodos de permissão (" <i>permission periods</i> ").....	115
11.2.10	<i>Offset</i>	117
11.3	Outras informações obtidas do Schlothauer & Wauer (2022b) [70].....	117
11.3.1	Posicionamento dos detectores	117
11.3.2	Comparação de arquitetura INES e SCOOT.....	118
11.3.3	Otimização da defasagem.....	119
11.4	Conclusões	120
12.	TrafGo	122
13.	BIBLIOGRAFIA	124

1. INTRODUÇÃO E OBJETIVO

Os grandes centros urbanos constituem sistemas complexos. Em particular, o sistema viário urbano é um grande sistema não linear complexo. É composto por vias com diferentes usos e características, com interseções e rampas de acesso, com fluxo misto de veículos de passageiros, ônibus, veículos de carga, bicicletas e pedestres, todos compartilhando e disputando o mesmo espaço. Os congestionamentos de trânsito ocorrem com frequência e causam todos os tipos de problemas e desafios, o que afeta a vida diária das pessoas. Os longos tempos de deslocamento afetam a qualidade de vida e degradam o ambiente urbano. A redução dos congestionamentos de trânsito não só reduz o tempo de deslocamento e a poluição ambiental, mas também melhora a segurança das viagens e a qualidade da vida das pessoas.

1.1 Custo do congestionamento

O custo dos congestionamentos nas grandes cidades alcança cifras de bilhões de reais anuais. Por exemplo, na cidade de São Paulo foram feitas as seguintes estimativas pelo documento “Os Custos do Congestionamento na Cidade de S. Paulo” [1] em 2000, 2004 e 2008, em reais:

Tabela 1 – Estimativa do custo de congestionamento na cidade de São Paulo (Valores em R\$)

	2000	2004	2008
Horas perdidas	6.991.996.050	14.076.392.149	26.634.269.801
Consumo de combustível	2.882.057.546	3.572.285.919	4.155.455.272
Poluição	270.672.171	343.274.015	406.985.538
Transporte de carga	1.019.098.353	1.388.063.866	1.955.718.654
Total	11.163.824.120	19.380.015.949	33.152.429.265

Fonte: “Os Custos do Congestionamento na Cidade de S. Paulo” [1]

Apenas para efeitos comparativos, Cintra (2014) [2] fornece as seguintes estimativas para o custo do congestionamento na cidade de São Paulo nos anos de 2004 e 2008:

Tabela 2 – Estimativa do custo de congestionamento na cidade de São Paulo (Valores em R\$)

	2000	2004	2008
Horas perdidas	---	13.128.172.049	24.282.318.282
Consumo de combustível	---	4.011.850.791	4.991.708.808
Poluição	---	548.333.268	684.837.747
Transporte de carga	---	2.740.466.789	4.431.435.429
Total	---	20.428.822.897	34.390.300.266

Fonte: Cintra, 2014 [2]

Cintra (2014) [2] estima que em 2012 o custo total de lentidão na cidade de São Paulo foi de R\$ 40.159.507.860.

Nessas estimativas não estão contabilizados os custos relativos à segurança viária, bem como outros custos de caráter qualitativo, de difícil quantificação.

Na Europa, a situação não é diferente. Segundo o artigo de Papageorgiou et al (2003) [3], “a seguinte impressionante afirmação foi incluída no *White Paper – European Transport Policy for 2010* da Comissão Europeia: Por causa do congestionamento há um sério risco de a Europa perder competitividade econômica. O mais recente estudo mostrou que o custo do congestionamento corresponde a 0,5% do PIB. Previsões para os próximos 10 anos mostram que o nível de congestionamento irá crescer significativamente até 2010. O custo atribuído ao congestionamento irá crescer 142%, alcançando € 80 bilhões por ano, que é aproximadamente 1% do PIB (estimado para 2010)”.

Um dos fatores que mais influenciam no tempo de deslocamento são os semáforos, pois envolvem longos períodos de espera. Assim, soluções para reduzir congestionamentos sempre envolvem semáforos. Segundo Souza et al (2017) [4]:

Estima-se que semáforos mal programados sejam responsáveis por cerca de 10% dos atrasos diários nas principais rodovias norte-americanas, considerando todos os veículos (USDOT, 2004). O conceito de tempo é relativo, sendo influenciado pelo estado de espírito do indivíduo. Dessa forma, mesmo poucos segundos perdidos em uma interseção podem exercer expressivo impacto no nível de serviço percebido pelo usuário, particularmente nos casos em que a programação semafórica aloca significativa parcela do tempo de verde para um sentido da via que apresenta reduzido volume de veículos.

De fato, nada é mais irritante para o usuário que perceber o precioso tempo de verde sendo desperdiçado na via transversal enquanto ele fica parado de forma desnecessária.

Às vezes pode-se pensar que a economia de alguns poucos segundos de espera numa aproximação semafórica, decorrente de uma melhoria na programação semafórica, em nada representa para o usuário e que nem seja percebido. Entretanto, esses poucos segundos multiplicados pelo volume de veículos vão representar milhões de horas anuais. Fazendo-se uma analogia, imagine uma redução de 10 centavos no pedágio de uma via de alto volume de tráfego. Individualmente, a economia de 10 centavos pode representar muito pouco e nem seja percebida pelo usuário como significativa. Mas, para a concessionária da rodovia, essa redução pode representar uma perda de receita da ordem de milhões de reais em um ano.

1.2 Sistemas adaptativos

Entre todas as soluções possíveis, o controle adaptativo de semáforos (*Adaptive Traffic Signal Control – ATSC*) é comumente pensado como o método mais importante e eficaz, por não envolver a necessidade de realização de obras viárias, como a construção de novas vias, pontes e viadutos. Os algoritmos de ATSC têm evoluído rapidamente, especialmente nas últimas décadas. Como resultado, vários sistemas de ATSC têm sido amplamente implementados no mundo, tornando o ATSC um componente importante do sistema de transporte inteligente (em inglês, *Intelligent Transportation System - ITS*).

Em geral, os sistemas adaptativos apresentam uma grande relação de benefício-custo, de forma que qualquer investimento nessa área representa uma alta taxa de retorno. Segundo o documento IDOM (2019 – D3) [5] uma redução de 10% do tempo de deslocamento representaria uma economia de até 40 milhões de reais diários (levando em conta somente os atrasos dos trabalhadores).

Muitas vezes, depara-se com a seguinte questão: os sistemas adaptativos são mais eficientes em períodos e/ou locais com baixo trânsito ou em interseções saturadas, em situações de lentidão e congestionamento?

À primeira vista, a resposta pode parecer que os sistemas adaptativos apresentariam melhores resultados em situações com baixo trânsito, pois o sistema ajustaria os tempos semafóricos de acordo com a aleatoriedade e as flutuações do tráfego. Entretanto, a resposta correta é: os sistemas adaptativos apresentam pouco benefício em locais e períodos de baixo trânsito. A explicação é que o sistema adaptativo pode reduzir, para cada veículo, alguns segundos na espera semafórica. Esses poucos segundos, multiplicados por poucos veículos, representam um baixo benefício. Ao contrário, em situações de grande demanda de tráfego, os mesmos poucos segundos multiplicado por um alto volume de veículos vão representar um alto benefício.

Além disso, os sistemas adaptativos podem:

- Reduzir a intensidade e a gravidade dos congestionamentos;
- Reduzir a área do congestionamento, evitando que se propague de forma descontrolada; e
- Postergar o início e antecipar o fim de congestionamentos, reduzindo o período do congestionamento.

Para efeitos do presente trabalho, entende-se por sistema adaptativo aquele que fornece uma temporização semafórica que varia automaticamente em função da demanda de tráfego vigente.

Em geral, os sistemas adaptativos incorporam no seu algoritmo um processo de otimização.

No processo de otimização existe uma representação matemática do critério de eficiência adotado. Essa representação é a função objetivo, que é influenciada pelas variáveis de controle do problema, isto é, aquelas pelas quais a minimização ou a maximização estão sujeitas para um ótimo da função¹.

Os sistemas adaptativos podem conter algoritmos de simulação e/ou algoritmos que maximizam ou minimizam funções objetivo e/ou algoritmos que englobam métodos heurísticos.

O algoritmo do sistema adaptativo pode ou não ser baseado em modelos de tráfego.

O objetivo do presente trabalho é fazer uma pesquisa, a mais abrangente possível, sobre o estado da arte dos sistemas adaptativos existentes, compilando dados e informações sobre a sua arquitetura e funcionamento e colocando tudo num só lugar, de forma a facilitar eventuais futuras consultas.

Por meio do presente trabalho, será possível fazer uma “viagem” aos sistemas adaptativos existentes mais conhecidos, identificando o seu funcionamento e os correspondentes processos de otimização, com maior ou menor profundidade (dependendo do material coletado na pesquisa), discutindo os seus pontos fortes e suas fraquezas, além de avaliações de desempenho por testes de campo ou por meio de simulações, bem como expondo as dúvidas dos autores do presente trabalho em relação a questões que não puderam ser esclarecidas com o material encontrado.

1.3 Abrangência da pesquisa

Para demonstrar a abrangência da pesquisa realizada, a [Tabela 3](#) mostra a quantidade de material encontrado na pesquisa, até o momento, em relação aos sistemas pesquisados.

¹ Fonte: Adaptado de:

<https://www.dicionarioinformal.com.br/fun%C3%A7%C3%A3o%20objetivo/#:~:text=Para%20um%20problema%20de%20otimiza%C3%A7%C3%A3o,para%20um%20%C3%B3timo%20da%20fun%C3%A7%C3%A3o.>

(Acesso em 01/07/2022)

**Tabela 3 – Abrangência da pesquisa realizada
(artigos, trabalhos, documentos e sites)**

	Sistema	Quantidade	%
1	SCOOT	212	56,1
2	SCATS	83	22,0
3	ITACA	7	1,9
4	TUC	25	6,6
5	CONTREAL	3	0,8
6	SMART GREEN	3	0,8
7	INES	5	1,3
8	TrafGo	2	0,5
9	RHODES	25	6,6
10	OPAC	24	6,3
11	ACS LITE	15	4,0
12	LA ATCS	5	1,3
13	InSync	16	4,2
14	SURTRAC	10	2,6
15	KADENCE	4	1,1
16	RTACL	1	0,3
17	MARLIN ATSC	3	0,8
18	ARTICS	1	0,3
19	ADIMOT	6	1,6
20	MOTION	11	2,9
21	UTOPIA/SPOT	12	3,2
22	PRODYN	8	2,1
23	CRONOS	4	1,1
24	BALANCE/EPICS	8	2,1
25	UTMS	4	1,1
26	CoSiCoSt	3	0,8
27	ATLTC	2	0,5
28	GLIDE	1	0,3
	TOTAL	378	

No total, a pesquisa identificou, até o presente momento, 28 sistemas adaptativos distintos e 378 documentos, entre artigos, relatórios, manuais, sites, etc.

Muitos dos artigos pesquisados tratam simultaneamente de vários sistemas, daí que a soma dos percentuais irá totalizar mais que 100%.

Devido à extensão, abrangência e complexidade da pesquisa, o trabalho será dividido em partes, a saber:

- A primeira parte, objeto da presente Nota Técnica, contém a introdução e objetivo da pesquisa (Capítulo 1), a classificação de sistemas semafóricos quanto a diferentes critérios (Capítulo 2), sistemas de detecção (Capítulo 3), protocolos de comunicação (Capítulo 4) e a descrição da arquitetura e do funcionamento dos seguintes sistemas:
 - SCOOT (Capítulo 5)
 - SCATS (Capítulo 6)

- ITACA (Capítulo 7)
- TUC (Capítulo 8)
- CONTREAL (Capítulo 9)
- SMART GREEN (Capítulo 10)
- INES (Capítulo 11)
- TrafGo (Capítulo 12)

Esta Nota Técnica contém também a descrição de alguns sistemas de controle semafórico local (item 2.2.1.2.2).

- A segunda parte, a ser elaborada futuramente, conterá a descrição da arquitetura e do funcionamento dos demais sistemas identificados na pesquisa (Tabela 3).
- Uma terceira parte, também a ser elaborada futuramente, conterá uma compilação de informações encontradas na pesquisa sobre avaliações de desempenho dos sistemas adaptativos e as perspectivas para o futuro.

O nível de detalhamento na descrição de cada sistema foi função do material disponível e do que foi possível localizar na Internet. No caso particular do SCOOT, foi possível fazer um detalhamento mais completo devido ao material disponível.

Deve-se ressaltar que os métodos utilizados na otimização são muitas vezes proprietários e não são publicados de forma detalhada, tornando-se difícil de avaliar a sua efetividade e eficiência. Isto é particularmente verdadeiro para sistemas desenvolvidos exclusivamente por empresas comerciais, sem a participação de meios acadêmicos ou de órgãos governamentais.

Para maior clareza e compreensão do presente trabalho, deve-se considerar os termos técnicos utilizados ao longo do texto e elencados no Glossário a seguir (item 1.4).

1.4 Glossário

Para efeitos do presente trabalho, os seguintes termos e expressões apresentam os significados indicados na Tabela 4 – Glossário

Tabela 4 – Glossário

Termo	Significado
Atraso de grupo semafórico	<ul style="list-style-type: none"> Nos casos em que há mais de um grupo semafórico com direito de passagem no estágio, para obter maior eficiência na operação semafórica durante os períodos de transição de estágios, pode ser necessário atrasar a perda do direito de passagem (atrasar o término do verde) ou atrasar o ganho do direito de passagem (atrasar o início do verde) de alguns grupos semafóricos envolvidos na transição².
Ciclo	<ul style="list-style-type: none"> Sequência completa dos estágios. Sequência completa das indicações luminosas em um grupo semafórico.
Controle semafórico	<ul style="list-style-type: none"> Controle semafórico está associado ao agente³ responsável pela implementação dos tempos do semáforo.
Entreverdes	<ul style="list-style-type: none"> Entreverdes é o período de transição entre os grupos semafóricos que irão perder o direito de passagem para os grupos semafóricos que irão ganhar o direito de passagem. No caso de grupos semafóricos veiculares, o entreverdes compõe-se do período de amarelo seguido do período de vermelho de limpeza. No caso de grupos semafóricos de pedestres, o entreverdes consiste do período de vermelho intermitente seguido do período de vermelho de limpeza
Estágio	<ul style="list-style-type: none"> Conjunto de intervalos luminosos formados por um intervalo de verde e período de transição para o estágio seguinte. Parte do ciclo em que não se altera a configuração do direito de passagem na interseção mais o período de transição do direito de passagem.
Fluxo de saturação	<ul style="list-style-type: none"> Maior número de veículos que consegue atravessar a linha de retenção durante uma hora, sendo que neste período o semáforo permanece continuamente em verde.

² Para maiores detalhes sobre Atraso de Grupo, ver a Nota Técnica 258 – Conceituação de Entreverdes, disponível em http://www.cetsp.com.br/media/829864/nt_258.pdf (Acesso em 17/11/2022).

³ O agente pode ser uma pessoa (caso o controle seja realizado manualmente) ou ainda um controlador ou central de controle, por exemplo.

<i>Gap</i>	<ul style="list-style-type: none"> Intervalo de tempo entre a passagem do para-choque traseiro de um veículo e do para-choque dianteiro do veículo subsequente por uma determinada seção transversal da via, independentemente de utilizarem ou não a mesma faixa de rolamento⁴.
Grau de saturação	<ul style="list-style-type: none"> Razão entre o fluxo (demanda) e o máximo fluxo que pode ser escoado.⁵
Grupo semafórico	<ul style="list-style-type: none"> Conjunto de grupos focais, ligados nos mesmos circuitos elétricos e que, portanto, apresentam sempre indicação luminosa idêntica.
Interoperabilidade	<ul style="list-style-type: none"> Capacidade de componentes do sistema de diferentes fornecedores de se comunicarem entre si.
Intervalo luminoso	<ul style="list-style-type: none"> Parte do ciclo em que as indicações luminosas de todos os grupos semafóricos permanecem constantes.
ITS	<ul style="list-style-type: none"> Sigla de <i>Intelligent Transportation System</i>. Conjunto de aplicações e tecnologias relativas a diferentes modos de transporte, gerenciamento e controle de tráfego e transporte.
<i>Link</i>	<ul style="list-style-type: none"> Entidade para representar uma ligação em aproximação semaforizada.
<i>Offset</i>	<ul style="list-style-type: none"> Defasagem⁶. Considerando duas interseções adjacentes e selecionado um ponto do ciclo tomado como referência, defasagem é o tempo decorrido entre a ocorrência desse ponto numa interseção em relação à ocorrência do mesmo ponto na outra interseção. No protocolo OCIT, <i>offset</i> é entendido como atraso de grupo.
Semáforo	<ul style="list-style-type: none"> Interseção semaforizada. Conjunto de elementos físicos e funcionais que determina, através de indicações luminosas, de forma alternativa, o direito de passagem de veículos e pedestres num determinado local onde há conflito de movimentos.
<i>Split</i>	<ul style="list-style-type: none"> Fração de verde em relação ao ciclo.
V2X	<ul style="list-style-type: none"> <i>Vehicle to X (Vehicle to everything)</i> Comunicações envolvendo: V2I (veículo para infraestrutura de transporte), V2P (veículo para pedestres), V2N (veículo para rede) e V2V (veículo para veículo).

⁴ Fonte: Adaptado de Sinal de Trânsito <http://www.sinaldetransito.com.br/glossario.php?ID=336> (acesso em 15/07/2022)

⁵ $x = \frac{F}{F_S} \times \frac{C}{g}$ onde:

x = grau de saturação

F = fluxo

F_S = fluxo de saturação

C = tempo de ciclo

g = verde efetivo

Ver o item 6.5.1

⁶ A defasagem e o *offset* podem assumir significados distintos. Por exemplo: o *offset* da interseção A é 2 e o *offset* da interseção B é 5. Então, a defasagem entre A e B é de $5 - 2 = 3$, isto é, B abre 3 segundos depois de A.

Muitas vezes, em engenharia de tráfego, os termos podem ter mais de um significado. Contudo, para compreensão do presente texto, os termos constantes no glossário devem ser interpretados conforme os significados indicados na [Tabela 4 – Glossário](#)

Termos ou expressões específicas que não constem no glossário acima têm seus significados explicitados ao longo do texto.

2. CLASSIFICAÇÃO DE SISTEMAS SEMAFÓRICOS

Os sistemas semafóricos podem ser classificados em dois grandes grupos: sistemas de tempo fixo e sistemas adaptativos.

2.1 Sistemas de tempo fixo

Os sistemas de tempo fixo operam por meio de planos semafóricos. O plano semafórico é o conjunto de configurações as quais determinam a sequência, o instante de início e a duração das luzes exibidas pelo semáforo.

Pode ser elaborada uma biblioteca de planos semafóricos, os quais normalmente são selecionados em função do horário do dia e/ou do dia da semana, por meio de uma tabela horária.

Durante a vigência de um plano, a sequência, o instante de início e a duração das luzes exibidas pelo semáforo são invariáveis.

A elaboração dos planos semafóricos é baseada nas condições médias de tráfego verificadas para o período de vigência dos planos.

Também é possível a elaboração de planos para dias específicos do ano, como feriados e/ou emendas de feriados.

Os sistemas de tempo fixo não necessitam de detectores, pois os planos são implementados de acordo com o horário e/ou dia de semana, independentemente das condições de tráfego reais da via no período de vigência do plano. O único uso de detectores em sistemas de tempo fixo é para acionar estágios dispensáveis⁷.

No caso particular de uma interseção sem coordenação com interseções vizinhas, com apenas 2 estágios, sendo que um deles é dispensável, tem-se uma variante de sistema de tempo fixo, denominado sistema semi-atuado. O estágio que não depende de solicitação de detector é denominado de estágio principal. Se, durante todo o tempo programado do estágio principal, não houver solicitação do detector, o estágio principal continua com direito de passagem, até que haja um acionamento do detector do estágio dispensável. O estágio dispensável entra em vigor depois de decorrido um intervalo de tempo após o acionamento do seu detector associado, denominado “tempo de retardo”, desde que já tenha sido cumprido o tempo de verde programado para o estágio principal, contado a partir do final do vermelho (ou início de verde) do grupo semafórico correspondente até o instante do acionamento do detector. Durante a vigência de um plano, a sequência, o instante de início e a duração das luzes exibidas pelo semáforo em cada estágio são invariáveis, exceto pelo fato de o estágio principal poder ter a sua duração prolongada para além do seu tempo programado.

Os planos de tempo fixo não acompanham a variação do trânsito de longo prazo, exigindo revisões periódicas (“*signal re-timing*”). Robertson et al (1991) [6] cita que Bell et al (1986) [7] estima que os planos semafóricos se defasam causando 3% de atraso extra por ano.

⁷ Estágio dispensável é aquele que só ocorre se tiver sido solicitado por um sensor, seja uma botoeira de pedestres, seja um detector de veículos.

Segundo o artigo de Robertson et al (1982) [8], os planos de tráfego apresentam em geral vários anos de idade, sendo que o Relatório do TRRL (Hunt et al, 1981) [9] apresenta evidência de que planos defasados causam 4% de atraso extra para cada ano de idade.

Contudo, as atividades de revisão periódica consomem consideráveis recursos para a atualização dos planos.

Nesse sentido, foram desenvolvidos sistemas que procuram, ou selecionar o melhor plano para as condições vigentes do trânsito (ver o item 2.1.1, a seguir), ou gerar, de forma automática, os planos de acordo com as condições vigentes do trânsito (ver o item 2.1.2, adiante). Esses sistemas continuam sendo de tempo fixo, pois, uma vez implementado o plano, a temporização não se altera até a entrada de um novo plano. O que é dinâmico não é o plano, mas a sua seleção (no primeiro caso) ou a sua geração (no segundo caso).

2.1.1 Sistemas de seleção dinâmica de planos

Uma variante de sistema de tempo fixo são os sistemas de seleção dinâmica de planos. Nesses sistemas, os planos não são selecionados por meio de uma tabela horária, mas são implementados de forma dinâmica de acordo com regras previamente estabelecidas, em função das condições de tráfego verificadas na via.

Durante a vigência de um plano, a sequência, o instante de início e a duração das luzes exibidas pelo semáforo são invariáveis.

A elaboração dos planos semafóricos é baseada em condições de tráfego associadas às regras utilizadas para a sua seleção.

Esses sistemas necessitam de detectores, que podem ser fixos ou móveis.

O ponto crítico desses sistemas reside na dificuldade para a elaboração de planos adequados para atender as condições de tráfego que estejam associadas às regras, sendo que a própria definição dessas regras constitui outro grande desafio para o bom desempenho do sistema.

Além disso, o processo na troca de planos recai nos mesmos problemas descritos no item 2.1.2, a seguir.

2.1.2 Geração dinâmica de planos

As primeiras tentativas de desenvolvimento de sistemas adaptativos se deram no sentido de gerar planos de forma dinâmica e automática, de forma que não houvesse mais a necessidade de elaboração manual dos planos, sendo que os planos gerados são adaptados às condições de tráfego em tempo real.

Contudo, segundo o artigo de Hunt et al (1982) [10], essas tentativas não foram bem sucedidas pelas seguintes razões:

- (a) Mudança frequente de planos. Alguns dos problemas dos primeiros sistemas adaptativos requeriam que os novos planos, calculados de forma *on-line*, fossem implementados assim que possível. Mesmo os melhores métodos de mudança de planos causam significativo atraso na transição de planos, de forma que um novo plano deve permanecer em operação por pelo menos 10 minutos para oferecer algum benefício (que compensasse os distúrbios na troca de planos).

- (b) Previsão inadequada. De (a) foi visto que há necessidade de que os planos sejam gerados com base no tráfego futuro de pelo menos 10 minutos. A variação randômica do tráfego torna essa previsão muito difícil e há a necessidade do uso de dados históricos para ajudar a identificar tendências.
- (c) Lentidão da resposta. Quando ocorrem eventos inesperados, a resposta é atrasada pelo elemento histórico usado na previsão e, portanto, também é atrasada a identificação da necessidade de um novo plano.
- (d) Efeitos de planos equivocados. Eventos não esperados ou dados de detectores defeituosos podem causar planos equivocados que não podem ser corrigidos até a entrada de um plano correto.

2.2 Sistemas adaptativos

2.2.1 Classificação de sistemas adaptativos quanto à arquitetura ou origem do controle semafórico

Os sistemas adaptativos podem ser classificados de acordo com a origem do controle semafórico em:

- Controle semafórico centralizado: quando a implementação da temporização semafórica é feita por um único agente central.
- Controle semafórico local: quando a implementação da temporização semafórica é feita por um equipamento localizado no cruzamento, normalmente o próprio controlador.
- Controle semafórico distribuído ou hierárquico: quando a implementação da temporização semafórica é feita de forma distribuída em níveis de controle, com objetivos distintos em cada nível. Essa arquitetura também é conhecida como multiagente, no sentido de que o controle é feito por mais de um agente. Por exemplo, a implementação do *split* é feita em nível local, enquanto a implementação do *offset* e do tempo de ciclo são feitas em nível central.
- Controle semafórico descentralizado: a rede é decomposta em regiões, com número variável de interseções. Com a decomposição, a implementação da temporização semafórica de uma interseção específica só precisa considerar o estado das variáveis em uma vizinhança predefinida, em vez de toda a rede (como no controle semafórico centralizado).

2.2.1.1 Controle semafórico centralizado

No controle centralizado, o controle de toda a rede é feito por um único agente central.

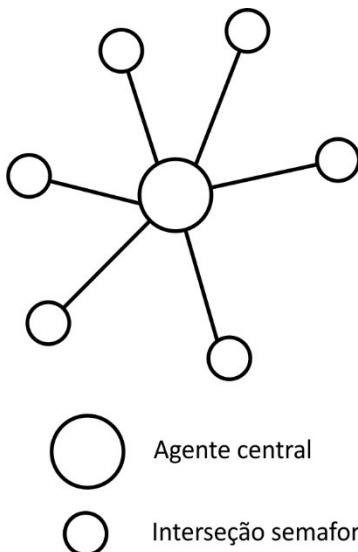


Figura 1 – Sistema de controle centralizado

Fonte: <https://www.geeksforgeeks.org/comparison-centralized-decentralized-and-distributed-systems/>

(Acesso em 11/07/2022)

De acordo com Adacher et al (2020) [11]:

São otimizados vários parâmetros (ou seja, duração do ciclo, *splits* e *offsets*) para todas as interseções da rede. No entanto, a otimização de toda a rede é um problema “NP-hard” (NP = *Nondeterministic Polynomial*⁸) e uma otimização centralizada pode não ser escalável e aplicável a grandes redes.

É evidente que um sistema centralizado é teoricamente capaz de encontrar soluções ótimas. No entanto, o tempo de computação de uma otimização centralizada aumenta exponencialmente com a dimensão da rede urbana, inviabilizando assim o seu processamento. Também é esperado que um sistema centralizado possa teoricamente fornecer uma política de controle mais eficaz do que suas contrapartes descentralizadas, com uma melhor coordenação entre os cruzamentos.

Segundo Chow et al (2019) [12]:

Espera-se que a eficiência e a resiliência das redes semaforicas possam ser significativamente melhoradas coordenando todos os cruzamentos da rede por meio de uma estrutura centralizada, como o bem estabelecido sistema SCOOT. No entanto, a complexidade das redes urbanas muitas vezes dificulta a gestão por um único agente central.

Nota-se que a complexidade do cenário centralizado inviabiliza a implementação de algoritmos eficazes, como o algoritmo de buscas repetitivas baseadas em simulação, como a abordagem de “*hill-climbing*” adotada no Sistema SCOOT.

O procedimento de “*hill climbing*” será detalhado mais adiante no item 5.

Ainda segundo Chow et al (2019) [12]:

O algoritmo centralizado supera as suas contrapartes descentralizadas em uma variedade de circunstâncias diferentes, devido à sua incorporação de visão ampla da topologia da rede e da distribuição de demanda. Demonstrou ser capaz de derivar políticas de controle que podem alcançar maior eficiência global, em termos de redução geral de atrasos, em particular sob alta demanda de cenários. No entanto, a abordagem centralizada requer um único agente de controle central para lidar com todo o tráfego. Tal complexidade do problema proíbe a implementação de algoritmos de solução eficaz além de simulações repetitivas.

⁸ Suspeita-se de que não existam algoritmos de tempo polinomial para problemas NP-hard, mas isso não foi comprovado.

Fonte:

<https://en.wikipedia.org/wiki/NP-hardness#:~:text=In%20computational%20complexity%20theory%2C%20NP,is%20the%20subset%20sum%20problem>
(Acesso em 28/06/2022)

Um algoritmo de tempo polinomial é um algoritmo cujo tempo de execução é dado por um polinômio no tamanho da entrada ou pode ser limitado por tal polinômio. Problemas que podem ser resolvidos por um algoritmo de tempo polinomial são chamados de problemas tratáveis.

Fonte:

<https://www.seas.upenn.edu/~cit596/notes/dave/p-and-np2.html> (Acesso em 28/06/2022)

Existem problemas para os quais não existem algoritmos eficientes. Significa dizer que há (grandes) instâncias desses problemas que não poderão ser resolvidas por programas reais, executando em computadores reais. Problemas deste tipo são ditos intratáveis.

Fonte: https://www.ic.unicamp.br/~mc102/mc102-2017/aulas/p_arnaldo/texto/parte9_problemas2_62_79.pdf
(Acesso em 28/06/2022).

2.2.1.2 Controle semafórico local

No controle semafórico local, cada interseção é responsável pelo seu próprio controle.

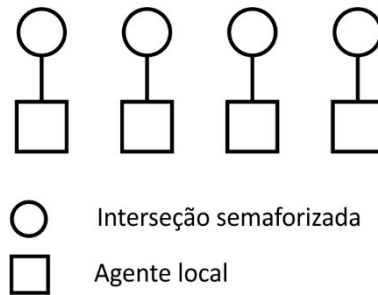


Figura 2 – Controle local

O controle semafórico local é mais apropriado para interseções isoladas.

Para corredores ou redes semafóricas, o controle local não é apropriado pois faltam as informações necessárias das interseções vizinhas para que se alcance uma coordenação semafórica ótima. O problema é que o equipamento local ou o controlador pode analisar uma parte dos dados, mas não tem o contexto geral da rede.

Para a otimização do *offset*, além dos dados de cada *link*, é necessário conhecer os estágios nas interseções a montante e a jusante, os atrasos de todas as aproximações das interseções adjacentes e o fluxo que está saindo na direção oposta, que vai para outra interseção. Assim, o equipamento local não dispõe de todas as informações necessárias para tomar a decisão de *offset*.

Da mesma forma, no controle semafórico local não há as informações necessárias para a otimização do tempo de ciclo da rede.

Essa análise é válida para sistema de detecção com sensores fixos. Com o advento da comunicação V2X e de equipamentos RSUs (*Roadside Units*), é possível que se torne factível a otimização de toda a rede de forma local.

Outro problema crítico do controle local, é que é necessário o uso exclusivo de controlador ou de equipamento do fabricante detentor do sistema, não havendo nenhuma possibilidade de interoperabilidade.

2.2.1.2.1 Sistemas atuados

Os sistemas atuados podem ser considerados como sistemas adaptativos no sentido de que o *split* e o tempo de ciclo variam conforme a demanda de tráfego. Entretanto, nos sistemas atuados não há propriamente um processo de otimização no sentido de que não há uma função objetivo a ser maximizada ou minimizada.

No sistema atuado, a duração dos estágios é decorrente da ativação de detectores veiculares, permitindo extensões de verde até um máximo programado. O tempo de verde do estágio varia entre valores programáveis de verde mínimo e de verde máximo, em função das solicitações provenientes dos detectores veiculares. A cada uma dessas solicitações, o respectivo tempo de verde, quando maior que o verde mínimo,

será incrementado de um período de tempo programável, denominado "extensão de verde", até o *gap out*⁹ ou até atingir o verde máximo. A duração da extensão de verde é programada em função do *gap* entre veículos.

A extensão de verde para uma aproximação depende do tráfego detectado nessa aproximação, sem levar em conta o tamanho das filas nas outras aproximações. Assim, não há um processo de otimização que considere a interseção como um todo.

Embora o sistema atuado seja específico para interseções isoladas, na América do Norte é comum o uso de sistemas atuados coordenados. Porém, neste caso, devido à necessidade de coordenação, o tempo de ciclo deve ser mantido fixo. Como o tempo de ciclo da rede é definido como sendo o tempo de ciclo requerido pela interseção mais saturada, nas demais interseções pode haver "sobras de verde". Neste caso, pode-se implementar um esquema similar à atuação, "cortando" o verde quando não houver mais demanda e transferir o verde "economizado" para outros estágios.

2.2.1.2.2 Alguns exemplos de sistemas adaptativos com controle semafórico local

a) MOVA – *Microprocessor Optimised Vehicle Actuation*

MOVA é um sistema de padrão do Reino Unido.

De acordo com MOVA (s.d.) [13]:

MOVA é um sistema projetado para interseções isoladas. Trata-se de um sistema atuado com funcionalidades adicionais.

Originalmente projetado pelo TRL (*Transport Research Laboratory*) na década de 1980, MOVA é atualmente uma estratégia bem consolidada para interseções isoladas. MOVA está instalado em aproximadamente metade das interseções do Reino Unido que suportam tal estratégia, e esse número cresce num ritmo de cerca de 200 a 300 instalações por ano.

MOVA M8 é a última versão, lançada em 2017, e representa um significativo *upgrade* em relação às versões anteriores, incluindo um novo método de comunicação, interface de usuário da Web e funcionalidade de priorização de transporte público.

MOVA é projetado para atender a um largo espectro de situações, desde fluxos muito baixos a cruzamentos congestionados. MOVA também pode operar em uma larga faixa de tipos de interseções, desde cruzamento muito simples até cruzamentos largos e complexos, com muitos grupos semafóricos e estágios.

Em situações não congestionadas, MOVA opera em um modo que minimiza o atraso. Em situações congestionadas, o sistema adota um procedimento para maximizar a capacidade.

MOVA também pode ser usado quando houver 2 ou mais cruzamentos muito próximos, podendo o conjunto ser considerado como isolado, há recursos de o MOVA operar esses cruzamentos como sendo um só cruzamento. As rotatórias são um bom exemplo dessa situação.

⁹ *Gap out* é a situação em que não houve demanda detectada em um período de tempo igual à extensão de verde.

Como o MOVA é um padrão do Reino Unido, qualquer controlador fabricado naquele país pode ter essa funcionalidade, podendo haver interoperabilidade entre esses controladores.

b) *Smart Intersection*

Trata-se de um sistema de controle local da empresa SWARCO.

De acordo com SWARCO (2020) [14] e o Manual do Usuário do *Smart Intersection*:

O *Smart Intersection* é uma solução de controle de tráfego adaptativo para interseções isoladas e foi projetada para medir o nível de serviço em uma interseção individual, avaliando a duração ideal do ciclo e o *split* de forma a minimizar o atraso, o número de paradas, a poluição e o consumo de combustível.

O objetivo do *Smart Intersection* é manter a alocação ideal do tempo de verde e reduzir ou aumentar o tempo de duração do ciclo de acordo com a demanda de tráfego.

O algoritmo de avaliação baseia-se nas informações recebidas dos detectores de tráfego localizados próximo da linha de retenção (máximo de 35 m de distância), de preferência em cada faixa de tráfego. Os detectores de 2 a 4 m de comprimento alimentam o sistema com dados como: *gap*, ocupação e volumes.

O módulo *Smart Intersection*, juntamente com o módulo *Smart Corridor*, faz parte do pacote *Smart Green*, o qual é um sistema com controle distribuído.

c) *EPICS*

O *Epics* é um sistema de controle local da empresa alemã PTV.

Segundo o site da PTV, <https://www.myptv.com/en/mobility-software/adaptive-traffic-control-system-ptv-balance-epics> (Acesso em 06/07/2022):

O *EPICS* foi desenvolvido especificamente para interseções isoladas. O processo baseado em modelo observa as condições locais em tempo real e calcula várias opções de controle a cada segundo. O sistema adaptativo usa seu modelo interno para avaliar as diferentes opções antes de aplicar a mais adequada. Nesse processo, leva em consideração todos os modos de transporte – desde o transporte motorizado individual e pedestres até a priorização do transporte público.

O módulo *EPICS* em conjunto com *Balance* forma um sistema de controle distribuído.

2.2.1.3 Controle semafórico distribuído

O objetivo do controle distribuído é reduzir o processamento computacional em relação ao processamento de uma abordagem centralizada, uma vez que o problema de otimização é decomposto em pequenos subproblemas locais, cujo tamanho não cresce com o tamanho da rede. Cada nível é responsável por determinadas funções.

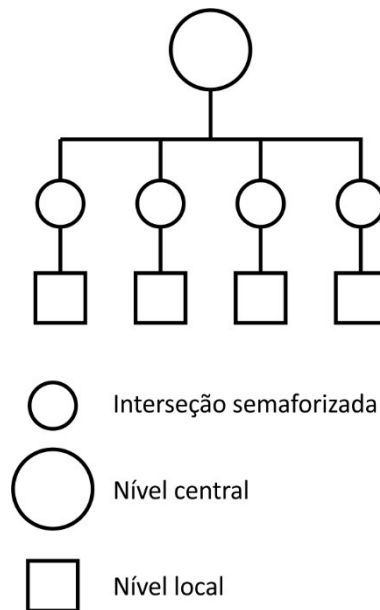


Figura 3 – Controle distribuído ou hierárquico

Uma vantagem do controle distribuído é a possibilidade de haver módulos separados e independentes: um para controle local e outro para controle centralizado, que, no entanto, podem funcionar de forma integrada. Exemplos disso são os pacotes “*Smart Intersection + Smart Corridor*” da SWARCO e “*EPICS + BALANCE*” da empresa PTV.

Assim como ocorre com o controle local, no controle distribuído, por haver inteligência embarcada no controlador, não há a possibilidade de interoperabilidade de controladores de diferentes fabricantes.

2.2.1.4 Controle semafórico descentralizado

Como resultado da descentralização, esses sistemas são escaláveis. E eles podem apresentar uma redução significativa de esforço computacional e de tempo na otimização. Mas, como o controle descentralizado busca otimizar, de forma isolada, cada região em que a rede foi decomposta, não necessariamente ele irá chegar à melhor solução global.

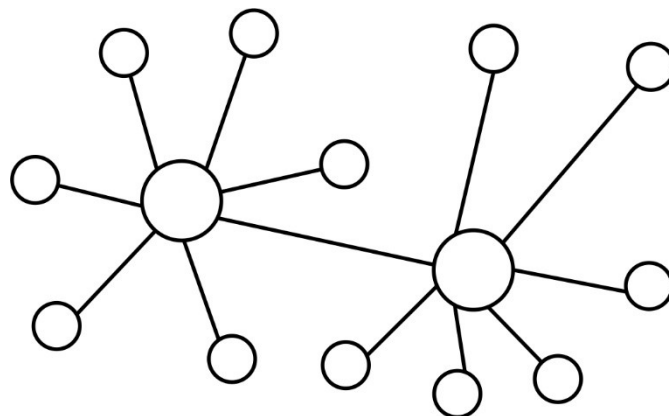


Figura 4a – Controle descentralizado

Fonte: <https://www.geeksforgeeks.org/comparison-centralized-decentralized-and-distributed-systems/>

(Acesso em 11/07/2022)

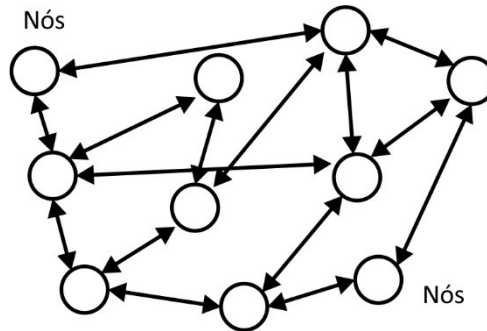


Figura 4b – Outra configuração de controle descentralizado

Fonte: <https://www.scaleyourapp.com/difference-between-centralized-decentralized-distributed-systems-explained/>

(Acesso em 11/07/2022)

Segundo Chow et al (2019) [12]:

O algoritmo descentralizado decompõe a rede original em um conjunto de redes menores interconectadas. O algoritmo descentralizado reduz significativamente o esforço computacional. No entanto, a economia de esforço computacional em algoritmos descentralizados tem como custo a perda de desempenho em cenários de alta demanda de tráfego.

Essa arquitetura também não permite a interoperabilidade entre controladores de diferentes fabricantes.

2.2.2 Classificação de sistemas adaptativos quanto ao momento da implementação da temporização semafórica

Quanto ao momento da implementação da temporização semafórica, os sistemas adaptativos podem ser classificados em:

- Sistemas de tempo real: a temporização semafórica é aplicada para a demanda de tráfego detectada. Esses sistemas não são baseados em planos.
- Sistemas que calculam os tempos semafóricos com base na demanda verificada no ciclo corrente e os implementa no ciclo seguinte. Esses sistemas não alcançam uma eficiência ótima, pois a demanda do ciclo corrente não é a mesma do ciclo seguinte. Os tempos semafóricos calculados com base na demanda passada podem não ser os melhores para a demanda verificada no ciclo futuro. É possível que o processo não se dê na base de ciclo-a-ciclo, mas baseado em intervalos de tempo: é verificada a demanda média no intervalo presente e a implementação da nova temporização ocorre no próximo intervalo.
- Sistemas preditivos: a temporização semafórica é aplicada para uma demanda de tráfego projetada com base na demanda de tráfego detectada no passado. Baseando-se na detecção do tráfego feita no passado recente, os sistemas preditivos fazem uma estimativa da demanda futura, com base na qual ocorre a implementação da temporização. Se a estimativa da demanda futura for feita com precisão, o sistema poderá antecipar futuras variações de tráfego, em vez de simplesmente reagir às variações, como o fazem os sistemas de tempo real. Por outro lado, se a estimativa da demanda futura resultar incorreta, o sistema não apresentará um bom desempenho.

2.2.3 Classificação de sistemas adaptativos quanto à sequência de estágios

De acordo com Fehon [15], os sistemas adaptativos podem ser classificados em:

- a) Baseados em sequência:
 - Usa o tempo de ciclo (semáforos coordenados);
 - Usa uma sequência pré-definida de estágios.
- b) Não baseados em sequência:
 - Não usa tempo de ciclo;
 - Não usa uma sequência pré-definida de estágios.

2.2.4 Classificação de sistemas adaptativos quanto à estrutura

Ainda de acordo com Fehon [15], os sistemas adaptativos podem ser classificados em:

- a) Sistema completo. Por exemplo: SCATS, SCOOT, LA ATCS.
- b) Módulo dentro de sistema proprietário. Por exemplo: OPAC.
- c) Módulo externo ao sistema proprietário. Por exemplo: ACS LITE, InSync, Rhodes.

Nota:

Embora o Fehon [15] tenha citado o SCOOT como um sistema completo, na verdade, o SCOOT é apenas um “kernel” que roda dentro de um sistema (interface homem-máquina) proprietário.

2.2.5 Classificação de sistemas adaptativos quanto à estratégia de otimização

De acordo com Papageorgiou et al (2003) [3], os sistemas adaptativos podem ser classificados em:

- a) Sistemas baseados na otimização de *split*, *offset* e tempo de ciclo, com alterações incrementais. Por exemplo: SCOOT.
- b) Sistemas não baseados em otimização de *split*, *offset* e tempo de ciclo. Por exemplo: OPAC, PRODYN, CRONOS, RHODES.
- c) Sistemas baseados na abordagem “*store-and-forward*”. Por exemplo: TUC

Nota:

A estratégia “*store-and-forward*” é uma estratégia usada na otimização do *split* e será detalhada mais adiante na Capítulo 8.

3. SISTEMAS DE DETECÇÃO

Os sistemas adaptativos precisam de dados de demanda de tráfego. Os dados de demanda de tráfego são obtidos a partir de sensores ou detectores. Esses sensores medem grandezas como volume, ocupação, *headway*¹⁰ etc. Os sensores são como os “olhos” do sistema.

Os detectores podem ser classificados em dois tipos: detectores eulerianos (*eulerian detectors*) e detectores lagrangeanos (*lagrangian detectors*).

O termo “lagrangeano” refere-se especificamente ao fato de que as medições são coletadas de sensores que se movem ao longo de uma trajetória, em vez de coleta dos dados em um local fixo. Tipicamente, o GPS de veículos e *smartphones* podem ser considerados como detectores lagrangeanos. Por outro lado, sensores “eulerianos” se referem a sensores fixos que coletam medições em pontos fixos predefinidos.

Infraestrutura clássica de monitoramento de tráfego conta com sensores eulerianos, por exemplo, detectores de laço indutivo, laços virtuais em câmeras de vídeo, *transponders* RFID, radares, magnetômetros etc. (Work et al, 2009) [16].

Os sensores lagrangeanos fornecem dados de espaço x tempo, que podem ser convertidos para representar a trajetória de veículos.

A Figura 5 mostra a trajetória de um veículo que chega no período de vermelho em uma aproximação semaforizada.

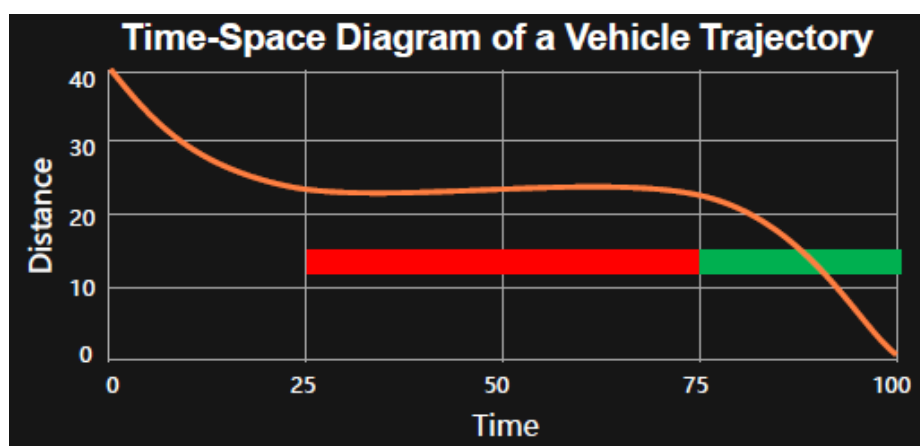


Figura 5 – Trajetória de um veículo em aproximação semaforizada

Fonte: Liu (2019) [17]

Os sensores eulerianos detectam todos os veículos que passam pelo ponto de detecção. Por outro lado, como hoje em dia nem todos os veículos tem GPS ou *smartphone*, então nem todos os veículos estão conectados ao sistema adaptativo. Assim, os veículos usados como sensores lagrangeanos representam apenas uma amostra do fluxo.

¹⁰ *Headway* é o Intervalo de tempo, em segundos, entre a passagem de dois veículos sucessivos pelo mesmo ponto de uma faixa da via, considerando-se a mesma parte da carroceria em ambos os veículos (por exemplo, o eixo dianteiro, ou o para-choques dianteiro).

Fonte: <http://www.sinaldetransito.com.br/glossario.php?ID=363> (Acesso em 27/06/2022).

A Figura 6 mostra a trajetória de 10 veículos numa aproximação semaforizada. Desses 10 veículos, apenas 2 estão conectados ao sistema adaptativo (cujas trajetórias estão representadas pelas linhas laranja). Assim, embora o volume real seja de 10 veículos, na posição s_0 o sistema só “enxerga” 2, representados pelos círculos azuis (amostragem espacial).

Os detectores eulerianos fazem a amostragem no tempo (amostragem temporal). Por exemplo, na Figura 6, no instante t_0 o sistema só “enxerga” 1 veículo (representado pelo círculo verde) no detector fixo (detector euleriano), embora vários veículos tenham passado nesse instante (em diferentes posições). Entretanto, ao longo do tempo, o detector fixo consegue detectar todos os veículos que passam pelo ponto de detecção.

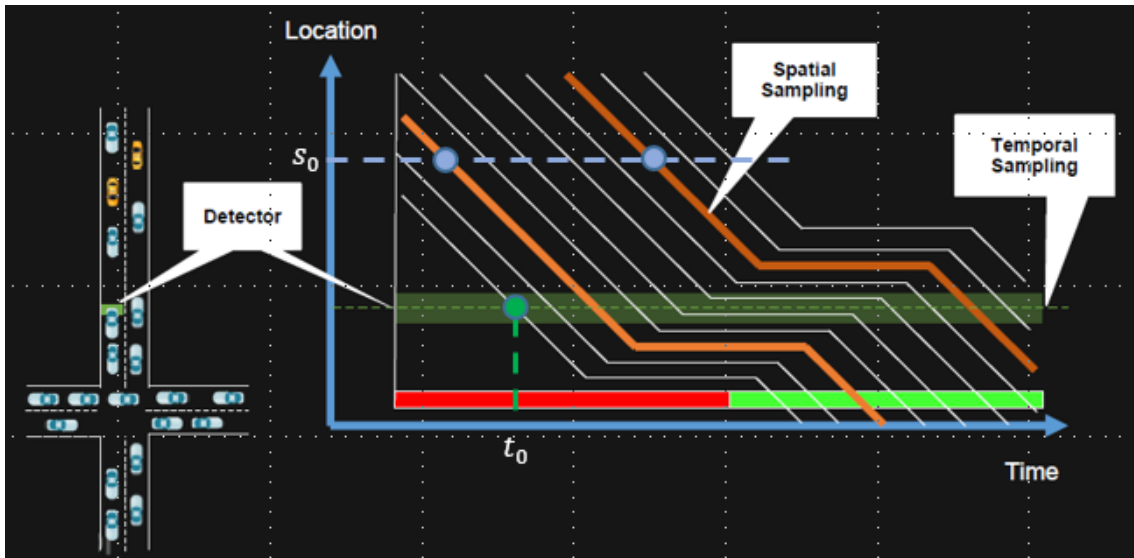


Figura 6 – Amostragem temporal (detector euleriano) e amostragem espacial (detector lagrangeano)
 Fonte: DASCOS (2019) [18]

A porcentagem de veículos conectados ao sistema em relação ao volume total é denominada “taxa de penetração” (*penetration rate*). Obviamente, quanto maior for a taxa de penetração, isto é, quanto mais veículos estiverem conectados ao sistema, mais confiável será o resultado da otimização semafórica.

A Tabela 5 mostra a diferença entre detectores fixos (eulerianos) e móveis (lagrangeanos).

Tabela 5 – Diferença entre detectores fixos (eulerianos) e móveis (lagrangeanos)

	Detectores fixos (eulerianos)	Detectores móveis (lagrangeanos)
Alcance de detecção	Todos os veículos em um ponto fixo	Veículos conectados ao sistema em toda a cidade
Medições	Volume, ocupação, 'headway'	Tempo de percurso, velocidade, número de paradas, atraso, fila

A Figura 7 adiante ilustra as grandezas que os sensores móveis podem medir. As linhas pretas mais escuras representam as trajetórias dos veículos conectados ao sistema. A linha vermelha mais fina representa a fila formada durante o tempo de vermelho. A linha verde mais fina representa o escoamento ou o desmanche da fila durante o tempo de verde. A tangente da trajetória do veículo em movimento é a velocidade (*speed*). Pela Figura 7 também é possível contar o número de paradas, o comprimento máximo da fila e o atraso.

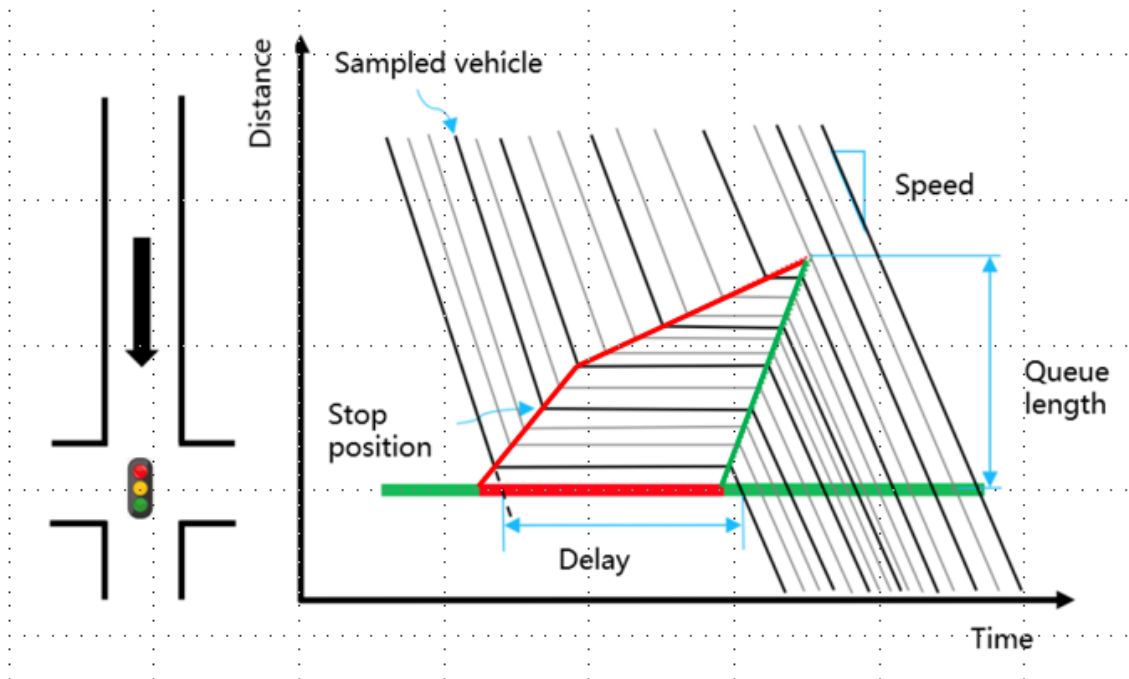


Figura 7 – Grandezas medidas com base na trajetória dos veículos conectados
 Fonte: DASCOS (2019) [18]

Entretanto, todas essas grandezas não podem ser medidas diretamente, pois os veículos conectados são apenas uma amostra do volume total. O desafio é estimar essas grandezas com base na amostra representada pelos veículos conectados.

A grande desvantagem dos sensores eulerianos (fixos) é que exigem instalação de infraestrutura física, com altos custos de implantação e manutenção. Muitas vezes, as interseções acabam operando em tempo fixo por causa da dificuldade da manutenção dos detectores.

Em termos funcionais, a desvantagem de detectores fixos é que o sistema não sabe o que acontece antes e depois do detector. Por exemplo, se houver congestionamento, o sistema não tem como saber a extensão da fila antes do detector. Analogamente, o sistema não sabe o que acontece depois que o veículo tiver passado pelo detector (ele pode ter estacionado ou ter sido retardado por algum evento na via).

Segundo Wei et al (2020) [19]:

Os sensores fixos são ativados apenas quando os veículos passam por eles; assim, eles só podem fornecer informações parciais (dão informação apenas quando o veículo passa pelo sensor). Como resultado, em alguns cenários o sistema não pode perceber e reagir com exatidão aos padrões de tráfego em tempo real. Os engenheiros precisam intervir manualmente no sistema em determinados cenários de condições de tráfego.

Atualmente, existem informações muito mais ricas que os sensores fixos que podem ser coletadas de várias fontes e que podem servir como entrada para fins de controle de semáforos. Por exemplo, câmeras, aplicativos de navegação (como o *Google Maps*), plataformas de compartilhamento de viagens (por exemplo, Uber) e veículos equipados com GPS podem fornecer uma descrição mais detalhada da situação do tráfego, especificamente: quantos carros estão esperando na aproximação, onde eles estão localizados e quão rapidamente eles estão se deslocando. Além disso, dados de trajetória em grande escala podem ser coletados para fornecer mais informações sobre como os veículos chegam e saem dos cruzamentos. A interpretação de dados de mobilidade já se encontra em estágio de desenvolvimento

suficiente para descrever com mais clareza a dinâmica do trânsito, o que é um recurso essencial para se melhorar o sistema de controle de tráfego.

A maioria absoluta dos sistemas adaptativos atualmente em operação no mundo ainda usa sensores eulerianos.

Entretanto, embora o uso de sensores lagrangeanos seja ainda bastante incipiente, já existem sistemas em operação usando veículos conectados. O exemplo é o sistema DASCOS (*Dynamic Area-Wide Signal Control Optimization System*) que opera em várias cidades da China utilizando veículos da empresa Didi, que é equivalente à 99 Táxis aqui no Brasil.

4. PROTOCOLOS DE COMUNICAÇÃO

4.1 Protocolo aberto

Entende-se por protocolo aberto aquele que é de livre acesso a empresas e usuários (com disponibilização de documentação suficiente para garantir a sua plena implementação) e apoiado por agências, com eventual participação ou colaboração de centros de pesquisa, fabricantes e administração pública, de modo que não é proposto por um único provedor (IDOM, 2019 – D2) [20].

Um protocolo é dito normalizado quando é padronizado por uma Norma, a qual é elaborada por um grupo formado por múltiplos agentes (como centros de pesquisa, universidades, administração pública e fabricantes) grupo este que é responsável pela manutenção e atualização do protocolo.

Protocolos desenvolvidos por uma única entidade sem a colaboração de agentes de diferentes áreas (pesquisa, administração pública e fabricantes de produtos), não garantem uma tecnologia estável. Por isso, não é desejável a adoção de protocolos que não sejam suportados por algum grupo formado por diferentes agentes.

Para garantir a interoperabilidade de equipamentos de campo é necessária a adoção de protocolos abertos. Porém, a simples adoção de protocolos abertos não garante a interoperabilidade. É necessário adotar um único protocolo, o qual será considerado padrão para a cidade ou país. Se forem adotados mais de um protocolo aberto, por exemplo protocolos A e B, não haverá interoperabilidade entre os equipamentos A e B, isto é, um equipamento de campo do sistema A não poderá ser substituído por um equipamento do sistema B e vice-versa. Essa situação irá gerar “ilhas” de reserva tecnológica na cidade, pois nas regiões onde opera o sistema A só poderão ser conectados equipamentos que operam com o protocolo A e assim por diante.

Além disso, o simples fato de o protocolo ser aberto também não garante que haja interoperabilidade de funcionalidades, pois cada fabricante pode inserir no seu sistema funcionalidades não previstas no protocolo aberto, criando uma parte proprietária dentro do protocolo para atender essas funcionalidades específicas do seu sistema.

4.2 Principais protocolos utilizados

O documento IDOM (2019 – D2) [20] faz a descrição dos seguintes protocolos:

- AENOR;
- OCIT;
- NTCIP;
- UTMC; e
- SCATS.

As informações sobre esses protocolos foram extraídas basicamente do documento IDOM (2019 – D2) [20].

A Tabela 6 apresenta as principais características dos protocolos apresentados.

Tabela 6 – Principais características dos protocolos apresentados

	AENOR	OCIT	NTCIP	UTMC	SCATS
Aberto	Sim	Sim	Sim	Sim	Não
Licença	Não	Sim	Não	Não	Sim
País	Espanha	Alemanha	EUA	Reino Unido	Austrália
Homologação	Sim	Não	Não	Não	Não
Sistemas	---	---	---	SCOOT	SCATS

4.3 Conceitos básicos

Para que haja entendimento entre os diversos equipamentos de rede é necessário que todos os dispositivos utilizem o mesmo protocolo, isto é, falem a mesma linguagem. Protocolo é uma linguagem constituída por um conjunto de regras, símbolos e convenções.



Figura 8 – Comunicação por meio de sinais de fumaça

Para uma melhor compreensão do que é protocolo, considere-se a comunicação por meio de sinais de fumaça, como na Figura 8. A comunicação por sinal de fumaça só funciona se houver uma codificação dos sinais. De forma simplificada, a codificação constitui o protocolo, o pano é o mecanismo para codificar a mensagem e a fumaça é o meio de transporte da mensagem. O protocolo é aberto se a codificação dos sinais de fumaça com os seus correspondentes significados for de acesso livre a todos, de forma que qualquer interessado pode entender os sinais recebidos ou transmitir os seus próprios sinais. O conjunto dos códigos com os seus significados constitui o dicionário do protocolo.

Com a finalidade de reduzir a complexidade, as redes são organizadas como uma pilha de camadas ou níveis. O objetivo é fazer com que cada camada ofereça determinados serviços à camada superior. A camada n de uma máquina se comunica com a camada n de outra máquina. Essa comunicação, camada a camada, é feita por meio de protocolos específicos. Assim, para que exista a comunicação da mensagem é usado um conjunto de protocolos, conforme a Figura 9.

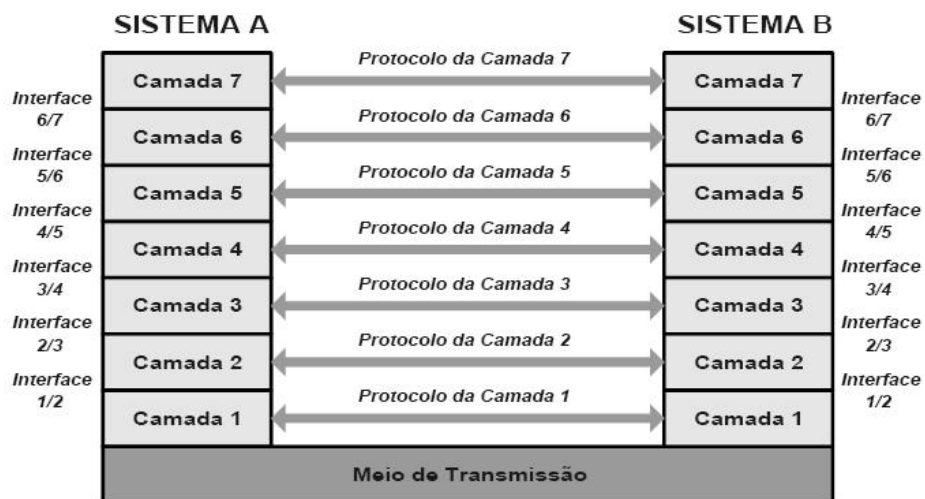


Figura 9 – Comunicação camada a camada

Fonte: https://pt.wikibooks.org/wiki/Redes_de_computadores/Protocolos_e_servi%C3%A7os_de_rede (Acesso em 30/06/2022)

No exemplo dos sinais de fumaça, pode-se estabelecer 3 camadas:

Camada	Função	Protocolos
3	Mensagem	Dicionário
2	Produção dos sinais	Pano
1	Transporte	Fumaça

4.3.1 Modelo OSI

Um exemplo de protocolos em camadas é o modelo OSI.

O Modelo OSI (*Open System Interconnection*) é um modelo de rede de computador, referência da ISO, dividido em 7 camadas, criado com objetivo de ser um padrão para protocolos de comunicação entre os mais diversos sistemas em uma rede local (*Ethernet*), garantindo a comunicação entre dois sistemas computacionais (*end-to-end*). O Modelo OSI não é uma arquitetura de redes, pois não especifica os serviços e protocolos exatos que devem ser usados em cada camada. Ele apenas informa o que cada camada deve fazer (https://pt.wikipedia.org/wiki/Modelo_OSI, acesso em 30/06/2022).

Portanto, o modelo OSI é apenas um modelo conceitual.

Camada	Função	Protocolos
7	Aplicação	HTTP, SMTP, SNMP etc.
6	Apresentação	XDR, TLS etc.
5	Sessão	NetBios etc.
4	Transporte	TCP, UDP etc.
3	Rede	IP etc.
2	Enlace	Ethernet etc.
1	Física	Modem, USB, Bluetooth etc.

Figura 10 – Modelo OSI

De acordo com Matheus (2018) [21], quando se fala das camadas do modelo OSI, costuma-se falar na ordem inversa, isto é, a camada mais alta é a camada 7, enquanto a camada 1 é a camada mais baixa, conforme a Figura 11, a seguir:



Figura 11 – As camadas do modelo OSI
Fonte: Matheus (2018) [21]

Camada 1

A camada 1 (Física) define especificações elétricas e físicas dos dispositivos. Em especial, define a relação entre um dispositivo e um meio de transmissão, tal como um cabo de cobre ou um cabo de fibra óptica.

A camada física seriam as estradas, ou seja, o caminho que os pacotes de dados percorrem para chegar ao destino.

Camada 2

A camada 2 (Enlace) detecta e, opcionalmente, corrige erros que possam acontecer no nível físico. É responsável por controlar o fluxo dos pacotes de dados (recepção, delimitação e transmissão de quadros).

Essa camada funciona como um fiscal. Ele observa se o pacote tem algum defeito em sua formatação e controla o fluxo com que os pacotes são enviados.

Nesta camada é que são definidas as tecnologias como as *VLans* ou topologias e dispositivos como os switches.

Camada 3

A camada 3 (Rede) realiza roteamento de funções. Roteadores operam nesta camada, enviando dados em toda a rede estendida e tornando a Internet possível. Este é um esquema de endereçamento lógico. O esquema de endereçamento não é hierárquico.

Fazendo-se uma analogia com os correios, é nesta camada que se verifica quem é destinatário e quem é o remetente da carta. Se existirem muitas cartas para serem enviadas, pode-se priorizar quais serão enviadas primeiro e qual é o melhor caminho para enviar.

Camada 4

A camada 4 (Transporte) é responsável por receber os dados enviados pela camada de sessão (camada 5) e segmentá-los para que sejam enviados à camada de rede, que por sua vez, transforma esses segmentos em pacotes. No receptor, a camada de Transporte realiza o processo inverso, ou seja, recebe os pacotes da camada de rede e junta os segmentos para enviar à camada de sessão.

Ainda usando a analogia dos correios, se na camada 1 tem-se as estradas e os caminhos que os dados percorrem, na camada 4 tem-se os caminhões e os carteiros.

Camada 5

A camada 5 (Sessão) permite que duas aplicações em computadores diferentes estabeleçam uma comunicação, definindo como será feita a transmissão de dados, pondo marcações nos dados que serão transmitidos. Se porventura a rede falhar, os computadores reiniciam a transmissão dos dados a partir da última marcação recebida pelo computador receptor.

Esta camada é responsável por estabelecer e encerrar a conexão.

Camada 6

A camada 6 (Apresentação) converte o formato do dado recebido pela camada de Aplicação em um formato comum a ser usado na transmissão desse dado, ou seja, um formato entendido pelo protocolo usado. Um exemplo comum é a conversão do padrão de caracteres (código de página) quando o dispositivo transmissor usa um padrão diferente do ASCII. Pode ter outros usos, como compressão de dados e criptografia.

Esta é a camada responsável por fazer a tradução dos dados para que a próxima camada os use.

Camada 7

A camada 7 (Aplicação) corresponde às aplicações (programas) no topo da camada OSI que serão utilizadas para promover uma interação entre a máquina-usuário (máquina destinatária e o usuário da aplicação).

A última camada do modelo OSI é a camada para consumir os dados. Nesta camada tem-se os programas que garantem a interação homem-máquina.

4.3.2 Estrutura do NTCIP

Para ilustrar o modelo por camadas para aplicações em ITS é apresentado a seguir a estrutura adotada pelo NTCIP, em 5 camadas (maiores detalhes sobre NTCIP estão no item 4.3.6), conforme mostra a [Figura 12](#).

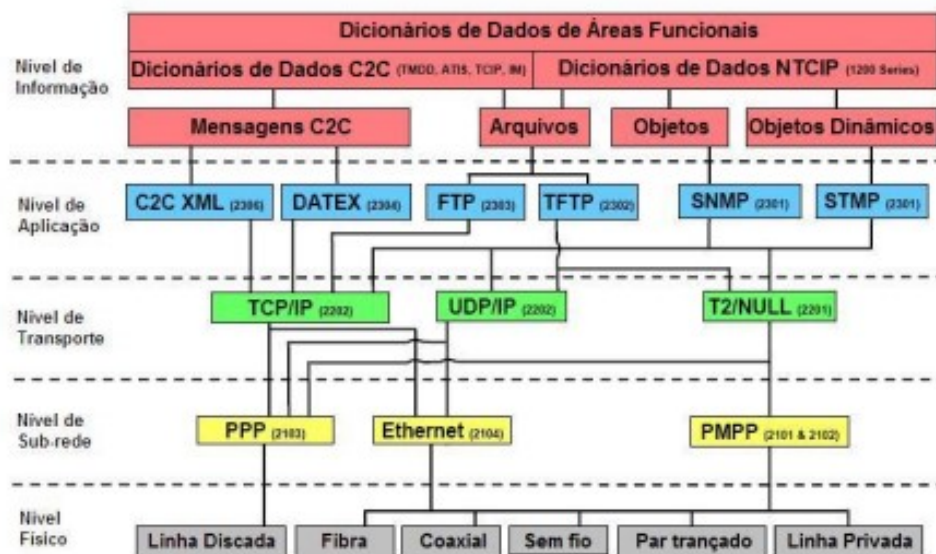


Figura 12 – Padrão NTCIP
Fonte: Kraus et al (2011) [22]

Conforme Kraus et al (2011) [22]:

A maioria dos protocolos incluídos pelos proponentes do NTCIP são padrões em redes de computadores e na internet. São exclusivos do contexto do NTCIP os dicionários de dados de áreas funcionais que definem o nível de informação, o qual não encontra similar nas camadas do modelo ISO/OSI. Tal fato deriva do caráter abstrato do modelo OSI, ao passo que o NTCIP tem domínio específico de aplicação em Sistemas Inteligentes de Transportes (ou ITS, do inglês *Intelligent Transportation Systems*), daí acarretando a necessidade de se definir dados padronizados através de dicionários.

Note que nas camadas do “Nível físico” até “Nível de aplicação” da Figura 11, os protocolos são aqueles dos computadores e da Internet. O que é exclusivo do NTCIP está na camada “Nível de informação”, onde estão os dicionários da aplicação. É nessa camada que ficam os protocolos das funcionalidades do sistema.

Na camada do “Nível de informação”, observa-se que do lado esquerdo constam os dicionários de dados C2C (*Center-to-Center*) que são os protocolos relativos às funcionalidades para a integração de diferentes sistemas de ITS (por exemplo, integração de sistemas de CFTV, PMV – Painel de Mensagem Variável, sistema de semáforos etc.). Já do lado direito dessa camada, estão os dicionários de dados NTCIP que são os protocolos C2F (*Center-to-Field*), responsáveis pela comunicação entre o sistema central e o equipamento de campo. São esses protocolos que representam as funcionalidades do sistema.

Um “*stack*” é um subconjunto da estrutura em camadas, um “caminho” através das camadas.

A Figura 13 mostra um exemplo de “*stack*”, indicado pela linha vermelha.

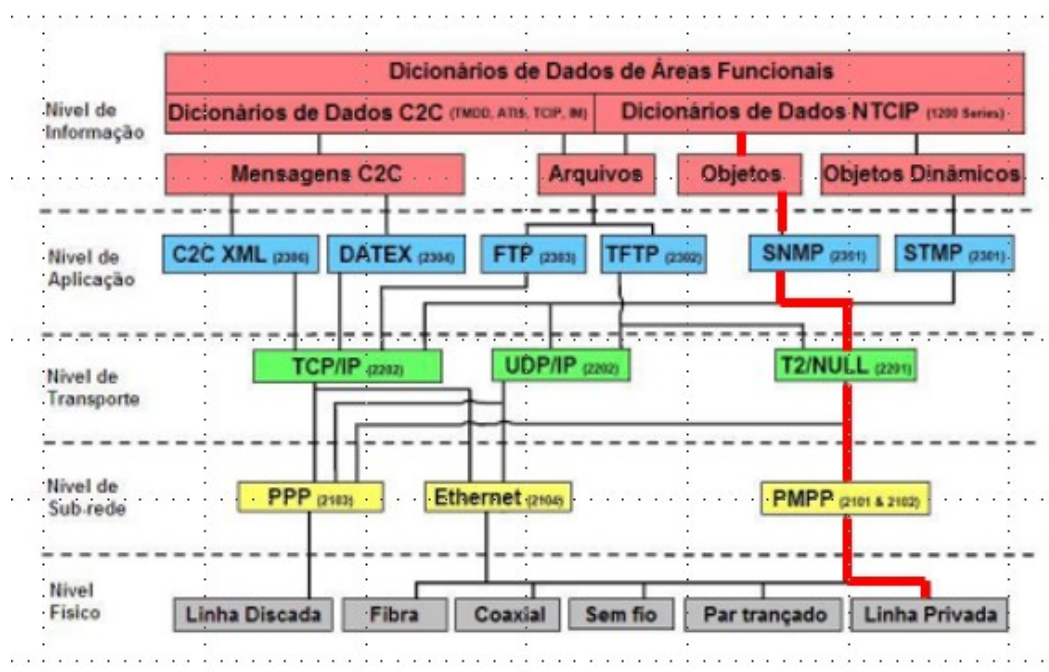


Figura 13 – Exemplo de “stack” na estrutura do NTCIP

Fonte: Adaptado de Kraus et al (2011) [22]

Dessa forma, para que haja total interoperabilidade é necessária a padronização de um “stack”.

4.3.3 MIB (Management Information Base)

MIB é um bloco de dados, que é um tipo de banco de dados contendo informações hierárquicas, estruturadas na forma de uma árvore, de todos os dispositivos gerenciados em uma rede de comunicações. Um protocolo pode ser desenvolvido com base em MIBs.

O MIB descreve a organização de um banco de dados que será criado na área de memória do dispositivo onde está instalado. Os bancos de dados MIB serão usados para armazenar informações, que por sua vez serão usadas para controlar os equipamentos de um sistema.

O MIB é um documento de texto que pode ser lido por um humano e "compilado" por um computador. "Compilar" significa converter de forma legível para o formato especial usado por um computador.

4.3.4 AENOR (Asociación Española de Normalización y Certificación)

O protocolo AENOR é um conjunto de protocolos. O protocolo para semáforos é o UNE 135401-4:2003 IN.

O protocolo foi desenvolvido com a colaboração de agentes da Administração Pública e dos fabricantes espanhóis: ETRA, SICE, KAPSCH (antigo TELVENT), INDRA e ACISA.

4.3.5 OCIT (Open Communication Interface for Road Traffic Control System)

O protocolo é uma marca registrada sendo que, para a sua implementação, é necessária a aquisição de licença. O protocolo foi desenvolvido por um grupo de empresas alemãs.

Segundo o Relatório IDOM (2019 – D2) [20]:

O protocolo foi desenvolvido por iniciativa de várias cidades alemãs e corresponde a uma marca registrada pelas empresas:

- Dambach-Werke GmbH¹¹ (pertencente ao grupo SWARCO)
- Siemens
- Swarco Traffic Systems GmbH (antiga Signalbau Huber)
- AVT STOYE GmbH
- Stührengberg GmbH

Além disso, o grupo trabalhou como parceiro com a empresa Verkehrs-Systeme AG na padronização da interface central.

O protocolo define em nível funcional o tratamento (realizado localmente pelo controlador) dos dados obtidos e dos elementos que compõem a interseção, de modo que, a partir do Centro de Controle, será configurada a regulação a ser desenvolvida pelo controlador. De acordo com este operativo, uma vez configurado, o regulador opera automaticamente.

A definição funcional do protocolo facilita a integração de produtos de diferentes fabricantes no mesmo Centro de Controle e com diferentes aplicativos.

A presença internacional dos fabricantes que desenvolveram o protocolo levou à extensão do protocolo para fora da Alemanha e dos países da Europa Central, com grande dependência econômica da Alemanha (Áustria, Suíça, Polônia, etc.), embora deva ser notado que:

- O protocolo é uma marca registrada, sendo possível a aquisição de uma licença para a sua implementação por um fabricante fora do grupo;
- Não é necessária a aprovação de controladores no protocolo OCIT;
- A documentação disponível está no idioma alemão.

O protocolo OCIT é um protocolo autocontido, ou seja, a implementação do protocolo é completamente limitada na documentação que o define, de modo que a interoperabilidade entre fabricantes está associada à conformidade com o protocolo, ao menos nas funções estabelecidas por ele.

Apesar de o Relatório IDOM (2019 – D2) [20] mencionar que a documentação está no idioma alemão, é possível baixar documentos em inglês do site <https://www.ocit.org/en/ocit/downloads/> (acesso em 17/10/2022).

Segundo o documento OCIT Outstations – Introduction (2018) [23]:

As interfaces OCIT formam a base para uma arquitetura de sistema aberta. Elas estão focadas em conexões padronizadas entre componentes centralizados e descentralizados em todo o mundo, como subsistemas, ferramentas e dispositivos de campo. Com o uso da tecnologia da Internet, elas possibilitam a construção de sistemas de gerenciamento de tráfego e redes sistêmicas compostas por dispositivos de campo e centros de controle.

¹¹ GmbH (Gesellschaft mit beschränkter Haftung) é o termo alemão para uma empresa limitada e é comum na Alemanha e em outros países na Europa.

Até agora, as seguintes áreas de interface foram definidas no sistema geral de controle de tráfego:

- OCIT *Center to Center* (OCIT C) são interfaces padronizadas entre componentes e sistemas centrais e cobrem praticamente todas as áreas de interface de gerenciamento de tráfego.
- OCIT-*Outstations* (OCIT-O) são interfaces padronizadas entre dispositivos centrais e dispositivos de campo.
- Os perfis OCIT-O são definições para a tecnologia de transmissão de dados.
- OCIT-O *Car* é um conjunto de interfaces padronizadas entre a unidade que fica na via e Centro de Controle.
- OCIT-LED é uma interface elétrica entre o controlador semafórico e os grupos focais a LED com tecnologia de 40 V.

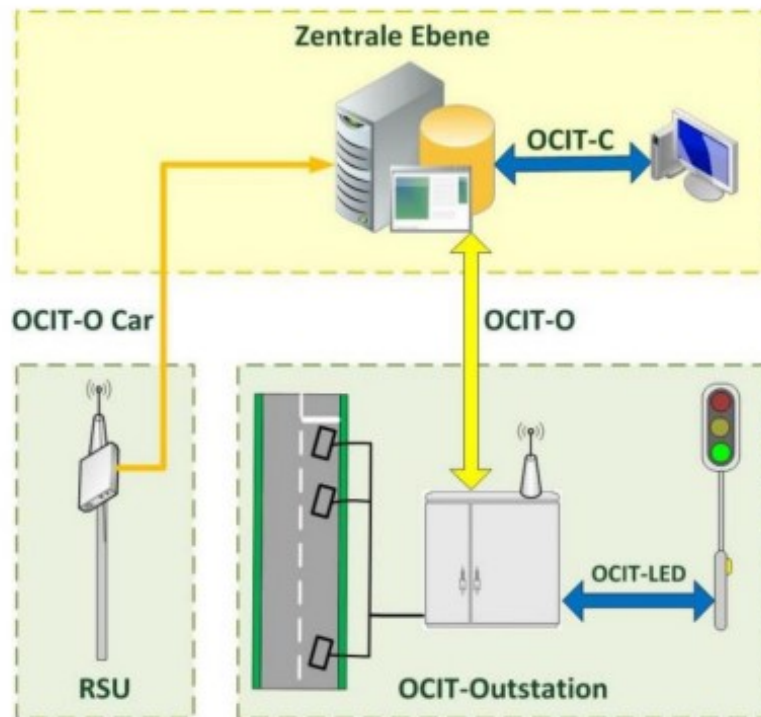


Figura 14 – Interfaces do sistema OCIT

Fonte: Figura extraída de *OCIT Outstations – Introduction* (2018) [23]

Os sistemas OCIT com a interface OCIT *Outstations* transmitem comandos e dados somente quando ocorrem determinados eventos.

A comunicação de dados usa o padrão ISO "Modelo de Camadas OSI".

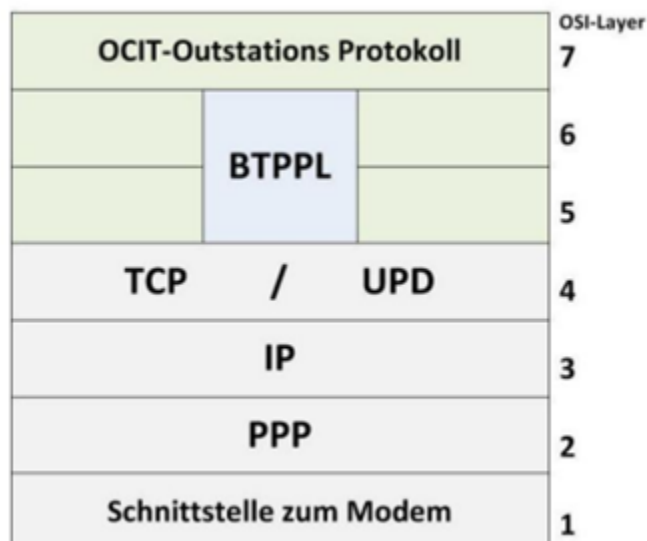


Figura 15 – Modelo de Camada OSI do protocolo OCIT
 Fonte: Figura extraída de OCIT Outstations – Introduction (2018) [23]

BTPPL apresenta menos sobrecarga de dados e, portanto, tempos de transmissão significativamente mais curtos. É por isso que as antigas conexões de cabo muitas vezes ainda em vigor, com sua qualidade de transmissão relativamente baixa, podem ser usadas.

De acordo com o documento OCIT-O Version 3.0 – Mirror (2018) [24]

Os controladores semafóricos com protocolo OCIT são projetados especificamente para uso em sistemas com estrutura descentralizada. Eles controlam modos complexos de atuação veicular e podem adquirir e processar dados de medições obtidas de sensores (“controladores inteligentes”).

- Referências de tempo

A central de controle disponibiliza o serviço de padronização de tempo NTP versão 4 (RFC 591 1305), que pode ser utilizado para sincronizar o controlador OCIT com a central de controle. O processo de sincronização compensa os tempos de transmissão na rede.

O OCIT prevê as seguintes referências de tempo para efeitos de coordenação:

- Referência: 1.1. 0:00:00
Tempo local, ano corrente

Nota: A referência “1.1” significa 1° de janeiro.

- Referência: 1980-01-01 0:00:00
Tempo local

- Referência: 0:00:00
Tempo local do dia corrente

- Referência: 1970-01-01 0:00:00
Tempo Universal Coordenado¹² (Universal Time Coordinated – UTC)

Para que haja a interoperabilidade de controladores é necessário que se padronize a referência de tempo.

- Número de anéis (“*Partial Intersections*”)

Os controladores OCIT podem comportar até 4 anéis.

- Lógica de atuação veicular

Os controladores OCIT podem ser equipados com diversas lógicas de atuação veicular. O tipo de lógica deve ser especificado explicitamente.

- Estágio

Conforme OCIT Outstations Version 3.0 – Glossary (2018) [25]:

Estágio é uma parte de um plano semafórico no qual um determinado status de sinais luminosos permanece inalterado (durante o início do estágio, as transições de estágio ainda podem estar ocorrendo).

Essa definição de estágio no Glossário parece indicar que no OCIT o entreverdes é alocado no início do estágio.

- Atraso de grupo (“*phase delay*”)

Conforme o documento OCIT Outstation (2012) [26]:

O atraso de grupo é designado no OCIT como “*offset*” e é classificado nos seguintes tipos:

	Tipo	Nome
1	Início para início	BeginBegin
2	Fim para fim	EndEnd
3	Início para fim	BeginEnd

Não é obrigatório que um controlador OCIT contemple todos os tipos listados na tabela acima.

O operador pode impor comandos sobre o atraso de grupo conforme a [Tabela 7](#), a seguir:

¹² O Tempo Universal Coordenado, também conhecido como tempo civil, é o fuso horário de referência a partir do qual se calculam todas as outras zonas horárias do mundo.

Tabela 7 – Comandos de operador (atraso de grupo)

	Função	Protocolos
1	Maior ou igual (o valor do grupo semafórico secundário é maior ou igual ao valor do grupo semafórico principal)	ge
2	Menor ou igual (o valor do grupo semafórico secundário é menor ou igual ao valor do grupo semafórico principal)	le
3	Igual (o valor do grupo semafórico secundário é igual ao valor do grupo semafórico principal)	eq

Nota: Aqui, “slave signal group” e “master signal group” foram traduzidos para “grupo semafórico secundário” e “grupo semafórico principal”, respectivamente.

Valem aqui as regras de sinal. Por exemplo: $-6 \leq -5$ (-6 le -5).

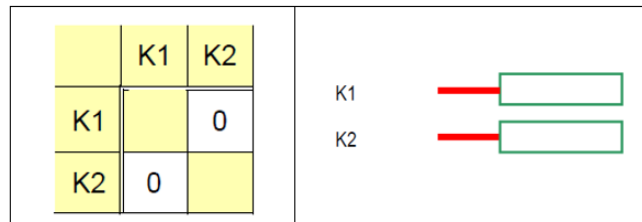
Os atrasos de grupo podem ser fixos ou variáveis.

Os atrasos de grupo fixos são aqueles com o comando “eq”, enquanto os atrasos de grupo variáveis são aqueles correspondentes a “ge” e “le”.

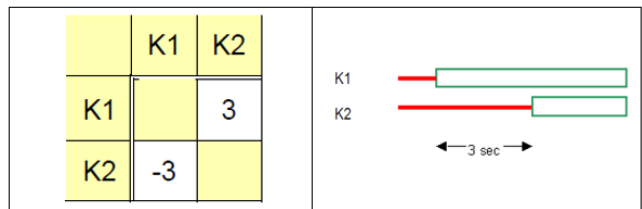
Atrasos de grupo fixos

a) Atrasos de grupo fixos do tipo (BeginBegin)

- K1 e K2 abrem juntos (sem atraso)

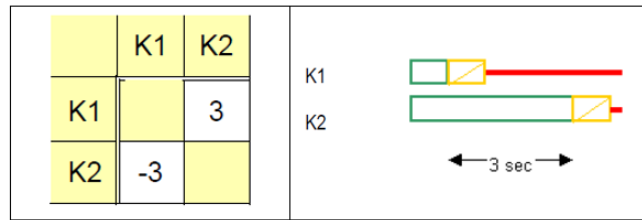


- K2 abre 3 segundos depois de K1



b) Atrasos de grupo fixos do tipo (EndEnd)

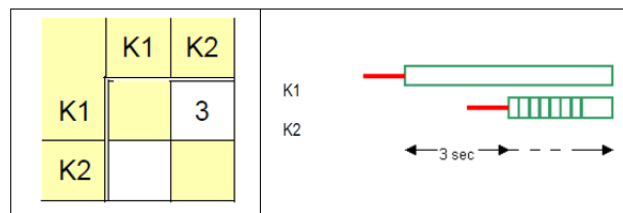
- K2 fecha 3 segundos depois de K1



Atrasos de grupo variáveis

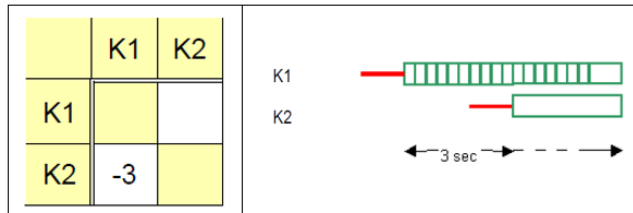
a) Atrasos de grupo variáveis do tipo (BeginBegin)

- K2 abre 3 segundos ou mais depois de K1



Aqui, K1 é o grupo semafórico principal e K2 é o secundário.

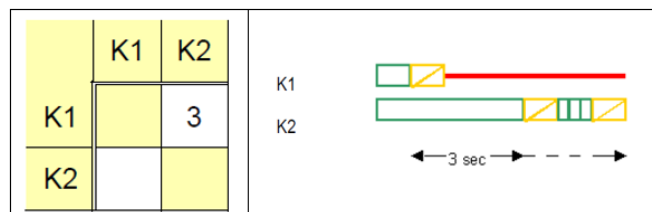
- K1 abre 3 segundos ou menos antes de K2 (possivelmente até depois de K2)



Aqui também, K1 é o grupo semafórico principal e K2 é o secundário.

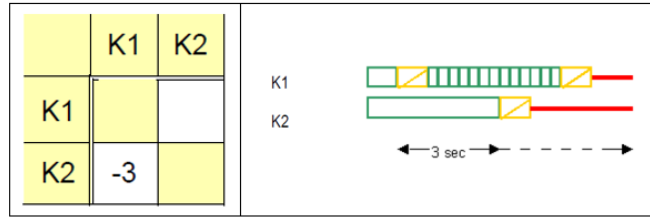
b) Atrasos de grupo variáveis do tipo (EndEnd)

- K2 fecha 3 segundos ou mais depois de K1



Da mesma forma que no caso anterior, K1 é o grupo semafórico principal e K2 é o secundário.

- K2 fecha 3 segundos ou menos depois que K1 (podendo até fechar antes de K1)



Diferente dos anteriores, aqui, K2 é o grupo semafórico principal e K1 é o secundário.

O cumprimento dos atrasos de grupo tem menor prioridade do que o cumprimento dos tempos de entreverdes e dos tempos de segurança. Caso se verifique da análise dos atrasos de grupo que eles não podem ser mantidos, o processo de transição de estágios é adiado em conformidade.

O fato de se prever a possibilidade de haver atrasos de grupo variáveis permite supor que a estratégia de controle adotada seja por grupo semafórico e não por estágio, de forma a permitir a alocação de tempos de verde por grupo (e não por estágio).

Os atrasos de grupo fixo e variável (comandos de operador) da Tabela 7, indicam que a adoção do protocolo OCIT implica em que o controlador deve estar preparado para as funções previstas no protocolo, diferentemente do protocolo UTMC, onde o controlador é apenas um acionador das luzes do semáforo. O uso do protocolo UTMC independe da lógica do controlador. Já a adoção do protocolo OCIT obriga o uso de controladores com lógica e funcionalidades compatíveis com o protocolo.

- Comunicação V2X

O protocolo OCIT, na sua versão 3.0, já prevê a comunicação V2X, conforme a Figura 16, extraída do prospecto OCIT Goes Cooperative [27]:

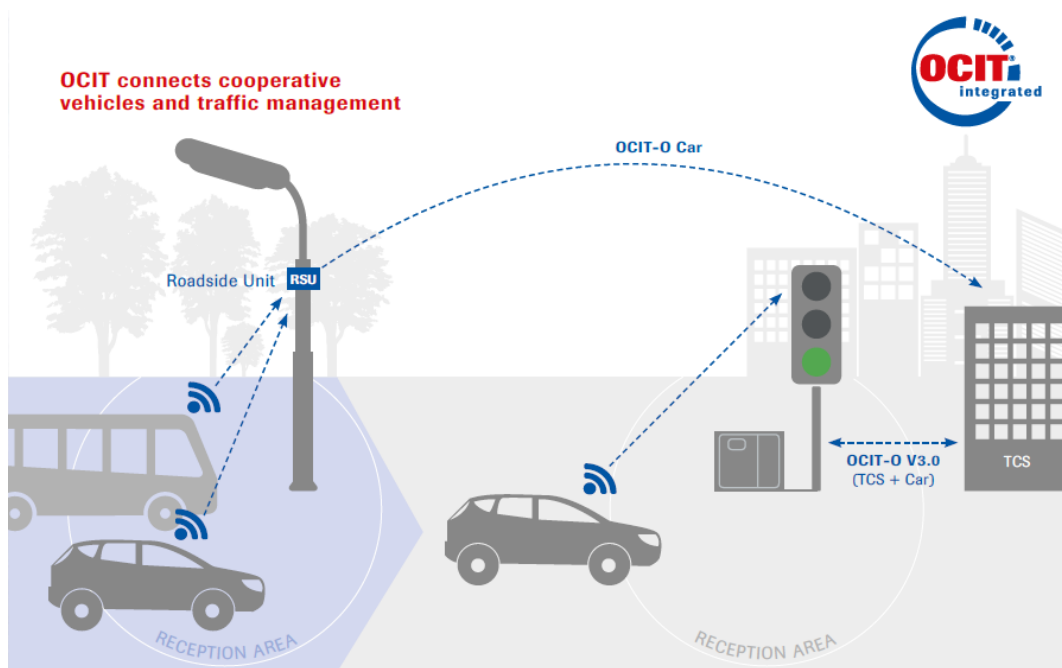


Figura 16 – Comunicação V2X no OCIT
 Fonte: Figura extraída do prospecto OCIT Goes Cooperative (2018) [27]

Da análise disponível, conclui-se que o protocolo OCIT prevê várias funcionalidades que não são todas obrigatórias. Entretanto, se o controlador for conectado a um sistema adaptativo, o controlador deve estar devidamente preparado para as funcionalidades do sistema. Por exemplo, quando o sistema adaptativo prever determinada lógica de atuação, ou usar o recurso de atrasos de grupo variáveis, a lógica do controlador deve permitir essas funcionalidades.

4.3.6 NTCIP (*National Transportation Communications for ITS Protocol*)

O NTCIP é um conjunto de protocolos. O protocolo específico para semáforos está padronizado pelo documento NTCIP 1202 – *Object Definitions for Actuated Traffic Signal Controller Units*. Note que se trata de um protocolo para semáforos atuados. Devido a essa limitação, alguns centros urbanos dos Estados Unidos optaram por adotar protocolos específicos, desenvolvidos a partir do NTCIP, como New York e Chicago.

O protocolo foi desenvolvido pelo grupo formado por NEMA, AASHTO, ITE e USDOT.

4.3.7 UTMC (*Urban Traffic Management and Control*)

O UTMC é um conjunto de protocolos de ITS. O protocolo para semáforos foi desenvolvido especificamente para atender a comunicação do SCOOT.

O protocolo específico para semáforos está normalizado pelos documentos UTMC-TS004.006:2010 *UTMC Objects Registry UM/008 – Full UTC MIB* e UTMC-TS004.0064bis:2017 *UTMC Objects Registry – Traffic Signal Interface*.

O regulamento do UTMC prevê que os fabricantes possam publicar MIBs específicos do seu sistema na documentação oficial dos regulamentos que são públicos. Isso pode ser uma vantagem em relação aos outros protocolos.

4.3.8 SCATS (*Sidney Coordinated Adaptive Traffic System*)

O SCATS não é apenas um protocolo, mas a própria plataforma. Assim, a plataforma e o protocolo se fundem num único sistema¹³.

¹³ O sistema INFLUUNT apresenta a mesma ideia do SCATS: a plataforma e o protocolo são fundidos num único sistema. Para maiores detalhes sobre INFLUUNT ver a Nota Técnica 273 – <http://www.cetsp.com.br/media/1262384/nt-273.pdf> (Acesso em 04/07/2022).

5. SCOOT: *Split Cycle Offset Optimization Technique*

O SCOOT foi desenvolvido pelo *Transport Research Laboratory* (TRL) do Reino Unido em 1979.

É um dos sistemas adaptativos mais bem estabelecidos e consolidados do mundo.

Segundo Aavani et al (2017) [28], SCOOT provou ser líder mundial em Controle de Tráfego Urbano e normalmente reduz o atraso em áreas urbanas em média 20%.

Nota: Em outro trecho do mesmo artigo é dito que os primeiros resultados mostraram que o SCOOT alcançou uma economia média no atraso de cerca de 12% quando comparado com planos de tempo fixo. No site do TRL – *Software* [29], é informado uma redução do atraso de 15%.

Considerando que o nível de maturidade de um sistema está relacionado com sua escala de implantação e com a experiência adquirida, refletida na evolução do sistema por meio de desenvolvimento e incorporação contínua de melhorias e de novas funcionalidades, pode-se afirmar que o SCOOT é um sistema de grande maturidade, pois:

a) De acordo com o site *Yunex Traffic* [30]:

SCOOT é o sistema adaptativo mais comprovado, mais amplamente implantado e mais eficaz do mundo. Isso é demonstrado pela implantação bem-sucedida de mais de 125 sistemas SCOOT, abrangendo mais de 7.000 interseções em um período de 30 anos.

b) Conforme o site do TRL [29], o SCOOT está implantado em mais de 350 cidades ao redor do mundo.

c) Conforme o artigo de Souza et al (2017) [4], SCATS e SCOOT, desenvolvidos há mais de 30 anos, são os sistemas adaptativos dominantes nos Estados Unidos, sendo as opções mais utilizadas em aplicações com mais de 50 semáforos.

d) O SCOOT está implantado em grandes centros urbanos como Londres, Beijing, Toronto, Seattle, Minneapolis, São Paulo, Santiago do Chile e Manchester. Também está implantado em cidades diferentes entre si, quanto à configuração da malha viária e características de tráfego.

e) SCOOT apresenta uma longa lista de versões, com a incorporação de melhorias e de novas funcionalidades. A primeira versão comercial, lançada no início de 1980, foi 2.2. Desde então, tem havido um programa regular de desenvolvimento, introduzindo melhorias operacionais e novas funcionalidades nos algoritmos, conforme mostram a [Tabela 8](#) e a [Tabela 9](#), demonstrando a evolução do sistema.

SCOOT é um sistema bem documentado. Por exemplo, o *SCOOT Traffic Handbook* é constituído por 41 volumes, explicando e detalhando o funcionamento do sistema, além do *SCOOT User Guide*. Note que não se trata de manuais de configuração e de operação do UTC, que constituem uma documentação à parte.

Além disso, SCOOT é um dos sistemas com mais artigos, trabalhos e documentos disponíveis na Internet, como atestou anteriormente a [Tabela 3](#), com 56,1% de todo o material pesquisado neste trabalho, que descrevem, analisam ou mencionam o sistema.

SCOOT é um sistema adaptativo de tempo real que faz uso da arquitetura física totalmente centralizada, bem como um algoritmo de temporização centralizado. Entretanto, funcionalmente o SCOOT é um sistema descentralizado, uma vez que a otimização do *split* usa tão somente os dados da própria interseção.

Se o UTMC for adotado como o protocolo padrão, então é possível haver a interoperabilidade entre controladores de diferentes fabricantes. Entretanto, devido à sua otimização em tempo real (por exemplo, a decisão do *split* é feita no próprio ciclo, 5 segundos antes do término do estágio – ver o item 5.4, adiante), bem como à sua estrutura de comunicação segundo a segundo, SCOOT requer uma comunicação com baixa latência.

Uma particularidade da comunicação segundo a segundo do SCOOT e do protocolo UTMC são os “Force bits” F1, F2, ... e os correspondentes bits de resposta G1, G2, A Central manda para o controlador segundo a segundo o bit F1, por exemplo, significando para o controlador ficar no Estágio 1. Enquanto o controlador estiver no Estágio 1, responde à Central, também segundo a segundo, o bit G1, significando “estou no Estágio 1”. Quando mudar para o Estágio 2, a Central deixa de mandar o bit F1 e manda o bit F2. O controlador, ao receber o bit F2 sabe que tem que implementar o entreverdes do Estágio 1 para o Estágio 2, respondendo, porém, o bit G2 para a Central. Devido a esse protocolo, no SCOOT o entreverdes tem que ser alocado no início do estágio, isto é, o estágio começa com o entreverdes, pois, a Central, ao receber o bit G2 sabe que o controlador está no Estágio 2. Caso contrário, o fato de a Central ao mandar o bit F2 e receber do controlador o bit G1 provocaria uma inconsistência. Na realidade, a alocação do entreverdes no início ou no final do estágio é arbitrária, sendo meramente uma convenção. É mais usual e natural considerar o entreverdes no final do estágio (afinal, o entreverdes faz parte da transição de um estágio para o seguinte¹⁴). A alocação do entreverdes no início do estágio provoca dificuldade adicional com a defasagem. Normalmente, o instante de referência para a defasagem é o instante da abertura do verde do estágio principal. Se o estágio começar com o verde fica mais fácil para visualizar a defasagem do que se o estágio começar com o entreverdes.

O SCOOT não é um sistema completo em si mesmo por se tratar apenas de um “kernel” (núcleo), em cima do qual outras empresas podem desenvolver o “knitting software” que conecta o SCOOT ao controlador de campo e provê a interface homem-máquina específica da empresa. A interface homem-máquina é normalmente denominada de sistema UTC (*Urban Traffic Control*).

O sistema fornece relatórios com os valores dos parâmetros internos do sistema, permitindo ao usuário fazer análise e diagnóstico de eventuais problemas de otimização em um local específico (Mensagens SCOOT).

SCOOT tem uma estrutura básica semelhante à do TRANSYT (*Traffic Network Study Tool*), que é um *software off-line* de otimização semafórica, desenvolvido também pelo TRL.

Entretanto, segundo Bell (2006) [31]:

O Sistema SCOOT não pode simplesmente ser considerado como uma versão *on-line* do TRANSYT. Embora o SCOOT adote também o procedimento do “*hill climbing*”, como o TRANSYT, o procedimento é incompleto no SCOOT, pois somente é tomado um passo do procedimento antes que a nova configuração dos tempos semafóricos seja implementada. Dessa forma, não se trata de uma verdadeira otimização, pois o ponto otimizado pode se tratar de um sub-ótimo.

¹⁴ Em algumas situações, o entreverdes não é o período de transição de um estágio para outro, sendo apenas parte dele. Maiores detalhes sobre o período de transição de um estágio para outro podem ser vistos na NT 258 – Conceituação de Entreverdes

http://www.cetesp.com.br/media/829864/nt_258.pdf (Acesso em 22/09/2022).

Outra diferença do TRANSYT é que, no seu modelo *off-line*, o perfil do fluxo de entrada é calculado para obter o equilíbrio no sistema, enquanto no SCOOT o fluxo de entrada é medido pelo detector.

Para entender melhor o porquê da afirmação de que não se trata de uma verdadeira otimização, segue explicação sobre *Hill Climbing*.

De acordo com Wikipedia [32]:

Em análise numérica, *Hill Climbing* é uma técnica de otimização matemática que pertence à família de “*Local Search*”¹⁵. *Hill Climbing* é um algoritmo iterativo que começa com uma solução arbitrária para um problema e depois tenta encontrar uma solução melhor fazendo uma alteração incremental na solução. Se a alteração produzir uma solução melhor, outra alteração incremental será feita na nova solução e assim por diante até que não sejam encontradas mais melhorias.

A Figura 17 ilustra o funcionamento do procedimento.

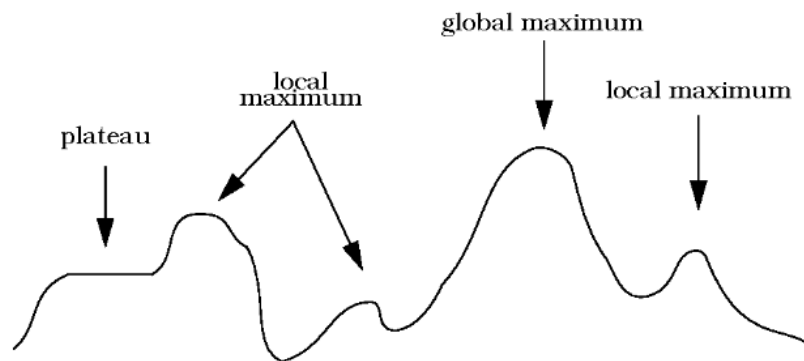


Figura 17 – Extraída da Figura 8.1: Possíveis problemas nos algoritmos de “*Hill Climbing*”: o algoritmo pode estacionar em “*platôs*” ou em máximos locais, sem achar o máximo global. Fonte: Shelby et al (2008) [33]

Dessa forma, conforme pode-se ver pela Figura 17, como o SCOOT só executa um passo do procedimento, é possível que o resultado obtido não seja o ótimo global, mas apenas um ótimo local, isto é, um sub-ótimo.

O fato de o SCOOT executar apenas um passo é decorrente de não haver tempo hábil para a execução do procedimento completo, tendo em vista que o sistema opera em tempo real. Isso parece confirmar Chow et al (2019) [12] quando afirma: o controle centralizado inviabiliza a execução completa de algoritmos eficazes de otimização.

¹⁵ “*Local Search*” é um método heurístico para resolver problemas de otimização. “*Local Search*” pode ser usado em problemas que visam encontrar uma solução maximizando um critério entre várias soluções candidatas. O algoritmo de “*Local Search*” é um processo iterativo, movendo-se de solução para solução no espaço de soluções candidatas (o espaço de busca) aplicando mudanças locais, até que a solução ótima seja encontrada ou um limite de tempo seja decorrido.

Fonte: [https://en.wikipedia.org/wiki/Local_search_\(optimization\)](https://en.wikipedia.org/wiki/Local_search_(optimization)).

Acesso em 25/06/2022.

O artigo de Wang et al (2019) [34] afirma que SCOOT usa regras heurísticas¹⁶.

O mesmo artigo ainda afirma que: “apesar de usar dados de tráfego em tempo real, o modelo de rede usado por SCOOT é construído com base em dados históricos que não são atualizados durante a operação em tempo real. Como os dados históricos podem não refletir as condições de tráfego vigentes, essa estratégia pode resultar em uma performance pobre”.

Entretanto, não se consegue atinar o porquê de o SCOOT usar dados históricos para a otimização em tempo real quando há disponíveis dados em tempo real coletados pelos detectores. De fato, conforme o artigo de Robertson et al (1991) [6]:

O modelo do SCOOT é baseado no uso de "perfis de fluxo cíclico". Um perfil cíclico é uma medida da média do fluxo de veículos durante um período do ciclo. O fluxo médio em cada período do ciclo é atualizado a cada 4 segundos. SCOOT constrói o perfil cíclico em tempo real com base nos dados coletados pelos detectores.

De qualquer forma, não se conseguiu encontrar outras fontes que pudessem justificar a afirmação de que SCOOT usa dados históricos que não são atualizados para a otimização em tempo real.

Existem no SCOOT um módulo para otimizar o tráfego em função da emissão de poluentes pelos veículos, bem como um módulo de priorização de ônibus (*Bus Priority*). Esses dois módulos não serão abordados no presente trabalho.

5.1 Evolução do SCOOT

A Tabela 8 demonstra a evolução do SCOOT desde o seu lançamento.

¹⁶ Em Ciência da Computação, normalmente existem duas propriedades principais na criação e elaboração de algoritmos: fazer o algoritmo ter um tempo de execução sempre aceitável e ser a solução ótima ou provavelmente boa para o problema em todos os casos.

No entanto, um algoritmo heurístico não cumpre uma dessas propriedades, podendo ser ou um algoritmo que encontra boas soluções a maioria das vezes, mas não tem garantias de que sempre encontrará ou um algoritmo que tem processamento rápido, mas não tem provas de que será rápido para todas as situações.

Fonte:

[https://pt.wikipedia.org/wiki/Heur%C3%ADstica_\(computa%C3%A7%C3%A3o\)#::~:~:text=A%20heur%C3%ADstica%20%C3%A9%20um%20conjunto,estuda%20a%20pesquisa%20das%20fontes.](https://pt.wikipedia.org/wiki/Heur%C3%ADstica_(computa%C3%A7%C3%A3o)#::~:~:text=A%20heur%C3%ADstica%20%C3%A9%20um%20conjunto,estuda%20a%20pesquisa%20das%20fontes.) (Acesso em 19/08/2022).

Tabela 8 – Evolução do SCOOT

Versão	Data	Funcionalidades ¹⁷
2.2	1980	
2.4	1990	SOFT Feedback Gating Bicycle SCOOT Congestion Offset
3.1	1995	Bus Priority Authority Target Saturation Level Split for Faulty Links
3.4	1999	Emissions Modelling Cycle Time Optimiser Enhancements Stop-Line Detection Flared Links Recovery from Absolute Priority Reduced Detection Modelling
4.5	2003	Supplementary Detection Enhanced Bus Priority and Statistics Enhanced Gating Logic Offset Optimisation by Emissions Simplified STOC Validation Optimisation of Un-Demanded Stages Filter Weighting Factor Cycle Time Optimiser Enhancement
MC3	2006	Congestion Supervisor Bus Priority Stage Skipping and Truncation Variable Intergreens Time Stamped Data One-Second Profiles
MC3 SP1		Pedestrian Control within SCOOT Stage Confirm Arrives Late Split Optimisation when Stages on Minimum Bus Priority on Filter Links Bus Priority – Long Journey Times Bus Priority – Cancel Detection Improved Exit Blocking Maximum Queue Clear Time on Long Links
MMX	2011	Cycle Time Independence Ghost Staging M18 Message Enhancement CHAN TPLN/MINS/MAXS/DEFS and Node Transfer Maximum Stage Lengths
MMX SP1		Journey Time Reliability

Fonte: SCOOT User Guide [35]

¹⁷ Para maiores detalhes sobre essas funcionalidades, consultar o SCOOT User Guide 666/HF/16940/000 [35].

No site do TRL – *Version History* [36], é apresentada a seguinte história de versões, Tabela 9:

Tabela 9 – História de versões do SCOOT pelo TRL

Versão	Data	Funcionalidades
3.1	1995	Bus Priority, fazendo uso do Active Vehicle Location (AVL) e Selective Vehicle Detection (SVD) ASTRID e INGRID Authority
4.0	1997	Emission data Cycle time optimizer Stop line loops
4.2	1998	Flared links Recovery from Absolute Priority Reduced detection modelling
4.4	2001	Emission optimization Enhanced Bus Priority Enhanced Gating Supplementary detectors Double cycling Filter links Automatic selection of alternative stage
4.5	1999	Simplified validation Supplementary detection Bus priority enhancements Cycle time optimization enhancements
5.0 MC3	2006	Time stamp MONACO congestion supervisor PUFFIN improvements Variable intergreen
5.1 MC3	2008	Pedestrian priority VISSIN interface Enhanced split optimization for stages on minimum and maximum Inter Green feedback with negative processing lag Partial exit blocking detection Queue clear time on long links
6.0 MMX	2010	New facilities to prioritize pedestrians Enhancement of emission model Cycle time independence Ghost staging Zero demand queue

Versão	Data	Funcionalidades
6.1 MMX	2011	MMX Service Pack 1 developed for TfL for the London 2012 Olympic Games
6.1-11 MMX	2016	Manutenção final do SCOOT disponível através dos distribuidores Siemens e Dynniq, tratando apenas de correções de bugs
7.0	2020	<p>First TRL Software only release of SCOOT</p> <p>GLOSA, or cooperative signals data to provide road users with information that can help with their journey</p> <p>Multiple split optimization allows the optimizer to make larger changes without the compromises</p> <p>Pedestrian SCOOT functionality for green man period optimized to number of pedestrians</p> <p>Modelling link departures to help with optimization in general</p> <p>Loop failure logic to reduce detection requirements</p> <p>Unlike other at market UTC systems running SCOOT, up to v.6.1.10, TRL Software have moved away from the previous reliance on OpenVMS with a cloud first approach</p>

Fonte: Version History [36]

5.2 Detecção

Uma inovação do SCOOT em relação a outros sistemas é a detecção. SCOOT não detecta veículos ou ocupação, mas LPU (*Link Profile Unit*). LPU reflete uma combinação entre o número de veículos, comprimento e velocidade. Um veículo de passeio é diferente de um caminhão, mas um veículo em situação de lentidão pode equivaler a um caminhão em trânsito livre. Da mesma forma, um fluxo baixo de veículos em trânsito lento pode se equivaler a um alto fluxo de veículos em trânsito livre. Todas as grandezas do sistema são definidas em termos de LPU: fluxo e fluxo de saturação em LPU/segundo, fila em LPU, e assim por diante.

SCOOT usa detectores localizados a montante, no fim do link que alimenta a aproximação, a cerca de 8 a 12 segundos de percurso até a linha de retenção. O sistema envia uma informação de detecção uma vez a cada 250 ms. Pode-se usar um detector por faixa de tráfego ou um detector cobrindo duas faixas de tráfego. São necessários detectores em todos os *links* da área controlada.

O sistema também possui outros posicionamentos de detectores:

- Detectores na linha de retenção;
- Redução de detectores;
- Detectores suplementares.

No caso de o detector ser laço indutivo, existem placas detetoras que possuem o recurso de configurar o “tempo de presença” que é o tempo em que uma parte do laço é ocupado (veículo estacionado), após o qual o detector volta a operar na área não ocupada (com um novo valor de indutância). Normalmente, o valor recomendado desse parâmetro é de 4 minutos.

O artigo de Robertson et al (1991) [6] reporta que estudos usando simulação sugerem que os benefícios do SCOOT são perdidos quando há perda de 15% dos detectores. Por outro lado, a experiência indica que, com procedimentos de manutenção apropriados, taxas de falha abaixo de 5% podem ser obtidas sem dificuldade.

O artigo de Hunt et al (1982) [10] afirma que o custo anual de manutenção dos detectores é de cerca 1/3 a 1/5 do custo para produzir bons planos de tempo fixo para a mesma área.

5.3 Modelo de tráfego

A operação básica do SCOOT está ilustrada na Figura .

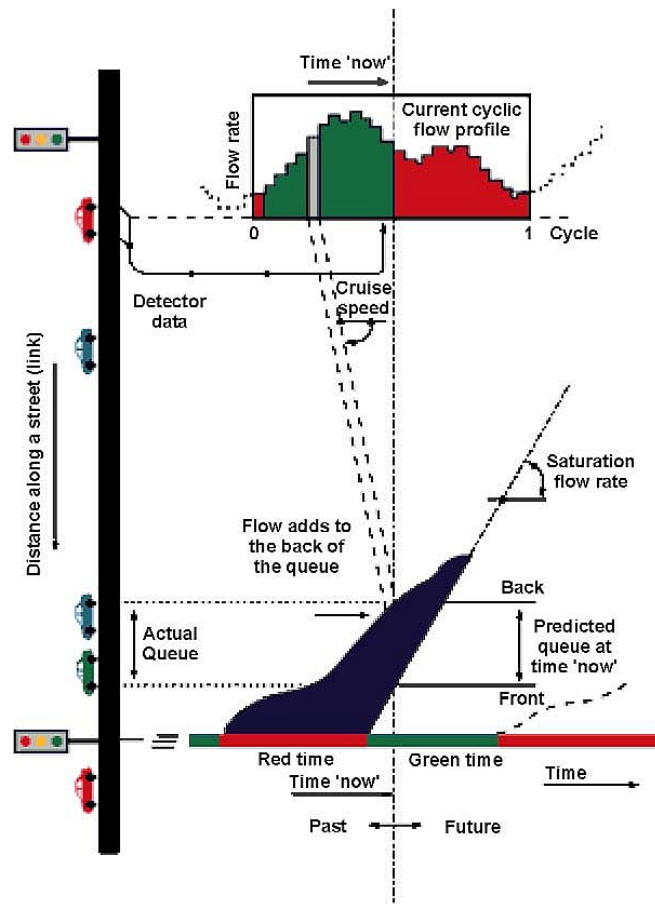


Figura 18 – Esquema operacional do SCOOT

Fonte: extraída de Aavani et al (2017) [28]

De acordo com Ming (1997) [37]:

A parte nevrálgica do sistema, onde está alojada a sua “inteligência”, é baseada em um modelo de tráfego. Em termos simplificados, o modelo, baseado no fluxo de veículos captados pelo sistema de detecção, projeta um perfil de como esses veículos chegam na linha de retenção, ao longo de um tempo de ciclo. Para a construção desse perfil na chegada dos veículos na linha de retenção, são considerados fatores como o “platô” e a dispersão de pelotão. Esse perfil não é nada mais do que um histograma (projetado) de chegada dos veículos na linha de retenção.

Assim, são considerados dois perfis de fluxo de veículos: um perfil que é captado pelo sistema de detecção e outro que é projetado pelo modelo de tráfego na linha de retenção.

O perfil que é captado pelo sistema de detecção é o “perfil cíclico do fluxo” (*Cyclic Flow Profile*) e é construído para cada *link*. O *Cyclic Flow Profile* está mostrado na parte superior da [Figura](#) (histograma colorido de acordo com a cor do semáforo a montante). Na parte inferior da [Figura](#) é mostrado o perfil de chegada dos veículos na linha de retenção, conforme foi projetado pelo modelo. No diagrama, os veículos que chegam à interseção durante o tempo de vermelho, formam uma fila que se desmancha durante o tempo de verde ao ritmo do fluxo de saturação.

Para garantir que o sistema tenha bom desempenho, é necessário que a fila projetada pelo modelo seja fiel ao que ocorre efetivamente na via. Para tanto, durante a fase de validação do *link* deve-se medir o tempo de escoamento da fila verificado na via e compará-lo com o valor modelado pelo sistema. Devem ser feitos ajustes na base de tentativa de acerto e erro até que os dois valores fiquem iguais.

O SCOOT é constituído por três otimizadores: *split*, *offset* e tempo de ciclo, que otimizam os três parâmetros de forma independente.

O SCOOT constrói dois tipos de *Cyclic Flow Profile*: o “perfil curto”, denominado “*Short Term Cycle Profile*” e o “perfil longo”, denominado “*Long Term Cycle Profile*”. O “perfil longo” é construído com base em “ciclos corridos”, utilizando valores médios suavizados. Um ciclo de “*Long Term Cycle Profile*” é formado eliminando-se o intervalo mais antigo (mais à esquerda) e agregando um novo intervalo à direita e assim sucessivamente.

O otimizador de *split* utiliza o “*Short Term Cycle Profile*” (com exceção de *links* curtos quando é usado o “*Long Term Cycle Profile*”), enquanto os otimizadores de *offset* e de *cycle* usam o “*Long Term Cycle Profile*”.

A otimização é feita de forma incremental para evitar a ocorrência de distúrbios ou perturbações, como aquelas ocorridas na mudança de planos nos sistemas de tempo fixo.

A [Tabela 10](#) resume a atuação básica dos três otimizadores.

Tabela 10 – Atuação básica dos otimizadores

Otimizador	Frequência	Mudanças possíveis	Mudanças de referência
SPLIT	Uma vez a cada mudança de estágio	-4, 0, +4	-1, 0, +1
OFFSET	Uma vez a cada ciclo	-4, 0, +4	-4, 0, +4
CYCLE	Uma vez a cada 2,5 ou 5 minutos	±16, ±8, ±4,0	±16, ±8, ±4,0

As variações possíveis mostradas na Tabela 10: (-4, 0, +4) para *split* e *offset* e de (±16, ±8, ±4, 0) para o ciclo são denominadas de “*authorities*”. Até versões anteriores à 3.1, os “*authorities*” eram fixos (iguais aos valores da Tabela 10). A partir da versão 3.1, os “*authorities*” passaram a ser configuráveis pelo usuário, sendo os valores da Tabela 10 valores *default*.

5.4 Otimizador de *split*

O otimizador de *split* atua 5 segundos antes do instante previsto de mudança de estágio (que será designado de instante de referência). Assim, cinco segundos antes do instante de referência previsto para a mudança de estágio, uma decisão deverá ser tomada pelo otimizador: encurtar o estágio em 4 segundos (-4); manter a mesma duração do estágio (0); ou prolongar o estágio em 4 segundos ($+4$). A [Figura 19a](#) mostra um exemplo onde a duração prevista do estágio é de 26 segundos. A decisão é tomada no instante 21. Se a decisão for de encurtar o estágio, ele terminará no instante 22 ($26 - 4 = 22$); se a decisão for de manter a mesma duração, o estágio terminará no instante 26 (não haverá mudança); e se a decisão for de prolongar o estágio, ele terminará no instante 30 ($26 + 4 = 30$).

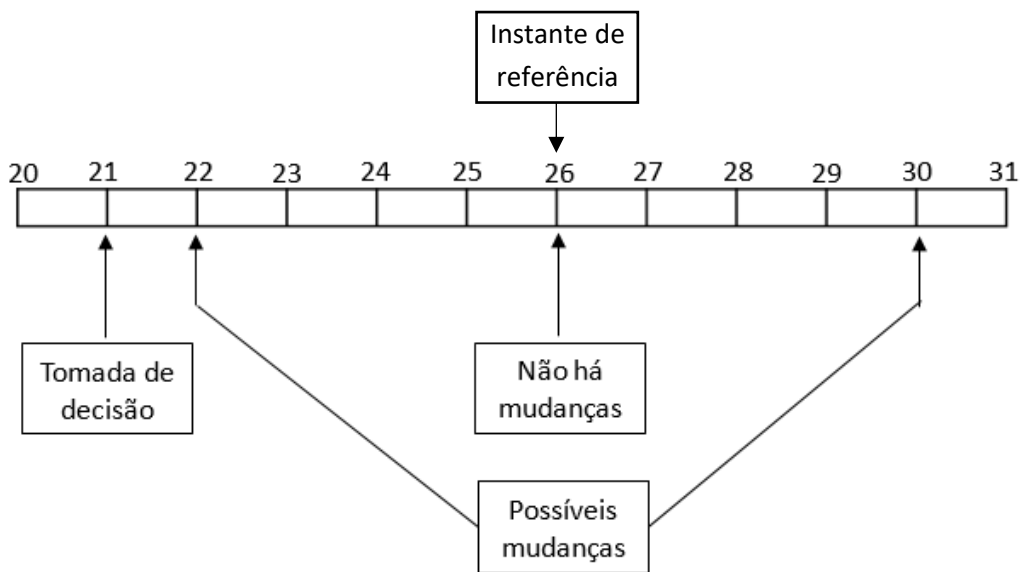


Figura 19a – Possíveis mudanças no split para uma duração prevista do estágio de 26 segundos

Para evitar mudanças muito bruscas, caso tenha havido encurtamento do estágio, no ciclo subsequente, o instante de referência passará a ser igual ao anterior diminuído de 1 segundo (e não diminuído de 4 segundos). Da mesma forma, se houve prolongamento do estágio no ciclo anterior, o instante de referência subsequente será acrescido de apenas 1 segundo (e não de 4). O acréscimo, a redução ou a manutenção do instante de referência seguem os valores de mudança de referência indicados na [Tabela 10](#), exibida anteriormente.

Assim, se a decisão tomada foi a de encurtar o estágio, no ciclo seguinte o instante previsto para a mudança do estágio (novo instante de referência) será 25 (ou seja, $26 - 1 = 25$) e a nova decisão deverá ser tomada no instante 20 (cinco segundos antes), podendo o estágio, nesse novo ciclo, terminar no instante 21, 25 ou 29, conforme se a nova decisão seja de encurtar, manter ou prolongar o estágio. Ver a [Figura 19b](#).

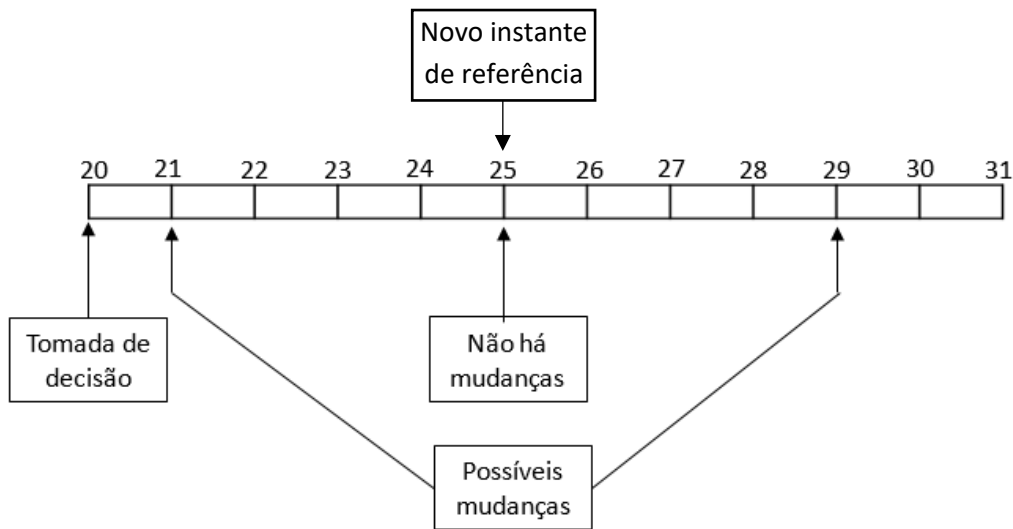


Figura 19b – Possíveis mudanças no split se no ciclo anterior (Figura 19a) a decisão tiver sido a de encurtar o estágio

Se a decisão no primeiro ciclo (Figura 19a) tiver sido de prolongar o estágio (e não de encurtá-lo), no ciclo seguinte, o novo instante previsto para a mudança de estágio será 27 (ou seja, $26 + 1 = 27$) e a nova decisão deverá ser tomada no instante 22, podendo o estágio, nesse novo ciclo, terminar no instante 23, 27 ou 31, conforme a nova decisão seja de encurtar, manter ou prolongar o estágio. Ver a Figura 19c.

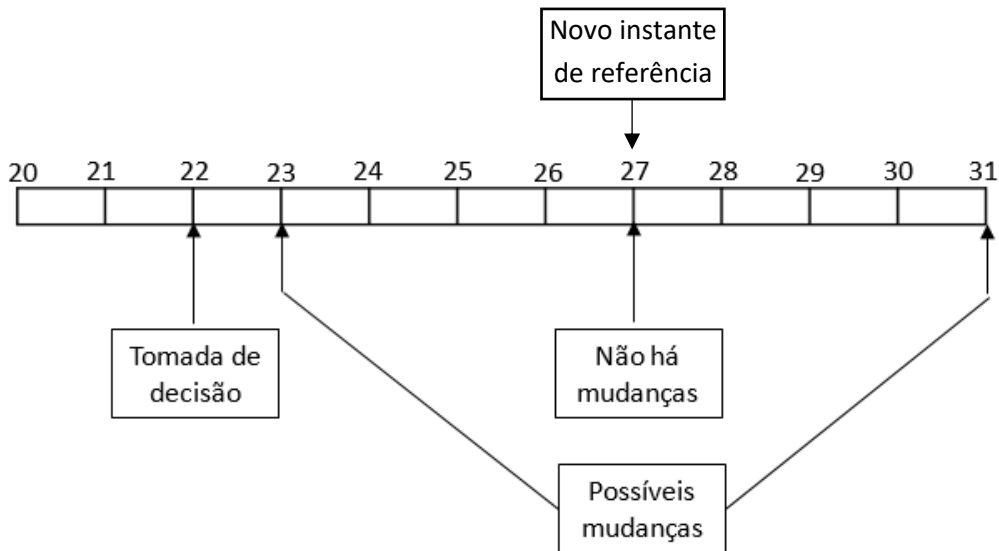


Figura 19c – Possíveis mudanças no split se no ciclo anterior (Figura 19a) a decisão tiver sido a de prolongar o estágio

Os valores das mudanças de referência da Tabela 10 (- 1, 0, +1) evitam alterações muito bruscas. Assim, tomando como base a Figura 19a como primeiro ciclo, se a decisão fosse a de prolongar o estágio por 3 ciclos consecutivos, a duração do estágio passaria de 26 para 32 segundos, em vez de 26 para 38 segundos.

A otimização do split é feita de forma a minimizar o grau de saturação máximo nos links.

5.5 Otimizador de *offset*

O SCOOT, assim como o TRANSYT, não adota a estratégia de maximização da largura da banda verde ou o conceito de “onda verde” para a otimização da defasagem (*offset*).

Conforme o artigo de Robertson et al (1991) [6]:

O grande mérito do critério de largura de banda é que não é necessário conhecer em detalhe os fluxos de tráfego e os diagramas Tempo-Espaço ajudam o engenheiro de tráfego a visualizar os padrões de fluxo. Mas não é possível traduzir as medidas de largura de banda em termos financeiros (em termos do custo de atrasos e de paradas).

Além disso, onde quer que ocorra congestionamento, o conceito de largura de banda falha porque o crescimento das filas distorce o tempo de percurso até a linha de retenção e, portanto, a velocidade de percurso adotada para a banda. Em áreas urbanas, onde há complexos movimentos de tráfego que se cruzam, “largura de banda” tem pouco significado.

Em vias de trânsito leve, com poucos veículos vindos de conversões, a otimização de largura de banda geralmente produz resultados satisfatórios. Em situações mais complexas, existem boas razões para adotar como objetivo da otimização a minimização de atrasos e do número de paradas.

O otimizador de *offset* atua uma vez a cada ciclo, tomando uma decisão no meio do estágio principal.

A decisão tomada é implementada no ciclo seguinte.

A decisão pode ser aumentar 4 segundos (+ 4), manter o valor atual (0) ou diminuir 4 segundos (– 4). No ciclo seguinte, a nova decisão é tomada com base no valor obtido no ciclo anterior.

Por exemplo:

Início: *offset* = 10 s

- Primeiro ciclo: decisão de aumentar 4 segundos → *Offset* = 14 s
- Segundo ciclo: supondo que a nova decisão também é aumentar 4 segundos → *Offset* = 18 s.

Embora o otimizador de *offset* atue uma vez a cada ciclo, como a defasagem de uma interseção é alterada em relação ao *offset* da interseção adjacente, a defasagem relativa entre duas interseções adjacentes pode ser alterada duas vezes por ciclo.

A decisão do otimizador de *offset* é tomada minimizando o valor total de PI (*Performance Index*) em todos os *links* a montante e a jusante.

O PI é uma função do atraso, número de paradas e congestionamento.

O índice de paradas é o número de paradas multiplicado por um fator denominado “*stop penalty*”.

Stop penalty é o valor em segundos equivalente a uma parada. *Stop penalty* é a ponderação em relação ao atraso, utilizado no cálculo do *Performance Index* (semelhante ao TRANSYT). Quanto maior o *stop penalty*, mais ênfase é colocada no número de paradas em detrimento do atraso.

De acordo com SCOOT Traffic Handbook SCOOT 0474, 2000) [38]:

O valor de *stop penalty* deve representar o atraso adicional (em segundos) para um veículo fazer uma parada. Por exemplo, em um percurso por três cruzamentos, é preferível que um motorista pare uma vez com 30 segundos de atraso, ou parar duas vezes com 15 segundos de atraso cada vez, ou mesmo parar em cada cruzamento, mas apenas por três segundos de cada vez. Um valor 0 (zero) significa que não será considerado o número de paradas, até 20.000, quando nenhum atraso será levado em conta. Na prática, porém, geralmente deve ser usado um valor de *stop penalty* na faixa de 8 a 30. Um Relatório do TRL (LR934) sugere que um valor de cerca de 20 segundos otimizará o consumo de combustível e, na ausência de quaisquer outros requisitos, isso é recomendado para sistemas SCOOT. Há evidência empírica que sugere que valores muito altos de *stop penalty* não têm o efeito que poderia ser esperado. Isso provavelmente se deve ao modelo de paradas relativamente simples usado pelo SCOOT, e, portanto, valores acima de 30 não são recomendados.

Segundo o Relatório do TRRL de Hunt et al (1981) [9]:

O otimizador de *offset* usa a informação do perfil cíclico para estimar se alterações no *offset* da interseção irão melhorar a progressão do tráfego considerando as interseções a montante e a jusante. Isso é feito comparando a soma dos PI's da interseção e das interseções adjacentes, somas essas resultantes de simulações de pequenas alterações no *offset* da interseção. É implementado o *offset* que produz a menor soma do PI. A Figura 20 ajuda a entender melhor a ação do otimizador de *offset*. A Figura 20 mostra uma pequena rede hipotética, composta pelas interseções 16, 27, 28, 29 e 37. Os *links* representados pelas setas pontilhadas em azul não afetam a otimização, pois são "pontas de rede" (não há outras interseções próximas). As decisões do *offset* na interseção 28 são afetadas pelas interseções 16, 27, 29 e 37. Essas 4 interseções compõem a "miniárea" da interseção 28 para efeitos de otimização do seu *offset*. O otimizador de *offset* procura achar o ponto de mínimo global do PI para a miniárea. É possível que os *offsets* das interseções 16, 27, 29 e 37 não estejam em seus mínimos globais. Mas, mesmo assim, a interseção que é o centro da miniárea (no exemplo, a interseção 28) pode estar no mínimo global.

É provável, mas não se pode garantir com certeza que, se todas as miniáreas estiverem próximas do seu mínimo global, então toda a rede também estará próxima do mínimo global.

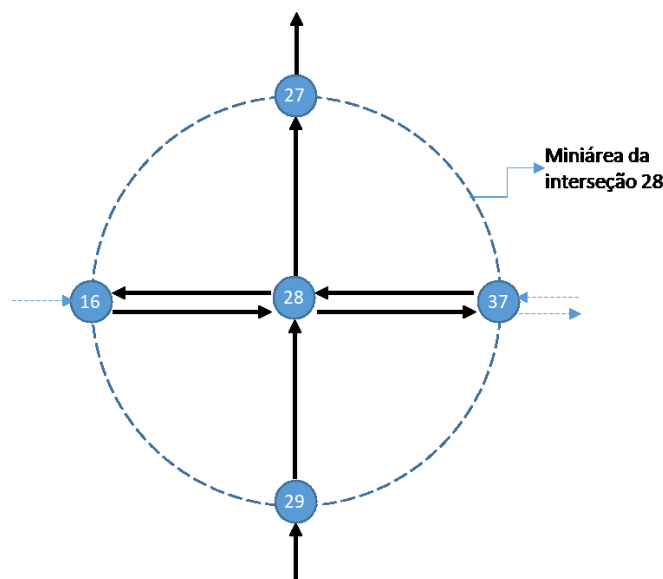


Figura 20 – Miniárea da interseção 28 para a otimização do *offset*
Fonte: Adaptado da Figura 7 de SCOOT 0414 (2000) [39]

5.6 Otimizador do tempo de ciclo

Um conjunto de interseções que opera com um tempo de ciclo comum é denominado no SCOOT de “Região”. O agrupamento de interseções em regiões é feito pelo usuário.

O otimizador do tempo de ciclo atua uma vez a cada 5 minutos. Entretanto, em determinadas condições de tráfego ele pode atuar a cada 2,5 minutos. A passagem de 5 minutos para 2,5 minutos é automática em condições crescentes de ciclo.

A partir da versão 4, o período de atuação do otimizador pode ser configurado pelo usuário.

Passos de incremento/decremento do tempo de ciclo:

- Ciclo de 32 a 64 segundos: passo de 4 segundos
- Ciclo de 64 a 128 segundos: passo de 8 segundos
- Ciclos maiores de 128 segundos: passos de 16 segundos.

Para cada interseção é calculado um tempo de ciclo chamado “ciclo ideal” (*Ideal Node Cycle Time*). O ciclo ideal é calculado em função do grau de saturação desejado para o *link* mais saturado da interseção. O grau de saturação desejado é denominado de “*Target Degree of Saturation*”.

À primeira vista, pode-se pensar que é desejável que se trabalhe com valores baixos de grau de saturação. No entanto, lembrando que o grau de saturação é a razão entre o fluxo e a capacidade (ver a definição no Glossário do item 1), um grau de saturação baixo significa haver capacidade ociosa e, portanto, tempo de verde ocioso. Assim, para efeitos de programação semafórica, o objetivo é sempre trabalhar com grau de saturação próximo de 100% (para que não haja desperdício de tempo de verde).

No SCOOT, o valor *default* do *Target Degree of Saturation* é 90%.

O ciclo ideal do nó é calculado pela seguinte expressão:

$$INCT = \frac{TS \times C \times LT}{TS \times C - NS \times (C - LT)}$$

Onde:

INCT = *Ideal Node Cycle Time*

TS = *Target Degree of Saturation*

C = tempo do ciclo corrente

LT = tempo perdido (*Lost Time*)

NS = nível de saturação do link mais saturado da interseção

Nota: No SCOOT *User Guide* [35], o *Ideal Node Cycle Time* é designado como IDNCY (em vez de INCT).

Para *TS* = 90%:

Se qualquer *link* da interseção tiver grau de saturação maior que 90% então seu *INCT* é aumentado até que tenha 90% de saturação. Neste caso, $INCT > C$.

Se todos os *links* da interseção tiverem o grau de saturação menor que 90%, então seu *INCT* é diminuído até atingir 90% de saturação. Neste caso, $INCT < C$.

No caso de $NS = TS = 90\%$, então $INCT = C$.

Segue um exemplo em que $NS < TS$:

$$\begin{aligned} TS &= 90\% \\ C &= 120 \\ LT &= 15 \\ NS &= 80\% \end{aligned}$$

$$INCT = \frac{90\% \times 120 \times 15}{90\% \times 120 - 80 \times (120 - 15)} = 67,5$$

No exemplo $NS = 80\% < 90\%$, então seu ciclo ideal é diminuído de 120 (ciclo corrente) para 67,5.

Define-se como Ciclo Mínimo Prático (*Minimum Practical Cycle Time – MPCT*) de uma interseção o número inteiro (maior que o *INCT*) que seja múltiplo de 4, 8, ou 16 e que seja mais próximo do *INCT*.

No exemplo dado, tem-se:

$$\begin{aligned} INCT &= 67,5 \\ MPCY &= 68 \end{aligned}$$

O maior *MPYC* da Região será o ciclo da Região.

A otimização permite a implementação automática de ciclos duplos, de acordo com a configuração do usuário.

5.7 ASTRID e INGRID

SCOOT pode operar de forma integrada com ASTRID e INGRID.

ASTRID (*Automatic SCOOT Traffic Information Database*) é um banco de dados usado para armazenar e analisar dados coletados pelo sistema de detecção. Usando ASTRID é possível analisar dados como fluxo, número de paradas, atraso, níveis de saturação etc.

INGRID (*Integrated Incident Detection*) é um sistema automático de detecção de incidentes, construído com base nos dados do sistema de detecção.

5.8 Congestionamento

Situações de congestionamentos podem ser identificadas diretamente por meio de detectores SCOOT.

Para tanto, deve-se localizar o detector em uma posição tal que o fim da fila normal do semáforo (fila de espera semafórica quando o semáforo está vermelho) não esteja sobre o detector. Dessa forma, se por determinado período, o detector estiver permanentemente ocupado, isto significará que a aproximação está congestionada.

a) Fila máxima

A Fila Máxima de um *link* é a fila desde a linha de retenção até o detector, em LPUs.

O Tempo de Dispersão da Fila Máxima (*Maximum Queue Clear Time*) é o tempo de verde necessário para desmanchar a Fila Máxima. O Tempo de Dispersão da Fila Máxima deve ser fornecido ao sistema durante o processo de validação do *link*. Com base no valor do Tempo de Dispersão da Fila Máxima, o sistema calcula o tamanho da Fila Máxima para o *link*.

b) Intervalos congestionados

Pode-se definir a proporção de congestionamento no ciclo conforme descrito abaixo.

Define-se intervalo congestionado como sendo um intervalo de 4 segundos (16 intervalos de 1/4 de segundo) em que o detector esteja permanentemente ocupado, conforme [Figura 21](#).

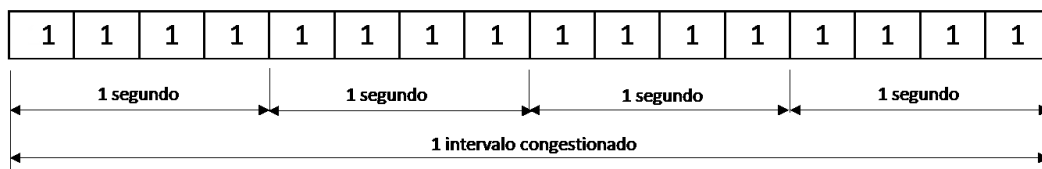


Figura 21 – Intervalo congestionado

Na [Figura 21](#), o “1” significa detector ocupado.

A proporção de congestionamento no ciclo é dada pela expressão:

$$\frac{4 \times N}{C}$$

Onde:

N = número de intervalos congestionados no ciclo

C = tempo de ciclo

Para evitar que o sistema interprete como congestionamento quando um veículo estaciona sobre o detector, por exemplo, o sistema faz uma verificação de consistência. Essa consistência é realizada por meio da verificação da fila modelada pelo SCOOT. Se a fila modelada pelo SCOOT for maior que a metade da Fila Máxima, então o intervalo de 4 segundos com ocupação contínua será considerado um intervalo congestionado consistente. No caso contrário, esse intervalo, embora haja ocupação permanente, será considerado como um intervalo congestionado não consistente. A proporção de congestionamento no ciclo considera apenas os intervalos consistentes. Entretanto, existe a possibilidade de inibir a verificação da consistência. Neste caso, qualquer intervalo de 4 segundos com ocupação contínua será considerado como intervalo consistente, independentemente do tamanho da fila modelada. Existem situações específicas em que se recomenda a inibição da verificação da consistência.

c) *Split*

c-1) *Congestion importance factor*

Se houver congestionamento em um *link*, o fato é levado em consideração pelo otimizador. Neste caso, a proporção do ciclo anterior que estava congestionado é incluída com o grau de saturação no processo de otimização. O termo de congestionamento permitirá que um *link* congestionado tenda a obter mais tempo de verde, qualquer que seja o grau de saturação calculado pelo modelo. O tamanho do efeito que o termo de congestionamento terá é determinado por um parâmetro, configurável pelo usuário, denominado de “*congestion importance factor*”.

Basicamente, o *congestion importance factor* é usado:

- Em *links* que sejam gargalos para evitar que cruzamentos a montante sejam bloqueados; ou
- Para priorizar determinados *links* em detrimento de outros durante o congestionamento.

O que torna o sistema eficiente no manejo do congestionamento é o fato de que a quantidade de verde extra dada ao *link* por causa do congestionamento é proporcional ao mesmo. Se aumentar o congestionamento, o sistema dará automaticamente mais verde. Se o congestionamento diminuir, a quantidade de verde extra dada ao *link* também diminuirá. Assim, o sistema age de forma responsiva e dinâmica, não só em função do fluxo de tráfego, mas também à quantidade de congestionamento existente em cada instante.

Segue um exemplo de aplicação do “*congestion importance factor*”, adaptado de *SCOOT Traffic Handbook* (SCOOT 0474, 2000) [38], conforme [Figura 22](#):

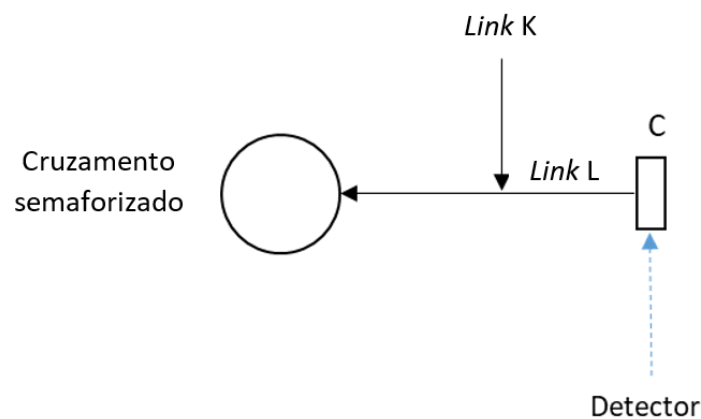


Figura 22 – Exemplo adaptado de *SCOOT Traffic Handbook* (SCOOT 0474, 2000) [38]

O diagrama da [Figura 22](#) representa um cruzamento semaforizado alimentado pelo *link* L. Existe uma alimentação forte proveniente do *link* K. Entretanto, devido à distância, não foi possível locar o laço detector a jusante de K, de forma a poder detectar não só o fluxo de L como também o de K. Também não foi possível locar um outro detector em K. Assim, foi locado o detector na sua posição normal, em C. Obviamente, essa disposição não é ideal para o SCOOT, pois o sistema não detecta o fluxo vindo de K. O sistema modela a fila utilizando-se apenas do fluxo detectado em C. A consequência disso é que a fila modelada será sempre menor que a fila real. O otimizador de split também não atuará de forma adequada, pois tenderá a dar menos verde do que o *link* realmente precisa, fazendo com que a fila cresça. Como fazer para que o sistema responda de forma adequada à demanda vinda de K e de L?

Uma solução possível para esta situação é localizar o detector em C o mais próximo possível do cruzamento, de forma que qualquer aumento de fluxo faça com que a fila fique em cima do detector. Dessa forma, o sistema acusará uma grande quantidade de intervalos congestionados. É importante, neste caso, inibir-se a verificação da consistência, uma vez que a fila modelada será menor do que a real. Isto garantirá que todos os intervalos congestionados sejam considerados como consistentes. Com um valor razoavelmente alto de *congestion importance factor*, o sistema dará um verde maior por causa do congestionamento, compensando o verde a menos que o otimizador de *split* normalmente daria. Uma resposta adequada do sistema dependerá da habilidade do engenheiro de tráfego em ajustar todos os valores de forma que a quantidade de verde dado a mais pelo sistema devido ao congestionamento seja exatamente igual à quantidade de verde dado a menos pelo otimizador de *split* por não poder detectar o fluxo vindo de L. Por este exemplo, pode-se verificar que existe uma íntima relação entre a localização dos detectores e a forma de como se deve configurar o sistema.

Uma outra possível solução para o exemplo da [Figura 22](#) pode ser viabilizada por meio de uma validação conveniente do *link*, de forma a diminuir o valor do fluxo de saturação para compensar o fluxo não detectado em K. Por exemplo, suponha que no período da manhã o fluxo em K é desprezível e a validação é feita normalmente, encontrando-se para fluxo de saturação um valor de “*m*”. À tarde, o fluxo em K é significativo. Então é feita nova validação à tarde, encontrando-se para fluxo de saturação um novo valor “*n*”, onde $m > n$. O menor valor do fluxo de saturação à tarde irá fazer com que o sistema dê mais verde ao *link* para a mesma quantidade de fluxo detectada. O verde adicional é obtido às custas de um valor artificialmente mais baixo do fluxo de saturação. O sistema estará bem ajustado quando a quantidade de verde exatamente necessária para escoar o fluxo de K for atendida por meio do ajuste do fluxo de saturação. De forma ideal, deveria haver uma validação toda a vez que o fluxo de K apresentasse uma variação, isto é, o valor do fluxo de saturação deveria variar em função da variação do fluxo de K. Na prática, o que se consegue na validação é obter um valor médio do fluxo de saturação que represente o fluxo médio de K.

Se o fluxo vindo de K for razoavelmente regular e uniforme (fluxo constante), este último método é mais eficiente, pois o sistema reage ao fluxo de forma mais imediata.

Por outro lado, se o fluxo vindo de K for irregular e totalmente aleatório (ou se vem em pelotões) então o método pela validação do valor do fluxo de saturação já não é mais adequado, pois é impossível obter um valor médio do fluxo de saturação que possa representar de forma adequada a grande variação de fluxo em K. Neste caso, o método de *congestion importance factor* configura-se mais eficiente, pois o sistema reage à variação do fluxo (tanto em L como em K). A desvantagem, no caso do método de *congestion importance factor*, é que o sistema não reage de forma imediata (reage somente quando a fila atinge o detector em C).

Percebe-se pelo exemplo que podem existir várias possibilidades para resolver um determinado problema. Cabe ao usuário determinar para cada caso o leque de alternativas possíveis e descobrir qual a mais eficiente para o caso em questão.

c-2) *Congestion link*

Além do *congestion importance factor*, existe o recurso do *congestion link*. Trata-se do recurso de priorizar um *link* por meio do congestionamento de outro *link*. Considere o diagrama da [Figura 23](#).

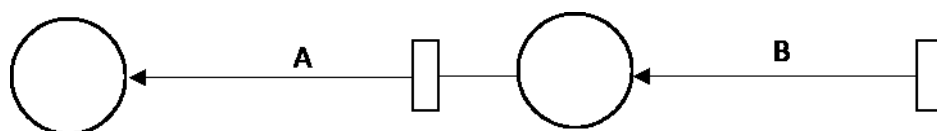


Figura 23 – Ilustração do congestion link

Na Figura 23, é possível configurar o link B como sendo o *congestion link* do link A, isto é, os dados de congestionamento do link B serão usados para otimizar o link A. Logo, toda vez que houver congestionamento em B, o link A será priorizado.

Considere a situação representada pelo diagrama da Figura 24.

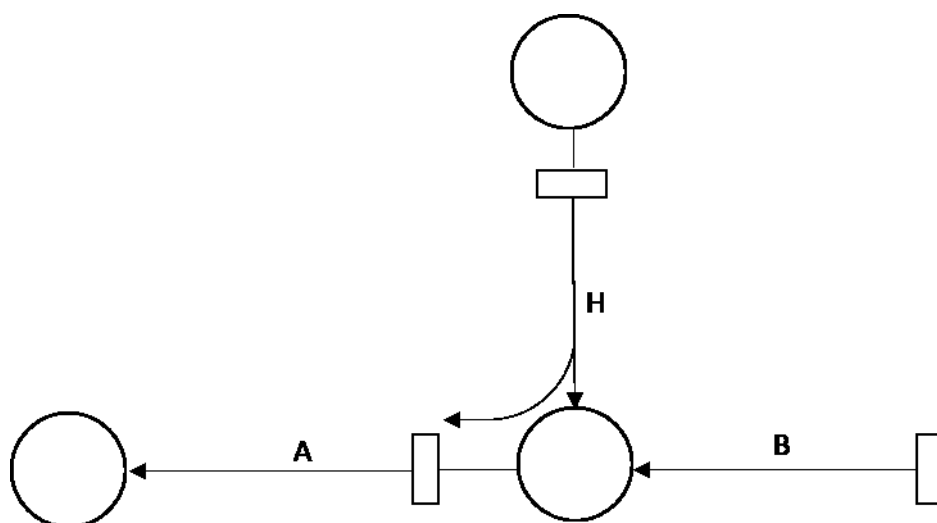


Figura 24 – Exemplo de aplicação do congestion link

No exemplo representado na Figura 24, o link A é um link importante que precisa ser priorizado.

Entretanto, o link A é um link curto. O link A é alimentado pelo link B e por um elevado volume de conversão proveniente do link H. Por isso, o link A fica constantemente congestionado nos horários de pico. Como o link A é importante e deve ser priorizado, então a primeira ideia é configurar um valor alto de *congestion importance factor* para este link. Como consequência, o SCOOT irá tentar eliminar o congestionamento em A, dando mais verde para este link. Quando o congestionamento é eliminado, o SCOOT volta à operação normal (pois o *congestion importance factor* só age no congestionamento). Com o tempo, o link A volta a ficar congestionado, repetindo-se o ciclo. Assim, o link A sofre um efeito oscilatório que causa a perda da defasagem e provoca distúrbios ao tráfego. Eventuais efeitos da oscilação em algum link podem ser identificados com o acompanhamento de Mensagem SCOOT, que fornece os valores do índice de congestionamento do link.

Qual seria então a solução para o problema de oscilação dado no exemplo da Figura 24?

Uma solução possível seria definir o link B como sendo o *congestion link* do link A e configurar um valor baixo para o *congestion importance factor* do link A (ao contrário da ideia inicial). Assim, o link A só seria priorizado quando o link B estivesse congestionado. Uma regra prática é que quando o link é curto, isto é, quando a fila normal formada durante o período de vermelho se estende até o detector, o *congestion importance factor* deve ser configurado com o valor 0.

Com esta configuração, o efeito de oscilação é evitado, pois o *link A* continuará a ser priorizado mesmo quando não estiver mais congestionado, bastando para isso que o *link B* esteja congestionado. Como o *link A* é um link curto, tanto o *link A* quanto o *link B* necessitam de mais verde na tentativa de desmanchar o congestionamento.

Também aqui o grau de intensidade da reação do sistema dependerá do grau de congestionamento (só que do *link B*). Novamente, uma resposta adequada ao sistema dependerá do acerto na escolha dos valores para os diversos parâmetros envolvidos, o que só será obtida após várias tentativas e experiências.

Nos horários fora de pico, quando não há problema de saturação no *link B*, esta configuração não tem efeito sobre a atuação do SCOOT.

Uma outra aplicação do *congestion link* é a priorização de *links* que tenham o seu detector a jusante da linha de retenção. Como o detector está locado após o cruzamento, ele não consegue detectar eventuais congestionamentos no *link e*, dessa forma, não é possível priorizar o *link* através do uso de *congestion importance factor*. A alternativa é usar um outro *link* como seu *congestion link*.

d) *Offset*

O congestionamento também é usado pelo otimizador de *offset* de forma que um *link* com congestionamento pode ser priorizado em relação a *links* sem congestionamento. Os parâmetros são *congestion offset* e *congestion weithing factor*. Esta facilidade permite que o usuário especifique uma defasagem para congestionamento e um fator de ponderação que encorajará o otimizador de *offset* a escolher esse valor de *offset* quando o congestionamento estiver presente. Valores maiores do *congestion weithing factor* farão com que o sistema tente utilizar mais o valor do *congestion offset* deste *link* em relação a outros *links*.

e) *Exit blocking*

Normalmente, em situações de congestionamento mais severo, a simples utilização do *congestion importance factor* não é suficiente para minimizar os efeitos do congestionamento. Isto porque o congestionamento pode se alastrar por vários links consecutivos, de forma que nada adianta ter um *congestion importance factor* alto no link se o link à frente (a jusante) estiver bloqueado.

O SCOOT apresenta o recurso de *exit blocking* para minimizar o efeito de bloqueios de cruzamentos. Se a fila modelada pelo SCOOT indicar uma posição de bloqueio do *link*, então o sistema não mais tentará dar mais verde neste *link* enquanto esta situação persistir. O efeito do *exit blocking* na prática é que o verde do *link* é cortado antes do previsto, mudando o direito de passagem para a via transversal.

O *exit blocking* ocorre no modelo quando o *link* a jusante de um *link* mostra intervalos congestionados.

f) *Gating*

Gating é uma facilidade disponível ao usuário a qual permite o controle da demanda numa determinada área. Um exemplo seria o controle de um *link* de entrada da rede. Quando a saturação da rede estiver acima de um valor limite, o *link* de entrada terá seu tempo de verde reduzido para diminuir o fluxo de entrada na rede.

O resultado seria a realocação de filas para a periferia da rede.

5.9 Pontos fracos e desvantagens do SCOOT

As desvantagens do SCOOT podem ser colocadas em termos de desempenho e em termos operacionais.

5.9.1 Em termos de desempenho

O SCOOT apresenta bom desempenho para a maior parte dos casos. Porém, em duas situações específicas pode ter um desempenho mais pobre. Essas duas situações são:

- a) Interferências na via, no trecho entre o detector e a linha de retenção; e
- b) *Links* curtos.

Essas duas desvantagens não são exclusivas do SCOOT, mas são intrínsecas a qualquer sistema adaptativo de tempo real (conforme a conceituação dada no item 2.2.2) e que tenha os detectores a montante da interseção, no final do *link*. Sistemas que tenham outro posicionamento dos detectores podem não ter essas desvantagens, mas poderão ter outras justamente pelo posicionamento dos seus detectores.

De acordo com SCOOT *Traffic Handbook* (SCOOT 0414,2000) [39]:

Sempre que um detector é coberto por uma fila estacionária, a detecção da demanda não dependerá mais apenas do comportamento do tráfego a montante. Em tais circunstâncias, a demanda detectada torna-se menos útil para coordenação dos semáforos. Dando um exemplo extremo: se o detector estiver localizado apenas um pouco antes da linha de retenção, então o primeiro veículo a chegar no período de vermelho irá ocupar o detector e não haveria como prever quantos outros veículos ainda vão parar no sinal vermelho. Assim, os detectores são mais úteis para definir progressões de tráfego quando eles estão posicionados além do final da fila formada no período de vermelho. Assim, é possível prever o tamanho da fila formada. Uma posição na extremidade a montante de um *link* é a que mais pode fornecer informações adequadas, tanto para definir progressões de tráfego, como para prever o escoamento da fila formada durante o tempo de vermelho.

- a) Interferências na via, no trecho entre o detector e a linha de retenção

Os detectores são os “olhos” do sistema. O sistema não “enxerga” o que acontece antes ou depois dos detectores. Assim, se houver perturbações ou interferências no trecho entre o detector e a linha de retenção, tais como manobras, estacionamentos, carga e descarga, obstruções temporárias etc., a modelagem do tráfego feita pelo sistema não vai refletir com fidelidade a realidade que ocorre na via. Dessa forma, em *links* onde ocorrem essas interferências, o sistema poderá não apresentar um resultado tão bom quanto desejado.

Neste caso, o sistema somente reagirá quando a fila atingir o detector, podendo, a partir daí, ser tratado com alguns recursos, como os mostrados no item 5.8.

Conforme as características das interferências, o problema também pode ser minimizado utilizando-se outros recursos como, por exemplo, a validação do link e SOFT.

b) *Links* curtos

Os *links* curtos apresentam uma dificuldade óbvia: se for locado detector no *link*, a fila formada durante o período de vermelho ficará estacionada em cima do detector.

Para situações com *links* curtos, também existem configurações do sistema que podem minimizar o problema, como os exemplos mostrados no item 5.8.

5.9.2 Em termos operacionais

Em termos operacionais, SCOOT apresenta as seguintes desvantagens:

- a) Custo de manutenção: SCOOT apresenta um alto custo de manutenção devido à infraestrutura de detecção e de comunicação necessárias; e
- b) Complexidade: SCOOT apresenta um alto nível de sofisticação técnica e exige pessoal técnico qualificado para sua validação e operação.

6. SCATS: *Sydney Coordinated Adaptive Traffic System*

SCATS, juntamente com SCOOT, é um dos sistemas adaptativos mais populares e é largamente utilizado no mundo. Há também abundante material na literatura. Nesta pesquisa, SCATS aparece em 20,1% de todo material pesquisado (Tabela 3). Segundo a página do *Transport for NSW* (s.d.) [40], SCATS foi desenvolvido em 1975 pelo Governo de NSW (*New South Wales*, na Austrália) e está instalado em mais de 55 mil interseções, em 187 cidades e em 28 países (das 55 mil interseções, cerca de 11 mil estão na Austrália).

Em dezembro de 2019, *Transport for NSW* começou a comercializar o SCATS. Atualmente, SCATS continua sendo aprimorado por *Department of Transport for NSW*. A última versão é SCATS 6.9.4.

6.1 Arquitetura do sistema

O sistema é composto por um computador *Central Manager* e por *Regional Servers*.

A Figura 25 mostra esquematicamente a arquitetura do SCATS.

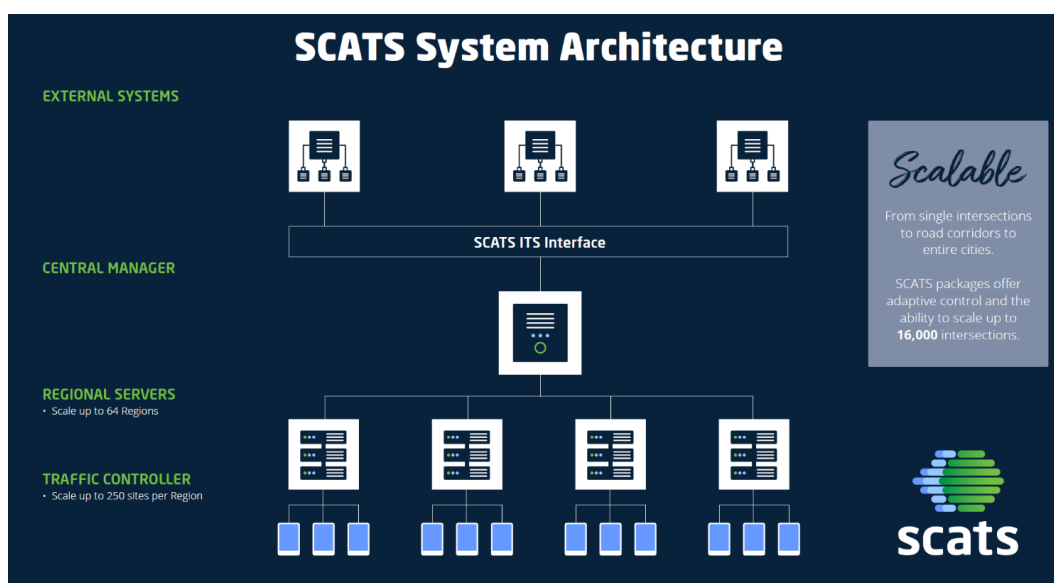


Figura 25 – Arquitetura do SCATS
Fonte: *Transport for NSW* (s.d.)

Segundo o documento Smardi (2001) [41], cada *Central Manager* pode controlar até 32 *Regional Servers*, sendo que cada *Regional Server* pode controlar até 250 controladores, totalizando 8.000 controladores por *Central Manager*.

Já de acordo com o documento SCATS 6 (s.d.) [42], o sistema comporta um máximo de 64 *Regional Servers*, cada um controlando até 250 interseções, totalizando um máximo de 16.000 interseções.

6.2 Comunicação

Segundo Wikipedia (2022) [42]:

SCATS pode operar com PABL, ADSL, PSTN e 3G IP, bem como pode operar com uma rede própria (Sydney opera dessa forma).

O protocolo usado é proprietário e está integrado na própria plataforma do sistema (ver o item 4.3.8). Portanto, não é possível a interoperabilidade de controladores. Segundo Smardi (2001) [41], SCATS requer um controlador específico com a funcionalidade do sistema. Dependendo do controlador, é possível fazer um upgrade e ser equipado com um módulo com as funcionalidades do SCATS.

6.3 Funcionamento do sistema

A Figura 26 mostra esquematicamente o funcionamento do SCATS.

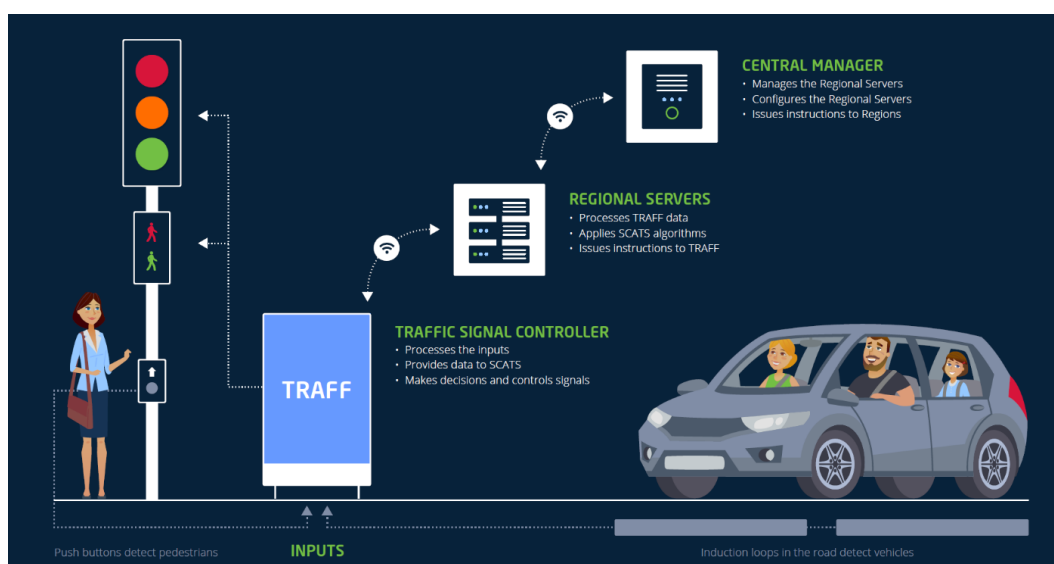


Figura 26 – Funcionamento do SCATS

Fonte: Transport for NSW (s.d.)

SCATS não é baseado em modelos de tráfego.

Ao contrário do SCOOT, SCATS não faz propriamente uma otimização. Em vez disso, SCATS atua como um sistema de *feedback* heurístico, ajustando os tempos semafóricos com base nas mudanças nos fluxos de tráfego dos ciclos anteriores.

O SCATS é composto por dois módulos: *Flexilink* e *Masterlink*. O *Flexilink* é o modo em tempo fixo e o *Masterlink* é o modo adaptativo.

O sistema opera em dois níveis: nível estratégico e nível tático.

O controle estratégico é exercido pelos *Regional Servers*. Usando os dados coletados pelos detectores, o sistema determina o tempo de ciclo, *split* e *offset*.

O controle tático é feito pelos controladores, em um esquema de atuação onde o verde (calculado pelo controle estratégico e considerado como verde máximo na atuação) pode ser encurtado e estágios omitidos. Neste esquema, o estágio principal não pode ser omitido e nem encurtado. Qualquer tempo de verde economizado em um estágio como resultado do seu encurtamento pode ser transferido para outros estágios ou adicionado ao estágio principal.

A transferência do tempo economizado para o estágio principal pode apresentar dois problemas:

- a) O tempo foi economizado num ciclo e transferido para o estágio principal no ciclo seguinte. É possível que esse tempo adicional não seja aproveitado devido a um decréscimo da demanda neste ciclo. Não se sabe se existe realimentação da informação, onde o sistema considere esse tempo economizado no seu cálculo do *split* ainda para o ciclo em questão.
- b) A antecipação do verde do estágio principal altera a defasagem. Seria necessário verificar se o benefício desse verde adicional supera o prejuízo causado pela alteração da defasagem.

6.4 Detecção

Os detectores são locados junto à linha de retenção em cada faixa de tráfego.

Conforme Smardi (2001) [41], “os detectores junto à linha de retenção produzem dados precisos de movimentos de conversão que são usados para a determinação do *split*. SCATS é muito bom na otimização do *split* graças aos dados de conversão coletados pelos detectores na linha de retenção”.

No entanto, não se consegue entender como os detectores junto à linha de retenção, que, normalmente são laços indutivos, conseguem captar o fluxo de conversão. No mínimo teria que haver um outro detector na via transversal ou na esquina anterior (após o cruzamento). Neste caso, o fluxo de conversão poderia ser deduzido pela diferença dos fluxos nos dois detectores.

O posicionamento dos detectores junto à linha de retenção apresenta a desvantagem de não permitir a obtenção de informação sobre a demanda (fluxo que chega), mas apenas sobre o fluxo que sai.

6.5 Grau de saturação

6.5.1 Grau de saturação conforme definição usual

A expressão usual do grau de saturação é:

$$x = \frac{F}{F_S \times \frac{g}{C}}$$

Onde:

x = grau de saturação

F = fluxo que chega na aproximação

F_S = fluxo de saturação

g = tempo de verde

C = tempo de ciclo

O numerador é o fluxo que chega e o denominador é o maior fluxo que pode ser escoado.

Para que $x > 1$: $F > F_S \times \frac{g}{C}$

Grau de saturação maior que 1 significa que o fluxo que chega é maior do que o fluxo que consegue escoar, portanto, tendo uma fila remanescente no fim do verde.

No caso do SCATS, não se tem informação do fluxo que chega (numerador), mas apenas do fluxo que sai (denominador).

6.5.2 Grau de saturação do SCATS

Ao contrário do SCOOT, em que o grau de saturação é estimado, SCATS calcula o grau de saturação diretamente dos detectores junto à linha de retenção. Porém, o grau de saturação medido pelo SCATS não é propriamente o grau de saturação conforme definido usualmente (apresentado no item 6.5.1). Para não confundir os dois conceitos, o grau de saturação medido pelo SCATS é designado como “grau de saturação do SCATS” (*SCATS Degree of Saturation*) e será representado como DS.

O grau de saturação DS medido no último ciclo é usado para determinar o *split* e o tempo de ciclo para o próximo ciclo.

Conforme SCATS 6 (s.d.) [42], “para fazer os ajustes no tempo de ciclo, *split* e *offset*, SCATS se baseia na medição do grau de saturação, designado como DS (*Degree of Saturation*). Os dados coletados pelos detectores são enviados do controlador para o *Regional Server* que calcula o DS. Valores de DS maiores que 1 ocorrerão em regime de saturação e SCATS irá rapidamente responder a essa situação.”

Contudo, graus de saturação (conforme definição usual) maiores que 1 significam que o verde foi insuficiente para escoar a fila. Ora, devido ao posicionamento dos detectores na linha de retenção, SCATS não tem informação do tamanho da fila e não sabe se houve fila residual após o final do verde. SCATS também não tem informação sobre o valor do fluxo de saturação. Em vez do fluxo de saturação, SCATS usa o parâmetro “máximo fluxo medido” (*maximum measured flow*).

Conforme a resposta do *Help Desk* do *Transport for NSW* em e-mail de 05/09/2022, “o fluxo máximo medido é o maior fluxo coletado durante o dia. Os cálculos do fluxo máximo medido são realizados à meia-noite nos dias configurados. Se a opção da região DM não for configurada, o fluxo máximo será medido nos dias definidos na Configuração da Região SCATS. O *default* é apenas nos dias úteis da semana. Se a opção de região DM estiver configurada, o fluxo máximo será medido em todos os dias da semana”.

Smardi (2001) [41] dá a seguinte expressão para calcular o DS:

$$DS = \frac{\textit{green} - \textit{unused green}}{\textit{green}}$$

Onde:

- *green* = tempo de verde disponível; e

- $unused\ green = (\text{soma de todos os } gaps \text{ durante o período de verde, em segundos}) - (\text{número de veículos}) \times (gap \text{ correspondente ao máximo fluxo medido, em segundos}).$

Alternativamente,

$unused\ green = (\text{número de veículos}) \times (gap\ médio \text{ durante o período de verde, em segundos}) - (\text{número de veículos}) \times (gap \text{ correspondente ao máximo fluxo medido, em segundos}).$

$unused\ green = 0$ se o gap médio no período de verde for igual ao gap correspondente ao máximo fluxo medido;

$unused\ green > 0$ se o gap médio no período de verde for maior que o gap correspondente ao máximo fluxo medido; e

$unused\ green < 0$ se o gap médio no período de verde for menor que o gap correspondente ao máximo fluxo medido.

Então:

$unused\ green > 0 \rightarrow DS < 1$: o fluxo verificado no ciclo é menor que o fluxo máximo medido.

$unused\ green = 0 \rightarrow DS = 1$: o fluxo verificado no ciclo é igual ao fluxo máximo medido.

$unused\ green < 0 \rightarrow DS > 1$: o fluxo verificado no ciclo é maior que o fluxo máximo medido.

A dúvida aqui é, em condições saturadas, o fluxo máximo medido tenderá ser igual ao fluxo de saturação e o fluxo medido no ciclo não poderá ser maior que o fluxo máximo e não existirá a possibilidade de $DS > 1$. Assim, parece ser um contrassenso que, justamente na saturação, DS não possa ser maior que 1 e que, em condições não saturadas (quando o fluxo máximo é menor que o fluxo de saturação), o DS possa ser maior que 1. Para confirmar se esse entendimento está correto ou não, seria necessário ter maior conhecimento do sistema. A alternativa seria que o fluxo máximo será sempre menor que o fluxo de saturação, mesmo em condições saturadas. Neste caso, deve-se verificar o conceito do “fluxo máximo” e como é ele medido ou calculado.

Por outro lado, Lee et al (2002) [44] fornece a seguinte expressão para o cálculo de DS :

$$DS = \frac{(g - \sum gaps + v \times sp(sf))}{g}$$

Onde:

g = tempo de verde

$\sum gaps$ = soma de todos os $gaps$ durante g , em segundos

v = número de veículos detectados durante g

$sp(sf)$ = gap médio correspondente ao **fluxo de saturação**

Neste caso, não há como DS ser maior que 1, pois $\sum gaps$ será sempre maior ou igual a $v \times sp(sf)$.

6.6 Car Equivalent Flow (VK)

Segundo Smardi (2001) [41], “Car Equivalent Flow” ou fluxo equivalente é utilizado na seleção do *offset* e é calculado pela expressão:

$$VK = DS \times g \times MF$$

Onde:

VK = Car Equivalent Flow

DS = grau de saturação do SCATS

g = tempo de verde, em segundos

MF = máximo fluxo medido, em veículos por segundo

6.7 Tempo de ciclo

A descrição de como o sistema determina o tempo de ciclo será feita com base nas seguintes fontes:

6.7.1 SCATS 6 (s.d.) [42]

O tempo de ciclo é aumentado ou diminuído para manter o DS próximo a 0,9 na faixa de maior saturação. O tempo de ciclo pode variar entre 20 e 240 segundos, mas um limite inferior (geralmente entre 30 e 40 segundos) e um limite superior (geralmente entre 100 e 150 segundos) podem ser definidos pelo usuário.

6.7.2 Day et al (2010) [45]

O tempo de ciclo é definido de acordo com dois cenários: de baixo volume e alto volume. No cenário de baixo volume, a seleção do tempo de ciclo é baseada em fluxo do link (*link flow* – LF). No cenário de alto volume, o tempo de ciclo é calculado usando DSs.

6.7.3 Smardi (2001) [41]

O SCATS tem valores de DS definidos pelo usuário para determinar a relação entre o DS medido e o tempo de ciclo. Essa relação é então usada para selecionar um tempo de ciclo alvo para o qual o tempo de ciclo real se move. O tempo de ciclo pode se aproximar do ciclo alvo em incrementos/decrementos de até +/- 6 segundos. A calibração e escolha do valor de DS e outros valores definidos pelo usuário são essenciais para um desempenho eficaz e essas habilidades são alcançadas apenas através de anos de experiência.

6.7.4 Fellendorf et al (1997) [46]

O tempo de ciclo C é uma função do maior valor de DS medido na subárea durante ciclos anteriores. O tempo de ciclo pode variar de um ciclo para o próximo de no máximo +/- 6 segundos. Uma função linear $f(C, DS)$ é usada para ajustar o tempo de ciclo entre ciclos sucessivos.

6.7.5 Stevanovic et al (2009) [47]

Segundo Stevanovic et al (2009) [47], o tempo de ciclo é ajustado de forma diferente conforme o fluxo.

Para fluxo baixo, o tempo de ciclo varia conforme o fluxo. No caso, “fluxo baixo” significa que o maior valor de DS da subárea é menor que um valor definido pelo usuário (SZ1-10).

Para fluxo baixo, são definidos 3 valores de tempo de ciclo.

- (LOWPER): entre 40 e 60 segundos.
- (STOPPER 1): entre 50 e 70 segundos.
- (STOPPER 2): entre 75 e 95 segundos.

Conforme o valor do fluxo, é escolhido um dos 3 valores de tempo de ciclo entre LOWPER e os STOPPERS. É estabelecido um conjunto de regras para evitar pulos frequentes entre os LOWPERs e os STOPPERS.

Para fluxo alto (o maior valor de DS da subárea é maior do que SZ1-10), o tempo de ciclo deixa de ser controlado pelo fluxo. Neste cenário, o tempo de ciclo é função de DS.

Para alto fluxo, é calculado um tempo de ciclo recomendado RLo com base numa relação linear entre tempo de ciclo e DS. Os valores pré-definidos nessa relação linear são:

Tempo de ciclo	DS
HIPER	SZ2
XPER	SZ1
MaxSTOPPER (1 ou 2)	SZ1-10

Por exemplo:

Tempo de ciclo	DS
HIPER = 150	SZ2 = 115%
XPER = 110	SZ1 = 93%
MaxSTOPPER (1 ou 2) = 80	SZ1-10 = 83%

Os 6 valores formam duas relações lineares entre o tempo de ciclo e DS. RLo é calculado usando uma das duas relações dependendo do valor de DS medido.

Com base nos 6 valores do exemplo, as duas relações lineares são:

$$RLo = 1,818DS - 59,074$$

$$RLo = 0,333DS + 52,361$$

Se o DS medido for 90%, então será usada a segunda relação e o valor de RLo será:

$$RLo = 0,333 \times 90 + 52,361 = 82$$

Após a determinação de um RLo inicial, há um conjunto de ajustes. O tempo de ciclo é atenuado e ponderado com base nos últimos 3 ciclos para evitar grandes flutuações. É permitido que o tempo de ciclo seja alterado até ± 6 ou ± 9 segundos em relação ao ciclo anterior.

6.7.6 McCann (2014) [48]

Esse documento descreve o sistema implantado em Dublin, Irlanda. Segundo esse artigo, o tempo de ciclo de uma subárea é determinado em função do fluxo e DS.

O usuário define os seguintes parâmetros para cada subárea:

- Tempo de ciclo mínimo (Min CT)
- Tempo de ciclo máximo (Max CT)
- Tempo de ciclo "Stretch" (Stretch CT)
- Tempo de ciclo "Alt Min 1" (Alt Min 1)
- Tempo de ciclo "Alt Min 2" (Alt Min 2)

Um estágio "Stretch" é configurado em cada interseção. A principal característica desse estágio é que ele recebe as "sobras" de verde dos outros estágios quando o tempo de ciclo estiver acima do tempo de ciclo "Stretch". Por exemplo, numa subárea com tempo de ciclo máximo de 120 segundos e um tempo de ciclo "Stretch" de 100 segundos, qualquer aumento do tempo de ciclo acima de 100 segundos será dado diretamente ao estágio "Stretch".

Em baixo volume, a subárea irá rodar com o tempo de ciclo mínimo até que seja alcançado um volume pré-definido pelo usuário, quando, então, o tempo de ciclo da subárea irá pular do tempo de ciclo mínimo para o tempo de ciclo "Alt Min" (podendo ser especificados até dois "Alt Min"). Em Dublin, o volume limite para o tempo de ciclo ir do mínimo para "Alt Min" é de 8 veículos por ciclo.

A Figura 27 mostra a relação dos 5 parâmetros de tempo de ciclo definidos pelo usuário em relação ao volume.

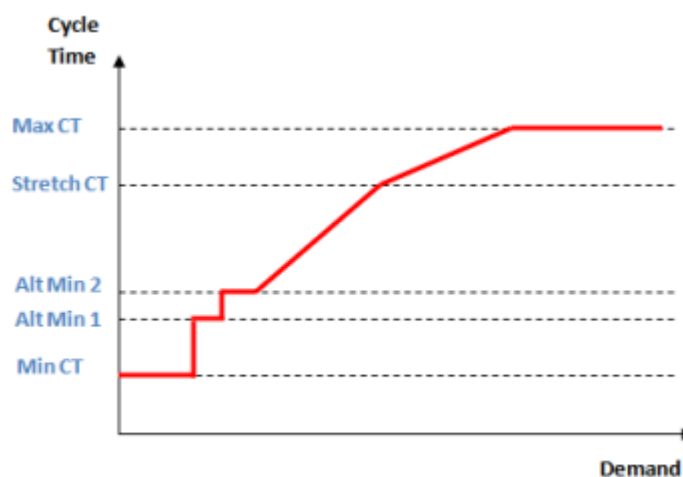


Figure 8: Cycle Time Reference Values

Figura 27 – Reprodução da Figura 8 do artigo de McCann (2014) [48]

Até o “Alt Min 2”, o tempo de ciclo é selecionado em função de volumes pré-definidos pelo usuário. A partir do “Alt Min 2”, o tempo de ciclo é definido pelo DS. Entre “Alt Min 2” e “Stretch CT”, prevalece uma relação linear entre o tempo de ciclo e DS. Entre o “Stretch CT” e “Max CT” prevalece uma segunda relação linear entre o tempo de ciclo e DS.

6.8 Split

Da mesma forma que o tempo de ciclo, a descrição de como o sistema determina o *split* será feita com base nas mesmas fontes.

6.8.1 SCATS 6 (s.d.) [42]

Splits são especificados como uma porcentagem do tempo de verde em relação ao tempo de ciclo e podem variar de um pequeno valor a cada ciclo, de forma a manter graus de saturação iguais em aproximações concorrentes.

6.8.2 Day et al (2010) [45]

Splits são ajustados minimizando DS nas aproximações selecionadas pelo operador. Isso é feito ajustando os *splits* de forma incremental para reduzir o DS do movimento mais forte. Isso tende a favorecer o estágio ativo se houver demanda de forma sustentada (quanto mais verde tem, mais verde usa).

6.8.3 Smardi (2001) [41]

SCATS tem uma biblioteca de planos de *split*. A cada ciclo, o sistema escolhe o plano que determina melhor a igualdade do grau de saturação nas aproximações concorrentes. O plano que melhor expressa essa igualdade é denominado plano “equisat”. “Equisat” é a situação em que o DS das aproximações é igual. Os valores máximos de DS projetados são calculados para cada plano usando o valor de DS do último ciclo e o plano com o menor valor do máximo DS é selecionado.

6.8.4 Fellendorf et al (1997) [46]

A duração do verde de um estágio depende do tempo de ciclo e do plano de *split*. Para cada controlador, há um conjunto de 4 ou 8 planos de *split*. Os planos de *split* podem ser trocados de um ciclo para outro. O algoritmo basicamente procura por um plano que forneça o menor valor de DS com base no volume do ciclo anterior.

6.8.5 Stevanovic et al (2009) [47]

O ajuste de *split* em SCATS equaliza a saturação de uma interseção minimizando o DS máximo em todas as aproximações concorrentes (definidas pelo usuário). SCATS geralmente seleciona o plano de *split* mais apropriado usando um método chamado *Incremental Split Selection* (ISS). O ISS é um processo de incremento e decremento de *split* em cada ciclo, onde o objetivo é reduzir o DS do movimento mais movimentado. Os planos de *split*, que podem variar até $\pm 4\%$ do tempo de ciclo, são armazenados nas tabelas do ISS.

Os tempos de verdes são limitados por tempos mínimos de estágio (verde mínimo mais *intergreen*) e *splits* são gerados dinamicamente pelo método ISS. SCATS suporta atuação; portanto, um estágio pode terminar a qualquer momento entre o verde mínimo e o *split* determinado pelo respectivo plano.

6.8.6 McCann (2014) [48]

Cada interseção da subárea divide um tempo de ciclo comum e opera com o plano de *split* de mesmo número.

SCATS pode selecionar o melhor conjunto de *splits* para cada subárea usando um dos dois métodos:

- *Discrete Split Selection (DSS)*
- *Incremental Split Selection (ISS)*

Usando DSS, SCATS seleciona de forma automática, ciclo a ciclo, o plano mais apropriado para as condições de tráfego vigentes, de uma tabela de planos pré-definida. SCATS usa o DS da faixa mais saturada para estimar como uma mudança do plano de *split* afetaria o fluxo. Por exemplo, se o DS for 67% e se o *split* for 50% no Plano 1, o DS estimado para o Plano 2 cujo *split* é de 55% seria:

$$\text{DS estimado} = 67 \times 50/55 = 61\%$$

Assim, selecionar o Plano 2 resultaria em um valor de DS menor. Esse procedimento é efetuado para todas as aproximações e para todos os planos e o plano escolhido será aquele que resultar no menor valor de DS.

ISS opera de uma forma similar, exceto que a seleção não é feita de uma tabela pré-definida de planos. Os *splits* são incrementados/decrementados de um pequeno valor a cada ciclo objetivando obter o menor valor de DS possível para a faixa mais saturada. SCATS armazena em seu *software* tabelas com possíveis variações de *split* para 2, 3 e 4 estágios. Por exemplo, uma tabela com 2 estágios com possíveis incrementos/decrementos é mostrada na tabela seguir:

Estágio	0	1	2	3	4	5	6
A	0	+1	-1	+2	-2	+3	-3
B	0	-1	+1	-2	+2	-3	+3

Se os *splits* forem A = 56% e B = 44%, então no próximo ciclo, os possíveis *splits* serão:

Estágio	0	1	2	3	4	5	6
A	56	57	55	58	54	59	52
B	44	43	45	42	46	41	47

A decisão de qual *split* selecionar é tomada comparando os valores de DS que cada *split* irá produzir. Assumindo que o estágio A tenha um DS de 82% e o estágio B um DS de 70%, o DS estimado associado a cada *split* será:

Estágio	0	1	2	3	4	5	6
A	82	81	83	79	85	78	87
B	70	72	68	73	67	75	66

A mudança número 5 produz o resultado mais “Equisat” e esse valor de *split* é adotado para o próximo ciclo. O verde do estágio A é aumentado em 3% e o verde do estágio B é reduzido em 3%. ISS suporta até 4 estágios.

6.9 Offset

A descrição do funcionamento da determinação do *offset* será feita utilizando-se das mesmas fontes, com exceção de Fellendorf et al (1997) [46] que não apresenta uma descrição sobre este tema.

6.9.1 SCATS 6 (s.d.) [42]

Os *offsets* são selecionados para cada subárea e também entre as subáreas que podem se interconectar. Subáreas com fluxos mais baixos podem não receber boa coordenação se o tempo de ciclo for inadequado. No entanto, quando as condições de tráfego permitem o uso de um tempo de ciclo que pode fornecer boa coordenação entre várias subáreas, o sistema tende a manter esse tempo de ciclo mesmo que um tempo de ciclo menor seja suficiente. O SCATS faz isso porque os *offsets* ideais nos *links* de fluxo mais alto minimizam o número total de paradas no sistema, reduzindo o consumo de combustível e aumentando a capacidade da rede.

6.9.2 Day et al (2010) [45]

SCATS não força todas as interseções de uma rede a usar o mesmo tempo de ciclo. Em vez disso, com base na qualidade da progressão, ele “casa” seletivamente as subáreas para que tenham uma boa progressão. Alternativamente, SCATS se “divorciará” das subáreas que têm progressão ruim ou que tenham diferenças muito grandes no tempo de ciclo.

Os ajustes de *offsets* são realizados dentro de subáreas. Vários planos de *offsets* predeterminados estão disponíveis nos quais o sistema “vota” para implementação. É possível para o operador direcionar a decisão para um padrão de fluxo específico, mas geralmente o procedimento busca o melhor *offset* para o maior volume de tráfego. A maneira pela qual isso é especificamente realizado não está bem documentada. Como não há detectores a montante, SCATS não pode desenvolver perfis de fluxo e não pode otimizar ou avaliar *offsets* dinamicamente. A qualidade da progressão no SCATS parece depender da qualidade dos planos de *offsets*.

6.9.3 Smardi (2001) [41]

SCATS tem uma biblioteca de planos de *offsets* em função do tempo de ciclo. SCATS calcula um parâmetro para diferentes planos de *offset* para cada *link*. Este parâmetro é calculado pela multiplicação do fluxo equivalente VK por um fator direcional (*directional bias value* – DB) para cada plano de *offset*. O plano com o maior valor resultante é escolhido.

O texto não explica e nem detalha o que é “*directional bias*”. No entanto, podem se encontrar informações sobre DB no item 6.9.4 a seguir.

O fluxo equivalente VK está descrito no item 6.6.

6.9.4 Stevanovic et al (2009) [47]

Em SCATS, os ajustes de *offset* seguem a seleção do plano de *offset* (chamado de *Link Plan* ou LP em SCATS), que representa um par de *offsets* acompanhado por um par de tempos de ciclo.

SCATS possui quatro LPs que oferecem *offsets* para facilitar uma boa progressão para várias condições de tráfego entre duas interseções. Assim, na convenção, SCATS comum, LP 1 e LP 3 proporcionam boa progressão em ambas as direções. LP 2 e LP 4 fornecem boas progressões para altos fluxos de entrada e saída, respectivamente. O usuário pode definir quatro valores de *Direcional Bias* (DB) (um para cada um dos quatro planos de *offset*), que representam pesos no processo de votação para cada *link*. Esses valores DB são multiplicados por fluxos equivalentes (VK) médios de três ciclos para cada aproximação e esses produtos são somados para um LP. O LP com a soma mais alta é selecionado. Para evitar altas flutuações de LPs, um LP deve obter 4 ou 5 votos consecutivos antes da implementação.

Os *offsets* dentro de um LP são ajustados de acordo com uma relação linear com os tempos de ciclo.

Cada um dos LPs terá dois valores de *offset* (*Low Offset* e *High Offset*) para dois tempos de ciclo limítrofes. Por exemplo, LP 3 pode ter *Low Offset* e *High Offset* de 2 e 8 segundos que correspondem a tempos de ciclo de 90 e 125 segundos, respectivamente. Então, os *offsets* para todos os tempos de ciclo menores que 90 segundos serão de 2 segundos, enquanto os *offsets* para todos os tempos de ciclo maiores que 125 segundos serão de 8 segundos. Para todos os outros tempos de ciclo entre 90 e 125 segundos, os *offsets* serão interpolados entre 2 e 8 segundos. Cada um dos tempos de ciclo de baixo volume é geralmente atribuído a um LP que fornece boa progressão em ambas as direções.

Se houver uma boa progressão entre as subáreas, SCATS irá 'casá-las'. O 'casamento' pode ser permanente ou condicional, sujeito a diferenças de tempo de ciclo (≤ 10 segundos). Alternativamente, as subáreas podem 'se divorciar' se a diferença entre os tempos de ciclo for superior a 10 segundos.

6.9.5 McCann (2014) [48]

A coordenação entre as interseções dentro de uma subárea é alcançada pela especificação de um *offset*. O *offset* entre duas interseções em uma subárea pode ser especificado em quatro planos diferentes. O *offset* é um valor positivo ou negativo com referência a um instante do estágio principal da interseção crítica da subárea; se houver apenas uma interseção na subárea, o *offset* se referencia a si mesmo.

Os *offsets* são especificados para tempos de ciclo baixos e altos. Os tempos de ciclo em que um *offset* baixo ou alto deve ser usado são inseridos manualmente pelo operador. Para tempos de ciclo entre esses valores, SCATS interpola o *offset*, o que significa que SCATS calcula automaticamente o *offset* apropriado com base em uma proporção dos valores limites.

Por exemplo, os valores para o Plano de *offset* 2 são -4 e -6 segundos. O valor de -4 segundos é usado quando o tempo de ciclo é de 90 segundos e o *offset* de -6 segundos é usado quando o tempo de ciclo é de 114 segundos. Se o tempo de ciclo for de 120 segundos, será usado o valor de *offset* de ciclo alto (-6 segundos).

6.9.6 Considerações sobre *offset*

Nenhuma das fontes acima mencionadas traz informação sobre possível prejuízo à coordenação entre as interseções devido à antecipação da abertura do verde em função da atuação, onde as "sobras" de verde são transferidas para o estágio principal.

Outra consideração a ser feita é sobre o “casamento” e “divórcio” de subáreas. O critério para a junção ou a separação das subáreas parece ser bastante simplório, levando em consideração apenas o tempo de ciclo e não os tempos de verde, a distância e o fluxo entre elas. Para que a coordenação tenha benefícios, é necessário que os veículos cheguem na interseção a jusante em formação de pelotão.

De fato, conforme citado por Lin et al (2010) [49], Yagoda et al (1973) [50] propôs um índice $I = V/L$ para determinar se interseções adjacentes precisam ser coordenadas. A necessidade de coordenação I é diretamente proporcional ao volume V e inversamente proporcional ao comprimento do *link*. Se I for maior que 0,5 a necessidade de coordenação deverá ser considerada.

Chang (1985) [51] propôs o seguinte procedimento para analisar a necessidade de coordenação entre interseções adjacentes por meio de um índice I :

$$I = \frac{1}{1+t} \left[\frac{Xq}{q_1 + q_2 + q_3} \right] - (n - 2)$$

Onde:

t = tempo de percurso do link (comprimento do link dividido pela velocidade média);

X = número de faixas que saem da interseção a montante;

q = fluxo que segue em frente na interseção a montante;

q_1, q_2, q_3 = fluxos de conversão à direita, à esquerda e em frente que chegam na interseção a montante; e

n = número de faixas que chegam na interseção a jusante.

Segundo Szasz (1991) [52]:

A fórmula tradicional que garante que dois semáforos próximos não sincronizados não interfiram um no outro é:

$$TV_{max} < \frac{L}{3}$$

Onde TV_{max} é o maior dos tempos de verde dos dois semáforos, isto é, $TV_{max} = \max(TV_1, TV_2)$ e L é a distância entre os dois semáforos (1 = semáforo de trás, a jusante; 2 = semáforo da frente, a montante).

Esta condição é **suficiente** para garantir que não haja necessidade de coordenação, ou seja, a possibilidade de terem tempos de ciclos independentes. Entretanto, quando esta condição não acontece, não significa que irá ocorrer a interferência, pois:

- a) O volume que chega em 2 (semáforo a jusante) pode não ser suficiente para encher a caixa; e/ou
- b) O semáforo 2 pode abrir antes que a caixa encha.

A verificação exata da interferência exige cálculos ou fórmulas complicadas ou técnicas de simulação. Eis algumas verificações mais simples:

- 1) Verificação do tempo de verde do semáforo de trás, a montante (supondo que o da frente não tenha problemas).

a) Teste de volume máximo

Se o volume que passa num ciclo pelo semáforo de trás (semáforo 1) for inferior à caixa, não há interferências:

$$V_1 \times TC_1 < caixa$$

Onde:

V_1 = volume que sai do semáforo 1 e que vai para o semáforo 2; e
 TC_1 = tempo de ciclo do semáforo 1 (semáforo a jusante).

Se a aproximação for regular com n faixas:

$$Caixa = \frac{n \times L}{d_o}$$

Onde:

n = número de faixas; e
 d_o = distância média entre carros parados.

$$V_1 \times TC_1 < \frac{n \times L}{d_o}$$

$$TV_1 < \frac{L}{S_o \times d_o \times x}$$

Onde:

$$x = \frac{TC_1 \times V_1}{n \times S_o \times TV_1} = \text{grau de saturação da aproximação; e}$$

S_o = fluxo de saturação de uma faixa.

Adotando $S_o = 0,5$ veíc/segundo e $d_o = 6$ m, tem-se:

$$TV_1 < \frac{L}{3x}$$

b) Teste de fila máxima

Na condição mais desfavorável, se o tempo de ciclo do semáforo da frente for sensivelmente menor que o do semáforo de trás e, além disso, a capacidade do semáforo da frente for maior que a do semáforo de trás, ou ainda, parte do fluxo não vier direto para o semáforo da frente, a fila máxima pode ser menor que o volume de um ciclo. Definem-se os seguintes termos:

V_v = volume que vai de 1 para 2 no período de verde;
 V_r = volume que vai de 1 para 2 durante o período de vermelho; e
 S_2 = fluxo de saturação do semáforo 2.

Calcula-se a fila N_m na pior condição, que é o semáforo da frente ficar em vermelho quando chega o fluxo principal.

Tem-se que:

$NV = TC_1 \times V_v$ = número de veículos por ciclo no período de verde de (1);
 $NR = TC_1 \times V_r$ = número de veículos por ciclo no período de vermelho de (2); e
 $NV + NR = TC_1(V_v + V_r) = TC_1 \times V_1$ = número de veículos por ciclo de $1 \rightarrow 2$.

Calcula-se a fila N_x que escoar ao fim do verde de (1):

$$N_x = (TV_1 + TR_2) \times S_2$$

Pode-se mostrar que:

- Se $N_x > NV$: a fila N_m acaba antes do fim de verde de (1) e é dada por:

$$N_m = \frac{TR_2}{\frac{TV_1}{NV} - \frac{1}{S_2}}$$

- Se $N_x < NV$: a fila N_m acaba no período de vermelho de (1) sendo dada por:

$$N_m = NV + \frac{(NV - N_x)}{\frac{S_2 \times TR_1}{NR} - 1}$$

Calculada a fila N_m , verifica-se se ela excede a caixa:

$$N_m < \frac{n \times L}{d_o}$$

2) Verificação do semáforo da frente (o de trás tem verde menor)

Aqui, o cálculo é mais simples, pois só ocorre perda de capacidade quando o fluxo que sai do semáforo de trás não consegue se sustentar durante todo o período de verde do semáforo da frente (verde desperdiçado).

A caixa de "plena carga" NF é calculada por:

$$NF = \frac{n \times L}{d_o} + \left[L \left(\frac{1}{vd} - \frac{1}{v1} \right) - tc \right] \frac{\phi c \times S_2}{S_2 - \phi c}$$

Onde:

L = distância entre os semáforos;

n = número de faixas;

d_o = distância média entre veículos;

vd = velocidade média na descarga da fila;

$v1$ = velocidade média livre;

tc = tempo perdido na saída do semáforo;

ϕc = fluxo de conversão no vermelho = NR/TR_1 ; e

S_2 = fluxo de saturação do semáforo 2.

Se $NF < N_t$ não há prejuízo na capacidade se não houver coordenação, onde:

N_t = Fluxo que chega no semáforo de trás por ciclo

Isto é, se $NF < N_t$ não há desperdício de verde no semáforo da frente.

Os 3 trabalhos, Yagoda et al (1973) [50], Chang (1985) [51] e Szasz (1991) [52], mostram que a questão de coordenar ou não dois semáforos adjacentes é mais complexa do que um simples critério de proximidade dos tempos de ciclo.

Segundo Hamilton et al (2013) [53], a maior fraqueza do SCATS é o *offset*. Por causa dos detectores na linha de retenção, o sistema não tem ideia do tamanho da fila.

6.10 Congestionamento

De acordo com Malcontenti-Wilson et al (2013) [54], SCATS usa os valores de DS , VK e VO para indicar congestionamento quando houver as seguintes condições:

- Se VK/VO for maior que 2,4 e
- Se DS for maior que 0,95.

Onde:

VO = número de veículos que cruzam a linha de retenção durante o ciclo.

Na pesquisa, não foi localizada informação de como o SCATS reage quando há obstrução a jusante (*Exit Blocking*). Neste caso, o método usando DS (proporção do tempo de verde efetivamente utilizado) não é apropriado e não se aplica devido a não haver escoamento da fila devido à obstrução adiante, apesar de o semáforo estar verde.

7. ITACA: *Intelligent Traffic Adaptive Control Area*

De acordo com o *Abstract* do artigo López et al (1996) [55]:

O artigo descreve o sistema de controle de tráfego adaptativo ITACA e sua aplicação ao controle de tráfego nas cidades espanholas de Madrid e Barcelona. ITACA oferece resposta em tempo real às demandas de fluxo de tráfego atuais e futuras e traz 'inteligência' para o controle semafórico de tempos fixos. Ele incorpora: (1) um sistema adaptativo, que é usado para evoluir para o melhor plano em cada interseção; e (2) um Sistema Experto (*Expert System*), que pode usar todos os dados e previsões do sistema adaptativo para obter uma solução global para o plano total de tráfego. A solução encontrada pelo *expert system* é comunicada ao sistema adaptativo por um uso sofisticado de fatores de importância (*importance factors*). O sistema adaptativo possui otimizadores de ciclo, *split* e *offset*, e usa perfis para atualizar o modelo de tráfego. Os componentes do modelo incluem: (1) comprimentos de fila; (2) indicador de congestionamento; (3) carga¹⁸; e (4) modificador de fluxo de saturação. O Sistema Experto é uma parte opcional do ITACA, que utiliza os dados atuais do modelo e suas regras para ajustar os pesos de cada movimento de tráfego. Seu uso mais óbvio é evitar congestionamentos secundários e bloqueios a jusante. Espera-se que as estratégias de congestionamento se desenvolvam de forma diferente para cada rede e dependam fortemente do usuário.

Segundo o Relatório IDOM (2019 – D2) [20]:

ITACA, *Intelligent Traffic Adaptive Control Area*, Área de Controle Adaptativo de Tráfego Inteligente, trata-se do módulo de operação adaptativo comercializado pela empresa KAPSCH (antiga Telvent). É uma solução completamente centralizada que faz modificações dentro do tempo de ciclo no nível de área. Os requisitos para a sua implementação são:

- A sensorização de todas as faixas de acesso (ou pelo menos aquelas que representam mais de 25% da contribuição para o cruzamento) em termos de filas e quantidades de veículos, não considerando informações relacionadas aos movimentos dos pedestres. Essa sensorização deve ser colocada, como no caso do SCOOT, 70-100 metros antes da linha de parada (Up Stream). ITACA usa principalmente laços indutivos como sensores mas, quando é exigido pela cidade, pode usar câmeras e sensores de radiofrequência (que precisam de uma menor manutenção, é o caso de Barcelona e Buenos Aires);
- As informações de sensorização devem ser recebidas com uma frequência de cinco (5) segundos e requerem uma latência máxima de rede de 2 segundos para sua implementação. As informações dos detectores são transformadas em dois dados principais:
 - ✓ Fluxo de veículos e;
 - ✓ Comprimento da fila.
- As modificações feitas pelo algoritmo na programação correspondem a: mudança de fase¹⁹, distribuição e ciclo:

¹⁸ Carga: pelo material analisado, não foi possível determinar o conceito de "carga". Provavelmente trata-se de conceito similar à taxa de ocupação (razão entre fluxo e fluxo de saturação).

¹⁹ "Fase", no contexto do texto, significa estágio.

- ✓ Entre os 12 (doze) e 6 (seis) últimos segundos da fase, é calculada a distribuição global do cruzamento, o que pode resultar em um adiantamento ou atraso de até cinco (5) segundos da fase;
- ✓ Nos primeiros cinco (5) segundos do ciclo, a duração total do ciclo é calculada, e é aplicada ao longo da fase.
- Pode ser implementado usando os protocolos NTCIP, AENOR ou UTMC2 (atualmente em desenvolvimento) para comunicações entre o sistema central e os controladores, embora a empresa tenha implementado extensões de protocolos proprietários da KAPSCH.

A empresa também tem a implementação da geração dinâmica de planos.

Esse modo de operação já foi implementado em cidades como Madri e Valladolid (Espanha), Buenos Aires (Argentina), Cidade do Panamá (Panamá) e Quito (Equador).

IMPORTANTE: Salvo erro, as informações foram coletadas diretamente dos fornecedores e, portanto, são de responsabilidade de cada empresa.

Embora o Relatório IDOM (2019 – D2) [20] não tenha mencionado, ITACA também foi implementado na cidade de Mumbai, Índia. Segundo o site https://www.trafficinfotech.com/areatraffic_control_system_improving_traffic_in_the_island_city/ (acessado em 24/10/2022), em notícia datada de 24/05/2012, informa que:

O sistema vencedor da licitação foi o ITACA (*Intelligent Traffic Adaptive Control Area*), desenvolvido e comercializado pela empresa espanhola Telvent Trafico y Transporte – hoje propriedade da Schneider Electric da França.

O sistema é composto por:

- Detectores de veículos;
- Rede de comunicação de dados alugada da MTNL;
- Sala de Controle Central no Quartel-General da Polícia de Trânsito e Centro de Informações nos Escritórios do MCGM; e
- Obras civis, incluindo o desenvolvimento de uma rede totalmente canalizada que deverá facilitar o processo de manutenção.

As principais características do sistema são:

Os controladores de tráfego verificam os dados de detecção a cada 0,01²⁰ segundo e os transmite ao centro de controle a cada cinco segundos. O sistema armazena os dados em intervalos de cinco minutos para cada detector juntamente com o cálculo de fluxo, tempo de ocupação e carga²¹.

²⁰ Embora o site mencione que o controlador verifica os dados dos detectores a cada 0,01 segundo, esse valor causa estranheza, pois não se conhece equipamento que possua tal sensibilidade.

²¹ Carga: pelo material analisado, não foi possível determinar o conceito de “carga”. Provavelmente trata-se de conceito similar à taxa de ocupação (razão entre fluxo e fluxo de saturação).

A área pode ser dividida em subáreas de cruzamentos com comportamento de tráfego semelhante. Pode haver um número ilimitado de subáreas e elas podem ser mescladas ou subdivididas. Os três parâmetros básicos, tempo de ciclo, *split* e *offset*, são otimizados automaticamente pelo sistema para minimizar as paradas e atrasos. O sistema, no entanto, pode não aceitar picos de tráfego de curta duração, pois os cálculos consideram a média do tráfego ao longo de três ciclos, a fim de evitar mudanças drásticas.

O sucesso de qualquer sistema dependerá do nível de calibração realizado. As condições em Mumbai exigiram ampla calibração das capacidades das vias, tráfego misto e comportamento dos motoristas.

Conforme a apresentação de Srivastava (2022) [56], ITACA foi implementado em 255 interseções de Mumbai, em 2 centros de controle.

ITACA também foi implementado na cidade de Putrajaya, Malaysia, cuja conclusão final estava prevista para 2012 (Sahri, 2005) [57]. Conforme esse documento:

ITACA consiste em dois subsistemas diferentes: Sistema Adaptativo e Sistema Experto (*Expert System*), que é opcional.

O Sistema Adaptativo é responsável pela determinação dos componentes de um plano de tráfego: ciclo, *split* e *offset*, usando um método algorítmico. Suas principais características são:

- coleta e processamento de dados a cada 5 segundos;
- aplicação local de seus resultados, feita separadamente para cada interseção (coleta de informações de todos os cruzamentos próximos, para os quais há conhecimento preciso de localização na rede);
- ação, uma vez a cada vários ciclos, em cada subárea para ajustar o tempo do ciclo, como resultado dos cálculos realizados;
- ação, várias vezes por ciclo, sobre a mesma interseção distribuindo adequadamente o tempo de verde entre estágios, como resultado de cálculos realizados; e
- ação, uma vez por ciclo, em cada cruzamento para ajustar o *offset*, como resultado dos cálculos realizados.

O Sistema Experto pode trabalhar com os dados do sistema adaptativo para elaborar uma solução global para o tráfego. Só pode ser usado quando o Sistema de Controle estiver trabalhando no Modo Adaptativo, uma vez que as informações produzidas pelo Modo Adaptativo são a fonte das decisões do Sistema Experto. Este sistema também tem como intenção ser um complemento para o engenheiro de tráfego, onde ele possa apresentar o seu conhecimento de forma compreensível (regras) e a partir do qual o Sistema Experto é capaz de tomar decisões.

Isso é feito com base em um conjunto de regras estabelecidas pelo usuário para ajustar a importância de certos movimentos de tráfego. Essas regras são adequadas para congestionamentos recorrentes, ou para situações específicas. Em suma, o Sistema Experto implementa qualquer curso de ação descrito pelo usuário no conjunto de regras, e o seu Editor de Regras de Tráfego permite que o usuário defina as regras em uma linguagem de tráfego.

O Editor de Regras de Tráfego é a porta de entrada para o Sistema Experto. O conhecimento do engenheiro é traduzido em regras com a ajuda do Editor de Regras. A eficiência do Sistema Experto depende da qualidade das regras e das ações que elas contêm.

O sistema de aplicação no Centro de Controle é o OPTIMUS, sendo o seu “core” o ITACA.

ITACA trabalha com controladores de tráfego RMY. Uma função dos controladores RMY é coletar dados obtidos dos detectores e transmiti-los ou enviá-los ao controlador de rede CMY antes de serem enviados para o Centro de Controle. A função do controlador de rede CMY é coletar todos os dados de fluxo de tráfego de todos os RMY a ele conectados.

Segundo o artigo de Dubey (2019) [58], ITACA foi implementado em 128 interseções em Putrajaya.

No documento Revitalização CTA-3 (2006) [59], podem ser encontradas as seguintes informações:

- Os dados dos detectores são coletados a cada 5 segundos. Através de um processo interno, os dados são agregados até completar um minuto.
- O sistema possui um modo denominado “Micro-Regulado”. Neste modo, o tempo de ciclo, a sequência dos estágios e os entreverdes do plano semafórico são respeitados, enquanto a demanda condiciona a distribuição de verde, podendo prolongar um estágio em detrimento de outros.
- A partir dos valores de fluxo e ocupação dos detectores, o sistema elabora perfis de fluxo e tempo de ocupação, calcula as filas nos *links*, graus de saturação e tempos de desmanche de fila, além de determinar o congestionamento e a demanda em cada *link*.
- O sistema calcula um tempo de ciclo global para cada subárea levando em conta a importância (peso) de cada interseção dentro da subárea. O sistema baseia-se no conhecimento contínuo dos ciclos ótimos dos controladores e na tendência de ciclo que a situação do tráfego determina em cada momento. Para não haver precipitação nas situações pontuais de tráfego, a decisão de uma mudança do ciclo é tomada quando se constata que existe uma tendência firme, durante um número de ciclos determinado, sempre no mesmo sentido. Se a tendência de mudança se mantiver, as mudanças de ciclo tornam-se mais frequentes. Para inverter uma tendência de alteração de ciclo, o sistema assegura que os graus de saturação sejam menores/maiores e que, portanto, o ciclo deve crescer/decrescer. Uma vez decidido que o ciclo deve mudar, este varia, não no valor calculado, e, sim, em um crescimento/decrescimento fixo. Esta variação depende do valor do ciclo atual.
- O tratamento para o cálculo da distribuição de verde realiza-se em nível de cruzamento, nos últimos segundos de cada estágio. O método baseia-se em distribuir o tempo total de verde do ciclo entre todos os estágios do cruzamento, de forma que haja um equilíbrio entre as demandas (grau de saturação + congestionamento) de cada *link*.
- O tratamento para o cálculo da defasagem realiza-se ao nível de cruzamento, até o final do estágio de maior duração. A partir de todos os *links* do cruzamento que considerem relevantes para este cálculo (*links* de entrada e saída do cruzamento) e o comprimento e posição das filas em cada *link*, calcula-se a melhor defasagem, que é aquela que produz o menor atraso.
- A união entre duas subáreas implica que uma delas adote o tempo de ciclo da outra. A união das subáreas aplica-se somente quando:
 - ✓ A união significa que uma das subáreas aumente o seu ciclo, não que o diminua.
 - ✓ Só é possível entre subáreas adjacentes.
 - ✓ O fluxo total entre as subáreas supera um valor máximo total configurado ou o fluxo em um dos *links* que conecta as subáreas supera um valor máximo configurado.
 - ✓ Em cada subárea é definido um valor de prioridade. A subárea que perde o seu ciclo e adota o da outra é aquela que tem menor valor de prioridade.
 - ✓ Quando as duas subáreas possuem o mesmo valor de prioridade, a que perde o seu ciclo é a que possui o menor ciclo.

- O Sistema Experto é um complemento para a atuação do engenheiro de tráfego, onde este introduz o seu conhecimento, a partir do qual o sistema é capaz de tomar decisões em tempo real. Seu uso só é estabelecido quando o sistema de controle estiver funcionando em modo de tempo real. O usuário introduz as regras mediante o Editor de Regras de Tráfego. O Sistema Experto pode atuar de forma ativa, executando as ações que decide com base nas regras introduzidas (com ou sem confirmação por parte do usuário). Pode atuar também de forma informativa, sugerindo atuações que o usuário pode ou não executar manualmente.

8. TUC: *Traffic-Responsive Urban Control*

Segundo artigo Manolis et al (2018) [60]:

- O TUC foi desenvolvido no final da década de 1990.
- A primeira implementação do TUC ocorreu em Glasgow no âmbito do projeto europeu TABASCO (*Telematics Applications in Bavaria, Scotland and Others*). Posteriormente, foi implantado na área central da cidade de *Chania*, Grécia.
- O sucesso da primeira implementação motivou a implementação do sistema em 3 cidades com diferentes condições de infraestrutura e tráfego: *Chania*, *Southampton* e *Munich*. Mais tarde, TUC também foi implantado em Macaé e outras cidades brasileiras²². Posteriormente, uma nova implementação, com a incorporação de novos recursos, foi efetivada em *Chania*.
- TUC é composto por três módulos distintos, mas interconectados, que em intervalos regulares implementam novas configurações de tempos semafóricos. Dois deles adotam a abordagem centralizada e são responsáveis pela definição de *split* e tempo de ciclo. O terceiro módulo adota a abordagem descentralizada para a defasagem.
- Na versão original do TUC, era usado um regulador de *split* que modificava o tempo de verde do plano de tempos fixos. Em condições de saturação, a influência do plano de tempo fixo é menor, mas em períodos não saturados, o regulador adotava tempos próximos ao do plano de tempos fixos. Daí a importância de ter planos de tempos fixos atualizados e bem regulados. Para contornar essa dependência de planos de tempos fixos, foi desenvolvida uma variante híbrida do TUC, usando um modelo em tempo real de Webster para condições não saturadas, enquanto se usa a versão original do TUC para condições de saturação.
- TUC toma decisões uma vez por ciclo.
- TUC exige baixo esforço computacional.

8.1 Estratégia do TUC

²² As implantações do TUC no Brasil receberam o nome de CONTREAL. Ver o Capítulo 9.

A Figura 28 mostra a estratégia geral do TUC.

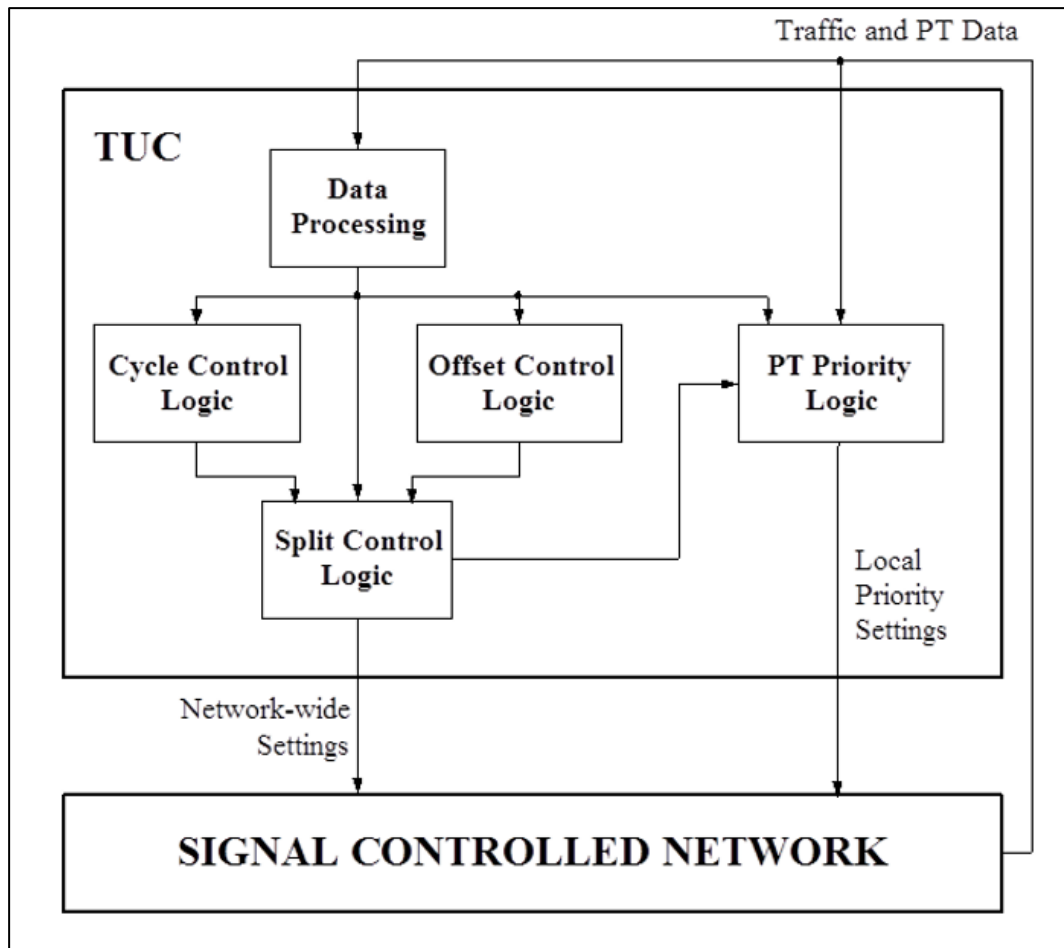


Figura 28 – Estratégia geral do TUC

Fonte: Figura extraída da apresentação de Papageorgiou (s.d.) [61]

Obs.: PT = Public Transport.

8.2 Arquitetura do TUC

A Figura 29 mostra a arquitetura do TUC. A otimização do *split*, tempo de ciclo e *offset* é feita de forma centralizada. A priorização do transporte público (PTP = *Public Transport Priority*) é feita de forma local.

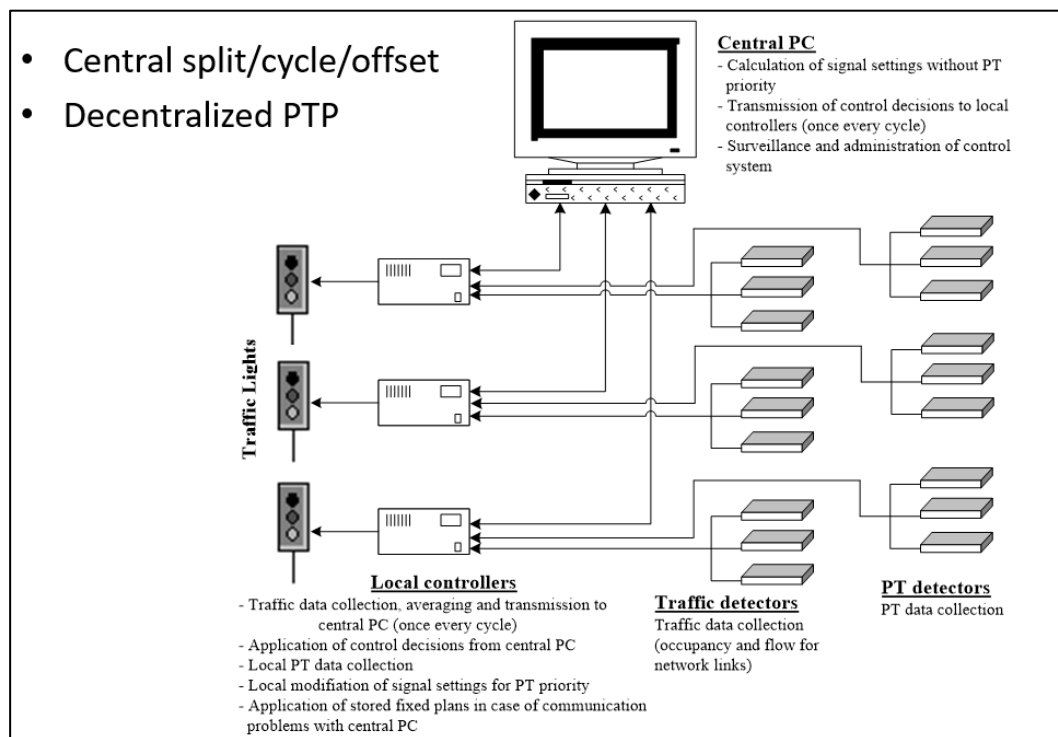


Figura 29 – Arquitetura do TUC

Fonte: Figura extraída da apresentação de Papageorgiou(s.d.) [61]

Obs.: PTP = *Public Transport Priority*.

8.3 Comunicação e protocolo

Respondendo a questionamento enviado por e-mail de 15/09/2022, o Prof. Markus Papagiorgiou esclarece que não há um protocolo específico. Para cada implementação foi usada uma conexão diferente, dependendo do sistema existente.

8.4 Detecção

De acordo com o artigo de Manolis et al (2018) [60], TUC usa um detector por *link*, que pode ser locado arbitrariamente em qualquer lugar do *link*.

Ainda no e-mail de 15/09/2022, o Prof. Markus Papagiorgiou esclarece que “TUC foi projetado para funcionar com os detectores em qualquer posição (e isso foi demonstrado em campo). Contudo, o sistema trabalha mais eficientemente se a posição do detector estiver no meio do *link* (para *links* curtos ou médios). Para *links* longos, o detector pode ser locado mais próximo da linha de retenção.”

O Prof. Papagiorgiou ainda esclarece que na versão inicial do TUC o detector coletava apenas dados de ocupação. Na versão estendida, que não necessita de planos de tempo fixo previamente estabelecidos, também são necessários dados de fluxo. Para *links* com pouca demanda não são necessários detectores. Em média, cerca de 70% dos *links* precisam ser dotados de detectores.

8.5 Abordagem “store-and-forward”

A otimização de *split* do TUC é derivada de uma abordagem denominada “store-and-forward”. A abordagem “store-and-forward” foi sugerida pela primeira vez por Gazis e Potts em 1963 (Gazis et al, 1963) [62] e, desde então, tem sido usada em várias aplicações, notadamente para controle de tráfego.

Conforme o artigo de Aboudolas et al (2009) [63], a abordagem “store-and-forward” permite uma simplificação fazendo a conversão de fluxo interrompido para fluxo ininterrupto. Considere a Figura 30.

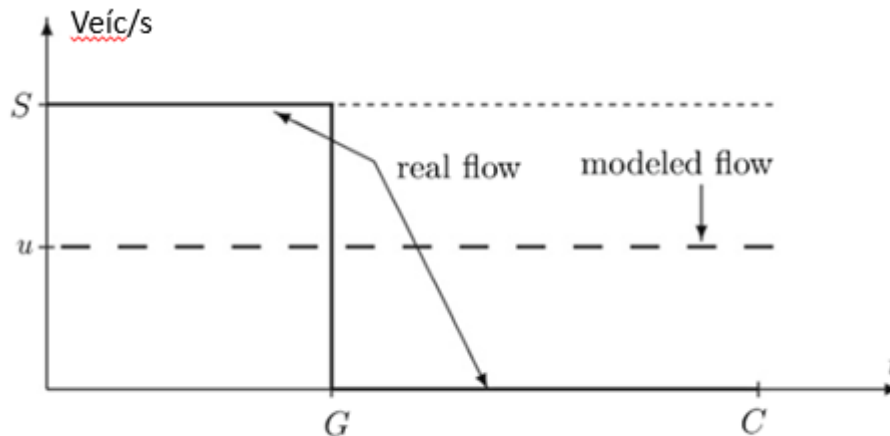


Fig. 2. Simplified modeling of link outflow.

Figura 30 – Conversão de fluxo interrompido para fluxo ininterrupto
Fonte: Figura extraída do artigo de Aboudolas et al (2009) [63]

Na Figura 30, o fluxo controlado por um semáforo é convertido para fluxo ininterrupto. Foram adotadas as seguintes simplificações no modelo:

- Durante todo o período de verde G , o fluxo escoado (fluxo de saída) é igual ao fluxo de saturação S ;
- Não há limitação de espaço a jusante; e
- Não são considerados os efeitos de transição de verde para vermelho e de vermelho para verde.

O fluxo de saída durante o tempo de ciclo C é igual a $(S \times G)/C$. Dessa forma, esse fluxo é equivalente a um fluxo ininterrupto $u = (S \times G)/C$.

Como consequência da simplificação, o modelo não reflete as oscilações verde-vermelho que ocorrem dentro dos ciclos. O modelo descreve um fluxo ininterrupto equivalente de saída para cada *link*, supondo haver demanda suficiente para sustentar um fluxo igual ao fluxo de saturação durante todo o período de verde e que haja sempre espaço suficiente a jusante. *Offset* e tempo de ciclo não têm impacto no modelo “store-and-forward”. *Offset* e tempo de ciclo podem ser fixos ou determinados em tempo real por outros algoritmos.

As consequências da simplificação são o preço a pagar para evitar um modelo mais realista e complexo de transições verde-vermelho “intra-ciclo” que pode resultar em um problema computacional de complexidade exponencial para a sua solução exata.

8.6 Otimização de *split*

O texto a seguir também é baseado no artigo de Aboudolas et al (2009) [63].

Para a compreensão da descrição do processo de otimização de *split*, considere a seguinte notação:

z = numeração de *link* da rede;

j = numeração de interseção;

I_j = conjunto de *links* de chegada da interseção j ;

O_j = conjunto de *links* de saída da interseção j ;

C_j = tempo de ciclo da interseção j ;

L_j = tempo perdido da interseção j ;

F_j = conjunto de estágios da interseção j ;

v_z = conjunto de estágios em que o *link* z tem direito de passagem;

s_z = fluxo de saturação do *link* z ;

$t_{w,z}$ = taxa de conversão, onde w são as conversões que alimentam o *link* de chegada da interseção j (é um elemento do conjunto I_j) e z é o *link* de saída da interseção j (é um elemento do conjunto O_j); e

$g_{j,i}$ = tempo de verde do estágio i da interseção j .

As hipóteses adotadas são:

- $C_j = C$ para todas as interseções j da rede (ciclo comum da rede);
- s_z e $t_{w,z}$ são constantes e conhecidos (para LQ²³ – *Linear Multivariable Feedback Regulator*);
- $\sum g_{j,i} + L_j = C$ (1) para todas as interseções j (a soma dos tempos de verde dos estágios mais o tempo perdido tem que ser igual ao tempo de ciclo da rede); e
- $g_{j,i} \geq g_{j,i\min}$ (2).

²³ A teoria do controle ótimo está preocupada com a operação de um sistema dinâmico a um custo mínimo. O caso em que a dinâmica do sistema é descrita por um conjunto de equações diferenciais lineares e o custo é descrito por uma função quadrática é chamado de problema LQ. Um dos principais resultados da teoria é que a solução é fornecida pelo regulador linear-quadrático (LQR), um controlador realimentado.

O LQR pode ser executado repetidamente com um horizonte de retrocesso; esta é uma forma de controle preditivo do modelo, conforme https://en.wikipedia.org/wiki/Linear%E2%80%93quadratic_regulator.

No LQ s_z e $t_{w,z}$ são constantes, mas podem ser variáveis com o tempo para QPC – *Open-Loop Constrained Quadratic Optimal Control Problem* e NOC – *Open-Loop Constrained Nonlinear Optimal Control Problem*.

Considere agora um *link* z que conecta duas interseções M e N , tal que z é um *link* de saída de M (O_M) e z é um *link* de entrada de N (I_N), ver a Figura 31.

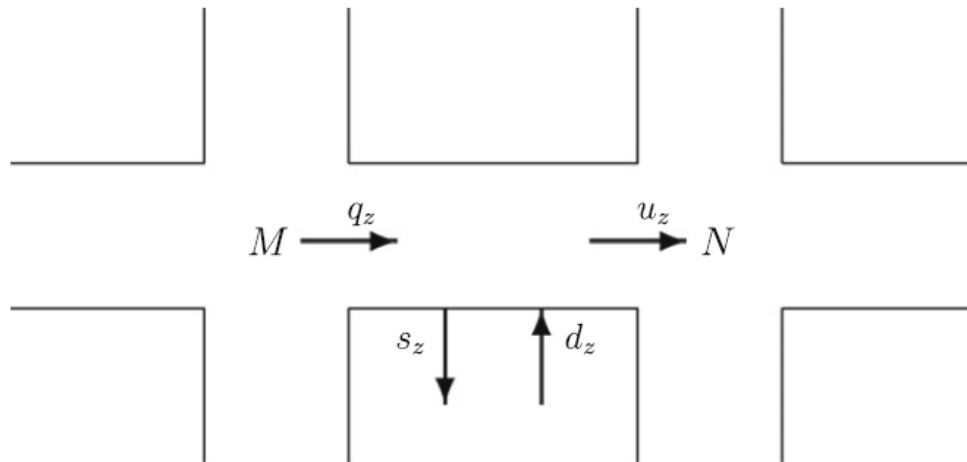


Fig. 1. An urban road link.

Figura 31 – Link z que conecta as interseções M e N
 Fonte: Figura extraída do artigo de Aboudolas et al (2009) [63]

A dinâmica do *link* z é dada pela equação de conservação de fluxos:

$$x_z(k + 1) = x_z(k) + T[q_z(k) - s_z(k) + d_z(k) - u_z(k)] \quad (3)$$

Onde:

- $x_z(k)$ = número de veículos no *link* z no instante kT (fila);
- $x_z(k + 1)$ = número de veículos no *link* z no instante $(k + 1)T$ (fila);
- $q_z(k)$ = fluxo que efetivamente entra no *link* z no intervalo $[kT, (k + 1)T]$;
- $u_z(k)$ = fluxo de saída do *link* z no intervalo $[kT, (k + 1)T]$;
- $d_z(k)$ = fluxo de demanda do *link* z ; e
- $s_z(k)$ = fluxo que efetivamente sai do *link* z .

Fazendo $T = C$ e substituindo na equação de conservação de fluxos (1):

$$q_z(k) = \sum t_{w,z} u_w(k)$$

$$u_z(k) = G_z(k) S_z / C \text{ (do modelo "store-and-forward")}$$

Onde:

$$G_z(k) = \sum g_{j,i}(k) = \text{tempo de verde do link } z \text{ na interseção } j.$$

Chega-se à seguinte equação vetorial:

$$\mathbf{X}(k + 1) = \mathbf{X}(k) + \mathbf{B}\mathbf{g}(k)T\mathbf{d}(k) \quad (4)$$

Onde as letras em negrito representam vetores:

$\mathbf{X}(k)$ = vetor de número de veículos de cada *link* z da interseção j ;

$\mathbf{g}(k)$ = tempos de verde de todos os *links* z da interseção j ;

$\mathbf{d}(k)$ = vetor com os fluxos de demanda d_z de cada *link* z da interseção j ; e

\mathbf{B} = matriz contendo as taxas de conversão e os fluxos de saturação da interseção j .

Adotando LQ – *Linear Multivariable Feedback Regulator*, é assumido um conjunto de planos de tempos fixos para toda a rede (tempos de verde), designado como \mathbf{g}^N . Pode-se mostrar que a Expressão (2) pode ser dada por:

$$\mathbf{g}(k) = \mathbf{g}^N - \mathbf{L}\mathbf{X}(k) \quad (5)$$

onde \mathbf{L} é a matriz de ganho de realimentação (*feedback*).

Essencialmente, o regulador (3) modifica em tempo real o conjunto de planos de tempo fixo \mathbf{g}^N em função da formação de filas $\mathbf{X}(k)$ nos *links* da rede.

O método LQ não prevê o atendimento às hipóteses (1) e (2). Por isso, é usado um algoritmo após a aplicação de (5). Dado um $g_{j,i}$ resultante da Expressão (5), a otimização do *split* visa determinar um $\tilde{g}_{j,i}$ tal que minimiza a função:

$$\phi(\tilde{g}_{j,i}) = \frac{1}{2} \sum_i \left(\frac{(\tilde{g}_{j,i} - g_{j,i})^2}{g_{j,i}} \right) \quad (6)$$

A aplicação em tempo real do método LQ precisa apenas de uma medida de $\mathbf{X}(k)$ por ciclo, baseado na qual é executada a Expressão (5), aplica as hipóteses (1) e (2) por meio de (6) e retorna novos valores de tempos de verde $\mathbf{g}(k)$ para aplicação no próximo ciclo.

Em condições não saturadas, o vetor \mathbf{g}^N pode ser continuamente atualizado em tempo real usando a fórmula de *Webster*:

$$g_{j,i} = \frac{d_{j,i}/s_{j,i}}{\sum_i (d_{j,i}/s_{j,i})} (C - L_j)$$

Onde:

$d_{j,i}$ = demanda do *link* mais carregado do estágio i na interseção j ; e

$s_{j,i}$ = fluxo de saturação do *link* mais carregado do estágio i na interseção j .

A fórmula de *Webster* substitui a regra do LQ (Expressão (6)) em situações de não saturação, consistindo numa estratégia de controle híbrido.

8.7 Otimização de tempo de ciclo

De acordo com Smaragdís et al (2003) [64]:

O objetivo da otimização do tempo de ciclo deve ser aumentar a capacidade das interseções tanto quanto possível para limitar a saturação máxima observada a nível de rede. Dentro do TUC, esse objetivo é realizado em três etapas, por meio da aplicação de um algoritmo simples de *feedback* que usa como critério para o aumento ou diminuição do tempo de ciclo, o nível de saturação máximo corrente de uma percentagem pré-especificada dos *links* da rede.

- Uma percentagem $p = \sigma_{cr}$ pré-especificada dos *links* da rede com máxima “carga” $\sigma_z(k) = x_z(k)/x_{z,max}$ é identificada e a média das correspondentes “cargas” (máxima carga média), $\sigma(k)$, é então calculada;
- O tempo de ciclo da rede é calculado com base na lei de controle de *feedback* (*P-regulator*):

$$C(k) = C^N + K^C(\sigma(k) - \sigma^N)$$

Onde:

C^N = tempo de ciclo nominal da rede (por exemplo, igual a C_{min} = tempo de ciclo mínimo da rede);

σ^N = carga média nominal (por exemplo, igual a zero); e

K^C = parâmetro de controle, valor que afeta a intensidade das variações.

- Se o tempo de ciclo resultante $C(k)$ for suficientemente alto enquanto todos os *links* tiverem níveis de saturação suficientemente baixos, isto é, a sua carga $\sigma_z(k)$ for menor que o valor pré-estabelecido σ_{cr} , então as interseções com menor saturação poderão rodar com tempo de ciclo igual a $C(k)/2$.

Conforme a apresentação de Papageorgiou (s.d.) [61], a expressão completa para a otimização do tempo de ciclo é:

$$C(k) = \begin{cases} C^N + K_1^C[\sigma(k) - \sigma^N], & \text{se } \sigma(k) \leq \sigma_{cr} \\ C^N + K_1^C[\sigma_{cr} - \sigma^N] - K_2^C[\sigma(k) - \sigma_{cr}], & \text{se } \sigma(k) > \sigma_{cr} \end{cases}$$

Onde:

K_1^C e K_2^C = Parâmetros de controle da rede.

De acordo com o artigo de Kouvelas et al (2011) [65], a expressão para o tempo de ciclo é:

$$C_{tc} = \begin{cases} C^N + K_1(\sigma_{tc} - \sigma_{N1}), & \text{se } \sigma_{tc} \leq \sigma_{cr} \\ C^N - K_2(\sigma_{tc} - \sigma_{N2}), & \text{se } \sigma_{tc} > \sigma_{cr} \end{cases}$$

K_1 e K_2 são parâmetros de projeto de toda a rede, sendo que a escolha destes valores afeta a intensidade de reação do módulo de controle de tempo de ciclo. Valores altos de K_1 e K_2 forçam a lei de controle a reagir mais intensamente mesmo para pequenas diferenças de σ_{tc} e de σ_{Ni} , $i = \{1, 2\}$.

8.8 Otimização de offset

De acordo com Smaragdís et al (2003) [64]:

A especificação de defasagem deve idealmente levar em conta a possível existência de filas de veículos no *link* e isso é de fato levado em conta no TUC. TUC realiza controle de *offset* de forma descentralizada, ou seja, para sucessivos pares de interseções. Para cada par de interseções, a especificação de defasagem altera o início de abertura do verde do estágio principal da interseção.

Considere duas interseções sucessivas j_1 e j_2 e o *link* z que as conecta levando o fluxo de j_1 para j_2 . O *link* z tem um comprimento l_z e v_z é a velocidade livre nesse *link*. Se o número de veículos no *link* z for zero, o *offset* entre as duas interseções deve ser igual ao tempo de percurso l_z/v_z , ou seja, o verde do estágio principal da interseção a jusante deve iniciar l_z/v_z segundos mais tarde do que na interseção a montante (*offset* positivo). À medida que o número de veículos dentro de z aumenta, o *offset* deve diminuir de acordo com o tamanho da fila, de modo a permitir uma descarga parcial da fila do *link*. Então o verde da interseção a jusante deve começar mais cedo do que no caso de a fila ser zero, e, em alguns casos pode até ter que começar antes da interseção a montante (*offset* negativo).

O objetivo do TUC é obter um *offset* tal que, na abertura de verde da interseção a jusante, o pelotão que partiu da interseção a montante alcance a cauda da fila formada quando essa fila estiver em movimento e que quando essa cauda estiver passando na linha de retenção o pelotão vindo de j_1 passe logo a seguir. Considere a Figura 32.

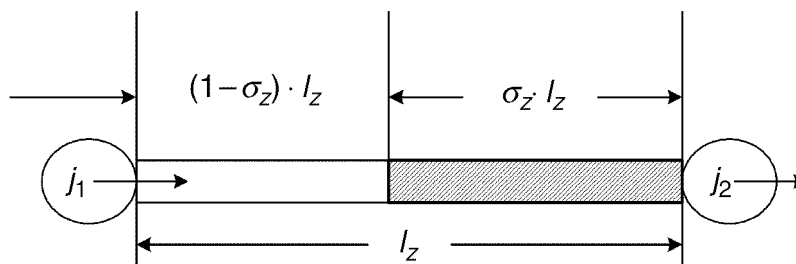


Figura 32 – Determinação do offset de j_1 para j_2
 Fonte: Figura adaptada da apresentação de Papageorgiou (s.d.) [61]

O pelotão que sai de j_1 move-se em direção a j_2 a uma velocidade livre v_z , indo alcançar a cauda da fila no instante $[1 - \sigma_z(k)]l_z/v_z$ após a mudança para verde na interseção j_2 .

A fila do *link* escoa na velocidade v^c , velocidade quando o fluxo é igual ao fluxo de saturação (normalmente em torno de 15 km/h). Logo, o tempo para escoar essa fila é $\sigma_z(k)l_z/v^c$.

Dessa forma, a defasagem t_{j_1,j_2} de j_1 para j_2 deve satisfazer a seguinte condição:

$$t_{j_1,j_2} = \frac{[1 - \sigma_z(k)]l_z}{v_z} - \frac{\sigma_z(k)l_z}{v^c}$$

Então, obtém-se a regra de controle de *feedback*:

$$t_{j_1, j_2}(k) = \frac{l_z}{v_z} - l_z K_z^0 \frac{x_z}{x_{z, max}}$$

Onde:

$$K_z^0 = \frac{v^c - v_z}{v^c v_z}$$

Se a via for de mão dupla, há uma segunda defasagem t_{j_2, j_1} , de j_2 para j_1 . Então, o *offset* $t_{1,2}$ pode ser calculado pela média ponderada entre os dois sentidos:

$$t_{1,2}(k) = W_{j_1, j_2} t_{j_1, j_2}(k) + W_{j_2, j_1} t_{j_2, j_1}(k)$$

Onde:

W_{j_1, j_2} e W_{j_2, j_1} são pesos com $W_{j_1, j_2} + W_{j_2, j_1} = 1$.

O sistema pode selecionar automaticamente o sentido de maior “carga” fazendo o seu peso ser igual a 1 e o peso do sentido oposto ser zero.

8.9 Considerações

Com relação ao posicionamento dos detectores, parece ser bastante estranha a possibilidade de local o detector em qualquer posição do *link*.

Existem 3 situações possíveis:

(1) Se o detector estiver próximo da linha de retenção, a fila formada durante o período de vermelho ficará estacionada em cima do detector. A ocupação medida pelo detector no período de vermelho não apresenta nenhum significado e o fluxo medido será zero. Nesse caso, o detector só é útil para medir o fluxo de saída durante o verde. Não se tem nenhuma ideia da demanda ou do fluxo de chegada.

(2) O detector está longe o suficiente para que a fila formada durante o período de vermelho não fique em cima do detector. Neste caso, a ocupação medida pelo detector reflete a demanda ou o fluxo de chegada.

(3) O detector está situado numa posição intermediária, onde há ciclos em que a fila formada durante o período de vermelho fica em cima do detector e há ciclos em que a fila não chega ao detector. Neste caso, os dados do detector podem ter significados diferentes a cada ciclo. Inclusive, a posição recomendada é no meio do link, que pode coincidir com esse caso.

Fica a questão de como o sistema consegue equalizar os dados dos detectores, com significados tão díspares (principalmente no caso (3)) que podem ser variáveis a cada ciclo?

Com relação à otimização do *split*, no modelo *store-and-forward* são adotadas duas hipóteses simplificadoras, que, a princípio, parecem ser críticas: (1) no modelo é suposto que existe sempre uma fila durante todo o período de verde, isto é, o fluxo de saída é sempre igual ao fluxo de saturação durante todo o período de verde; (2) no modelo é suposto que sempre há espaço suficiente a jusante do cruzamento. A hipótese (1) só se verifica em períodos de congestionamento (sem obstrução a jusante); fora do congestionamento, a hipótese (1) não é verdadeira. A hipótese (2) só ocorre em períodos de baixo trânsito. Em períodos saturados, a hipótese (2) não é verdadeira.

Fica a dúvida se o não atendimento sistemático dessas hipóteses não prejudica a acurácia do modelo, tanto em situações não saturadas como em situações de congestionamento.

Ainda com relação à otimização do *split*, há a necessidade de conhecimento prévio dos fluxos de saturação e taxas de conversão. Os fluxos de conversão são variáveis ciclo a ciclo, da mesma forma como o fluxo que segue em frente. Mesmo o fluxo de saturação pode ser variável. Em princípio, parece que isso pode prejudicar a acurácia do sistema numa operação em tempo real, onde se busca a melhor temporização a cada ciclo.

Com relação à otimização do tempo de ciclo, a otimização é feita para a rede (tempo de ciclo da rede). Entretanto, na rede pode haver um par de interseções que, devido à distância e aos volumes apresentados, não há nenhum benefício em haver coordenação entre elas, sendo melhor terem tempos de ciclo independentes. À primeira vista, o sistema não prevê essa situação, uma vez que o tempo de ciclo não é calculado de forma individual, mas, sim, para toda a rede.

Com relação à otimização do *offset*, da forma como é feita conforme foi descrita no item 8.8, não necessariamente se alcança o melhor resultado, por exemplo, o resultado de menor atraso possível.

Tanto na otimização do tempo de ciclo como na do *offset* é utilizada a razão $\sigma_z = x_z/x_{z,max}$ do *link* z , sendo x_z o número de veículos no *link* (fila) e $x_{z,max}$ o número máximo de veículos que o *link* comporta. Ora, devido a posições arbitrárias do detector, é possível que não se tenha nenhuma ideia da demanda e, portanto, da fila. Assim, é possível que o x_z medido pelo detector não representa a fila do *link*.

Finalmente, deve-se observar que na maioria dos artigos analisados, fala-se que o TUC foi desenvolvido para redes congestionadas em larga escala (*Large-Scale Congested Urban Road Networks*). Entretanto, a eficácia do sistema para redes congestionadas de grandes centros urbanos ainda não foi comprovada na prática. De fato, segundo a apresentação de Papageorgiou (s.d.) [61], o sistema foi implantado nas seguintes cidades:

Cidade	População	Número de interseções	Ano
Glasgow	650.000	7	1998
Jerusalem	874.000	24	2002
Tel Aviv	436.000	6	2002
Chania	54.000	23	2003
Southampton	254.000	37	2003
Munich	1,5 milhão	18	2003
Macaé, Mauá, Santos	---	Sem informação	Sem informação

Pode-se observar que o sistema foi implantado apenas em pequenas redes, de no máximo 37 interseções, em cidades de pequeno e médio porte. Não houve ainda a implantação em nenhuma grande metrópole, em redes que apresentam grandes congestionamentos.

9. CONTREAL: Controle de Tráfego por Área em Tempo Real

De acordo com o Relatório da IDOM (2019 – D2) [20]:²⁴

O CONTREAL é uma solução desenvolvida pela empresa BRASCONTROL em colaboração com a Universidade Federal de Santa Catarina e com o Prof. Markos Papageorgiou. É uma solução centralizada que faz mudanças ciclo por ciclo (geração dinâmica). Os requisitos para sua implementação são:

- Que haja sensorização de todas as faixas das vias de acesso, não considerando informações relacionadas aos movimentos de pedestres; e
- Que exista uma latência máxima de rede de 10 segundos, necessária para sua implementação;

Ele pode ser implementado utilizando o protocolo proprietário da BRASCONTROL ou o UTMC2, embora o protocolo proprietário seja o recomendado pela empresa.

Esse modo de operação tem sido implementado em cidades do Brasil, como Maceió, Mauá e Santos.

No site <https://cienciahoje.org.br/sinal-verde-para-os-veiculos/> (acessado em 20/07/2022), pode-se ler a seguinte matéria de Juliana Marques, Ciência Hoje On-line, de 21/01/2009:

Um sistema desenvolvido por pesquisadores brasileiros regula o intervalo de tempo dos semáforos com base no fluxo de veículos nos cruzamentos.

Motoristas que perdem tempo em semáforos mal ajustados poderão contar com uma tecnologia nacional para resolver esse problema. Um sistema criado por pesquisadores da Universidade Federal de Santa Catarina (UFSC) regula o tempo dos semáforos com base na contagem de veículos que passam pelos cruzamentos, o que pode melhorar o tráfego em alguns pontos das grandes cidades.

Diversos sistemas importados semelhantes ao brasileiro – batizado de Controle de Tráfego por Área em Tempo Real (CONTREAL) – já são utilizados em várias capitais para facilitar o fluxo de veículos e, assim, diminuir o tempo do percurso, os gastos com combustível e os danos ao meio ambiente. As vantagens do CONTREAL são o baixo custo e a facilidade para o fornecimento de suporte local.

Em testes na cidade de Macaé, no Rio de Janeiro, o CONTREAL tem mostrado bons resultados em relação ao sistema convencional – que se baseia em horários críticos, com altos índices de engarrafamentos, para regular o tempo dos semáforos. Em simulações, o uso do CONTREAL reduziu de 15% a 20% o tempo gasto pelos motoristas para percorrer um trajeto, em comparação com o sistema tradicional.

Para fazer o monitoramento do fluxo de veículos, são instalados sensores no pavimento dos cruzamentos e os dados são enviados para um computador. Com base na contagem de veículos e em informações sobre o horário, as condições meteorológicas e a existência de acidentes, obras na pista e outros eventos capazes de interferir no tráfego, um programa de computador desenvolvido pela equipe do departamento de automação e sistemas da UFSC determina o intervalo de tempo mais apropriado para o semáforo.

²⁴ Observe-se que as informações foram coletadas diretamente da empresa.

Segundo o engenheiro eletricitista Werner Kraus Junior, professor da UFSC e coordenador do projeto, a tecnologia do CONTREAL é a mesma usada em aparelhos celulares. A transmissão de dados dispensa fios, pois é feita pelo Sistema Global para Comunicações Móveis (GSM, na sigla em inglês). “Além disso, o CONTREAL conta com o suporte de um serviço de rádio, que permite uma transmissão mais rápida dos dados”, destaca.

Solução conjunta para congestionamentos

Apesar dos bons resultados do uso do CONTREAL na análise de fluxos, Kraus Junior alerta que, sozinho, o sistema não resolve o problema dos congestionamentos. “O monitoramento facilita e organiza o trânsito, porque controla o tempo dos semáforos e detecta aqueles mal ajustados”, justifica. E completa: “Mas somente o investimento em transportes coletivos reduzirá os engarrafamentos substancialmente”.

Com os resultados dos testes em Macaé, Kraus Junior e sua equipe estão organizando um portal na Internet, que vai disponibilizar dados atualizados sobre o fluxo de veículos para qualquer interessado em monitorar as condições de trânsito na cidade, tanto motoristas quanto pesquisadores.

No site <https://brascontrol.com.br/2020/06/23/prefeitura-de-florianopolis-comeca-a-2a-fase-para-a-implantacao-dos-semaforos-inteligentes/> (acessado em 20/07/2022), consta notícia, de 23/06/2020, sobre a implantação de semáforos inteligentes nas avenidas Beira-Mar Norte, Paulo Fontes e Rio Branco, além da rua Padre Roma, pela Prefeitura de Florianópolis. A tecnologia, adquirida junto à empresa paulista BRASCONTROL Indústria e Comércio, foi desenvolvida em parceria com a UFSC (Universidade Federal de Santa Catarina).

Apesar de não ter sido possível localizar outras fontes independentes sobre o sistema, bem como material com informações técnicas sobre os princípios de otimização, o Prof. Werner Kraus, da Universidade Federal de Santa Catarina, em e-mail de 11/08/2022, informa que o sistema CONTREAL foi desenvolvido com base na técnica do TUC e sua primeira implantação foi em Macaé. Aquela versão tinha em seu núcleo o código do TUC. Logo em seguida, o código foi vertido para outra linguagem e o sistema foi aperfeiçoado pela BRASCONTROL, em parceria com a UFSC.

10. SMART GREEN

O pacote SMART GREEN é formado pelos módulos *Smart Intersection* e *Smart Corridor*.

10.1 *Smart Intersection*

De acordo com SMART INTERSECTION (2017) [66]:

O algoritmo de avaliação baseia-se nas informações recebidas dos detectores de tráfego localizados perto da linha de parada, de preferência em cada faixa de tráfego (até 35 m de distância da linha de retenção). Os detectores de 2 a 4 m de comprimento alimentam a avaliação com dados essenciais de tráfego, como: *gaps*, ocupação e volumes de tráfego de cada faixa de tráfego.

Uma análise mais aprofundada dos dados leva à avaliação do fluxo de saturação e capacidade em cada faixa de tráfego.

$$C = S \frac{G}{T}$$

Onde:

C = Capacidade;

S = Fluxo de saturação;

G = Verde efetivo; e

T = duração do ciclo.

A avaliação da capacidade resulta nos valores de duração do ciclo e do *split*.

A Figura 33 mostra o posicionamento dos detectores.



Figura 33 – Posicionamento dos detectores no SMART INTERSECTION

Parâmetros do ciclo:

- Ciclo mínimo;
- Ciclo máximo;
- Limite de mudança de verde (passo máximo de mudança de verde); e
- Tempo perdido (avaliado automaticamente pelo sistema como parte do algoritmo, no entanto, o usuário pode definir o tempo perdido que será usado em vez do valor avaliado).

Parâmetros do estágio:

- Verde mínimo; e
- Verde máximo.

A nova temporização (ciclo e *split*) é implementada no início do estágio em cada ciclo.

10.2 *Smart Corridor*

De acordo com SMART CORRIDOR (2019) [67]:

SMART CORRIDOR é a solução adaptativa de controle coordenado de interseções.

SMART CORRIDOR é uma solução de controle de tráfego dedicado e adaptável para corredores semaforizados coordenados. É um pacote de *software* em *firmware* ITC adicional que decide a duração ideal do ciclo para um corredor inteiro e suporta cruzamentos individuais com a estimativa do *split*.

Com o SMART CORRIDOR, a mudança do ciclo e do *split* é aplicada de forma gradual para manter sempre a coordenação. O sistema também pode lidar com interseções a alguma distância e que devem operar com um ciclo diferente.

SMART CORRIDOR é uma solução modular que permite diferentes modos de controle de coordenação dependendo do sistema de detecção e necessidades do usuário:

- Rede coordenada com ciclo e *split* fixo;
- Rede coordenada com ciclo fixo e com semi-atuação;
- Seleção de situação de tráfego, com coordenação semi-atuada (DOGS); e
- Modo adaptativo para ciclo e *split*, com coordenação semi-atuada (com SMART INTERSECTION operando nas interseções individuais).

O documento não define o que se entende por “semi-atuação” e não fornece mais informações sobre os princípios do sistema.

10.3 *Smart Green*

O pacote “SMART INTERSECTION + SMART CORRIDOR” forma o “SMART GREEN”.

O sistema funciona apenas com os controladores ITC-2 e ITC-3 da SWARCO.

11. INES - *Intelligent Network Control*

11.1 Introdução

INES – *Intelligent Network Control* – é um *software* para controle adaptativo de redes semafóricas, desenvolvido pela empresa alemã *Schlothauer & Wauer* e que se propõe a operar integrado aos sistemas de gerenciamento de tráfego.

A descrição feita no presente trabalho é baseada tão somente em material da própria empresa detentora do sistema. Na pesquisa realizada, não foi possível localizar nenhum documento como artigos, relatórios etc. sobre o sistema INES provenientes de fontes independentes ou acadêmicas.

Assim, basicamente as informações constantes nesta pesquisa são provenientes de empresas detentoras do sistema, a saber:

- Site da empresa *Schlothauer & Wauer* (*Schlothauer & Wauer, s.d.*) [68];
- Vídeo da empresa *Schlothauer & Wauer* (*Schlothauer & Wauer, 2022a*) [69];
- Segundo vídeo da empresa *Schlothauer & Wauer* (*Schlothauer & Wauer, 2022b*) [70]; e
- Site sobre Bogotá da empresa SWARCO (Bogotá, SWARCO) [71].

O sistema INES está atualmente em uso em Hamburgo, Augsburg e Heidelberg, Alemanha. Não se sabe em quantas interseções em cada cidade.

Mais recentemente o sistema foi implantado em Bogotá. De acordo com *Schlothauer & Wauer, (2022a)* [69], o Projeto de Bogotá analisou 1500 interseções e implantou 808 controladores conectados ao INES, divididos em 58 subáreas. A detecção é feita por 1300 câmeras. Estudos realizados apontam redução de atraso de 5 a 19% e uma redução média do tempo de percurso de 10% em relação à situação anterior.

Sobre o projeto de Bogotá, conforme (Bogotá, SWARCO) [71]:

O projeto de Bogotá inclui a renovação dos controladores de tráfego, entrega de software e planejamento do sistema de gerenciamento de tráfego em toda a cidade de Bogotá. A SWARCO está encarregada de entregar um terço – uma tarefa importante em uma cidade tão grande como Bogotá, com cerca de 7,5 milhões de cidadãos.

Entregamos um novo *hardware* que cria uma solução melhor para controle de tráfego. Com o novo *hardware* podemos replanejar o controle de tráfego e instalar sistemas de controle de rede e mais de 2.000 detectores de câmeras que entendem o fluxo de tráfego. Esta informação *online* pode alterar os tempos de verde e controlar as ondas verdes.

O projeto começou em 1º de setembro de 2018. Um ano depois, 50% dos controladores estavam instalados. O prazo previsto para a conclusão do projeto é setembro de 2020.

SWARCO e *Schlothauer & Wauer* fundaram uma nova empresa com sede em Bogotá chamada SWARCO ANDINA. Esta empresa é a parceira local em Bogotá e foi ela que entregou os controladores.

Segundo o site da empresa (Schlothauer & Wauer, s.d.) [68]:

INES é um sistema adaptativo de controle de rede que é independente de fabricantes. É um sistema flexível e suas interfaces abertas garantem uma fácil integração com os sistemas de gerenciamento de tráfego existentes. A estrutura modular e de fácil entendimento do algoritmo de controle do INES pode ser adaptada a uma ampla gama de necessidades específicas. A interface de usuário baseada na web facilita a operação do controle da rede.

INES melhora o desempenho do controle adaptativo em nível local ao identificar e realocar os tempos de verde não utilizados. INES garante que seja fornecido tanto tempo de verde quanto é necessário.

INES otimiza o *offset* de redes coordenadas. Ao alterar os parâmetros específicos do projeto em tempo de execução, INES permite um grande número de soluções individuais para otimizar o tráfego.

Embora o texto informe que o sistema é “independente de fabricantes”, deve-se observar que a estrutura adotada é a do protocolo OCIT. Assim, o mais correto seria afirmar que o sistema é independente de fabricantes, desde que adotem o protocolo OCIT.

Ainda no site da empresa (Schlothauer & Wauer, s.d.) [68], é informado que existem dois tipos de detectores: junto à linha de retenção e detectores avançados. Presume-se que os detectores avançados são os detectores estratégicos (ver o item 11.2.8).

11.2 Análise do vídeo (Schlothauer & Wauer, 2022a) [69]

Esse vídeo foi encaminhado pela empresa Schlothauer & Wauer em resposta a um e-mail solicitando informações sobre o INES.

11.2.1 Cenário e capacidade

Considere a Figura 34.

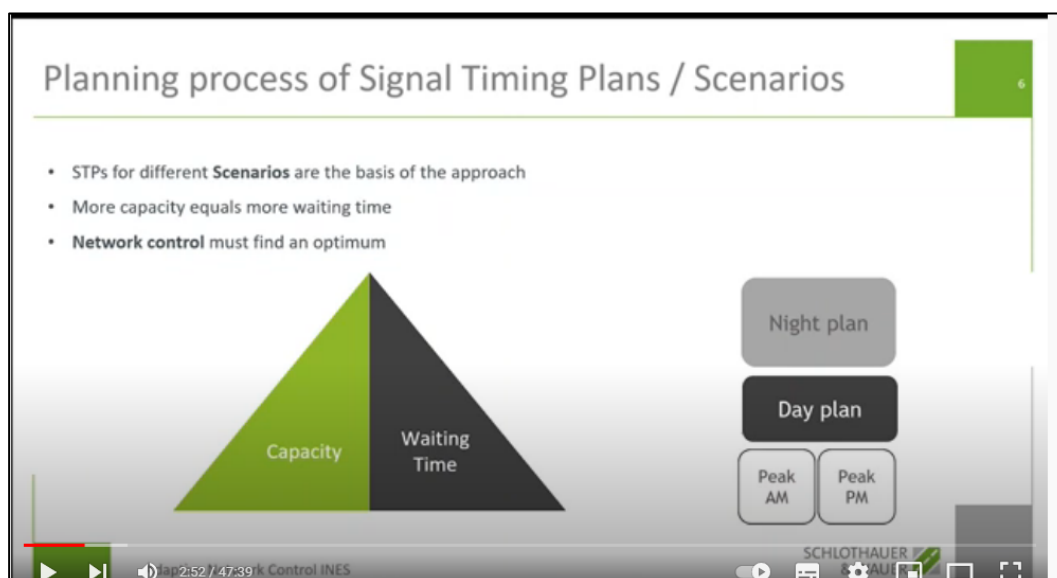


Figura 34 – Cenários e capacidade
Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Na Figura 34 é dito que o sistema é baseado em planos semafóricos (STP = *Signal Timing Plan*) para diferentes cenários. No entanto, o vídeo não explica o que é “cenário”. No site (Schlothauer & Wauer, s.d.) [68], pode-se ler que um “cenário” consiste em uma solicitação de plano, uma estrutura de coordenação e, eventualmente, uma seleção adequada de parâmetros de priorização (por exemplo, de transporte público). Por essa definição de cenário, já se permite presumir que o sistema é baseado em seleção de planos.

A Figura 34 ainda mostra que “mais capacidade é igual a mais tempo de espera”. Essa afirmação soa estranha uma vez que mais capacidade significa maior fluidez e, por consequência, menor tempo de espera. A capacidade de uma aproximação semaforizada é dada por:

$$Capacidade = \frac{g}{C} \times F_S$$

Onde:

g = tempo de verde durante o qual o fluxo é igual ao fluxo de saturação;

C = tempo de ciclo; e

F_S = fluxo de saturação.

Pela expressão, pode-se observar que a capacidade aumenta com o aumento do tempo de verde, desde que o fluxo seja igual ao fluxo de saturação **durante todo o período de verde**. Ora, se o tempo de verde for excessivo, haverá verde desperdiçado ou não aproveitado, o que irá gerar aumento de atraso pois esse tempo de verde poderia ter sido aproveitado para os movimentos concorrentes. Mas, neste caso, o tempo de verde não aproveitado não gera aumento de capacidade, pois o fluxo não será igual ao fluxo de saturação durante todo o período de verde. Dessa forma, não foi possível entender o significado da afirmação de que aumento de capacidade é igual a aumento do tempo de espera.

11.2.2 Arquitetura do INES

No site (Schlothauer & Wauer, s.d.) [68] é apresentado o esquema da Figura 35.

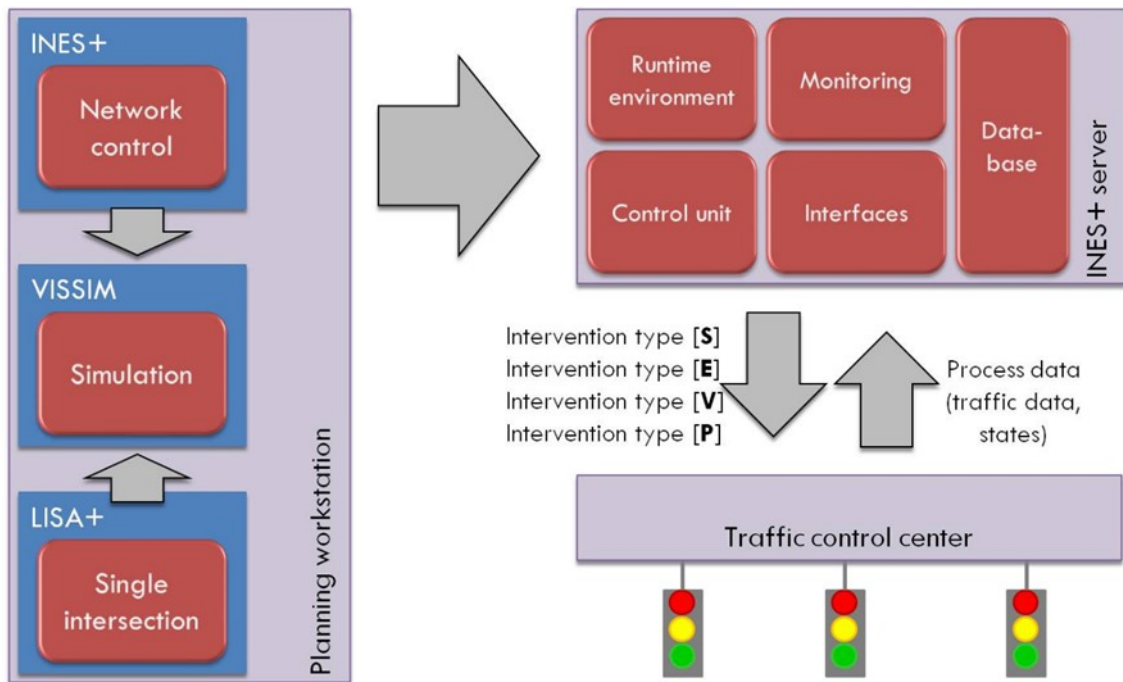


Figura 35 – Esquema básico do INES

Fonte: Figura extraída de Schlothauer & Wauer (s.d.) [68]

Embora o diagrama da Figura 35 mostre VISSIM e LISA como componentes do sistema, a matéria constante no site (Schlothauer & Wauer, s.d.) [68] não detalha qual é a função ou contribuição de cada um, além dos termos "Simulation" para o VISSIM e "Single intersection" para o LISA, sem especificar ou detalhar o que significam.

Ainda no site da empresa (Schlothauer & Wauer, s.d.) [68] pode-se ler:

Uma interface para a ferramenta de simulação VISSIM foi implementada para permitir os testes de controle do INES. Isso permite que os usuários, no estágio de planejamento, sigam com precisão todas as etapas e decisões tomadas pelo algoritmo. A capacidade do controle central de intervir no nível local é integrada diretamente no painel de controle do ambiente de simulação (computador de tráfego emulado). Usando uma interface WEB, o controle central pode ser monitorado e executado exatamente como seria na operação real. Isso permite que alterações sejam feitas e análises sejam executadas enquanto a simulação está sendo executada. Testar os controles antes de implementá-los reduz o tempo e o esforço gastos para ajustá-los durante a operação real.

Por outro lado, o vídeo (Schlothauer & Wauer, 2022a) [69] (ver a Figura 36) mostra o LISA com a função de "Local planning", sendo que o apresentador do vídeo fala apenas que se trata de uma ferramenta complementar, sem detalhar exatamente qual é o papel do LISA no sistema. Além disso, o vídeo não menciona VISSIM.

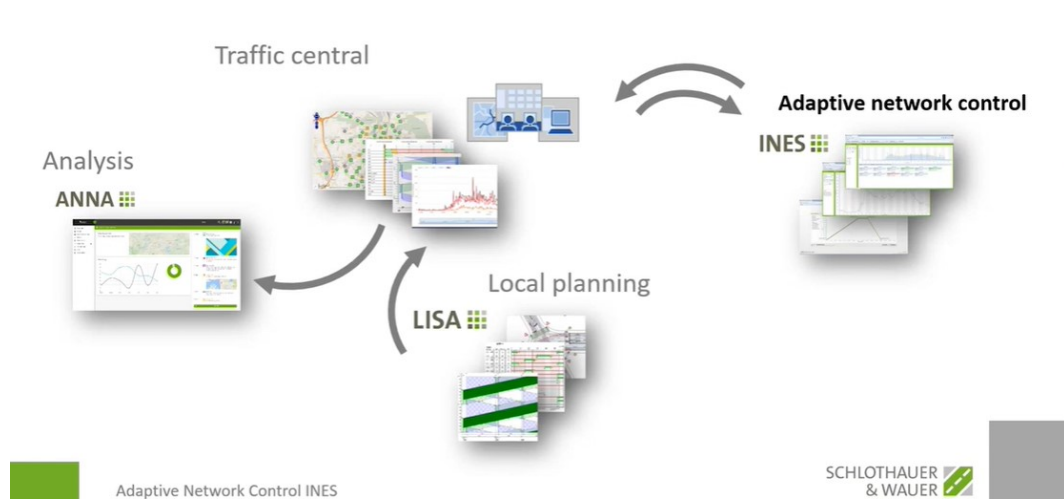


Figura 36 – Componentes do sistema
 Fonte: Figura extraída de (Schlothauer & Wauer, 2022a) [69]

O vídeo de Schlothauer & Wauer (2022a) [69] apresenta o diagrama da Figura 37.

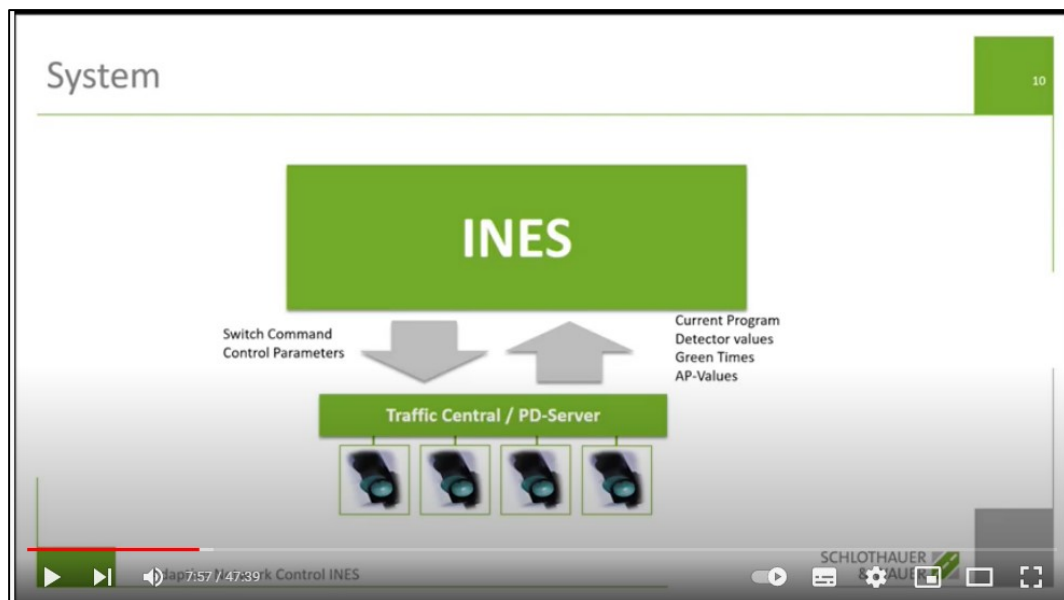


Figura 37 – Arquitetura do INES
 Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Na Figura 37 verifica-se que a Central fornece ao INES, além dos dados de detectores e de tempos de verde, os AP-Values.

Conforme a definição dada no OCIT Outstations Version 3.0 – Glossary (2018) [25], AP-Value é: “Termo genérico para designar variáveis internas dos controladores que são dinamicamente calculadas por programas de usuário ou que (se configurável) podem ser modificadas dinamicamente por aplicativos centrais de nível superior para programas de controle.” Talvez por “programas de usuário” (*user programs*) entenda-se programas complementares como o LISA, por exemplo.

11.2.3 Módulos do INES

O vídeo Schlothauer & Wauer (2022a) [69] apresenta como módulos do INES (ver a Figura 38), com as correspondentes sub-etapas:

- Seleção de cenários:
 - ✓ *Saturation*;
 - ✓ *Loss Time*; e
 - ✓ *Strategics*.
- Redistribuição de tempos de verde:
 - ✓ *Greentime*; e
 - ✓ *Offset*.
- Green Corridor:
 - ✓ *Manage Congestion*.

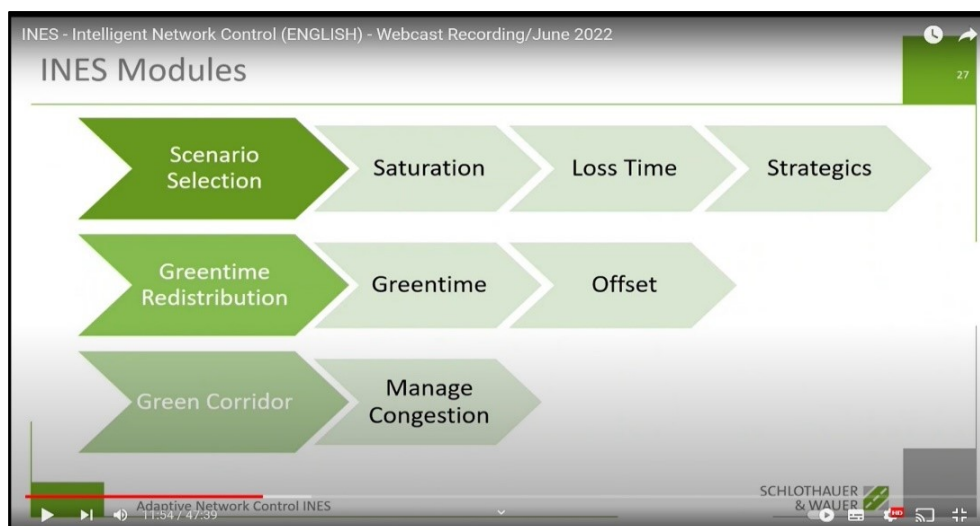


Figura 38 – Módulos do INES

Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Esses “módulos” parecem não ser propriamente módulos. Talvez uma designação melhor fosse “etapas”.

Por outro lado, o site Schlothauer & Wauer (s.d.) [68], apresenta como “módulos” 4 “*Intervention types*” (ver a Figura 35), conforme a descrição a seguir:

Tipo E – Neste módulo o INES altera o parâmetro para o tempo máximo de verde e, portanto, o fim do verde, para um ou mais grupos semafóricos. Isso modifica e redistribui as reservas de verde à disposição para resposta ao fluxo de tráfego na interseção.

Tipo V – Aqui o INES fornece parâmetros calculados para otimizar o fluxo de tráfego nos principais eixos coordenados em resposta a mudanças nas condições de tráfego ou modificações feitas usando o tipo E.

Tipo S - O INES seleciona um cenário para a rede, que consiste em demanda de plano para as interseções, uma estrutura de coordenação e possivelmente uma seleção adequada de parâmetros de priorização, tipos de intervenção P.

Tipo P – Por fim o INES altera um parâmetro no controle local (por exemplo, o nível de priorização do transporte público). As intervenções são definidas no estágio de planejamento e permitem que sejam feitas alterações precisas e individuais no controle enquanto ele está em execução.

As ações ou “intervenções” descritas acima parecem estar relacionadas com os módulos da Figura 38. Entretanto, não fica claro qual é a conexão exata entre os módulos da Figura 38 com as intervenções da Figura 36.

11.2.4 Procurando o cenário ótimo

Para determinar qual é o cenário ótimo, a primeira tarefa é o cálculo da saturação (ver a Figura 39a).

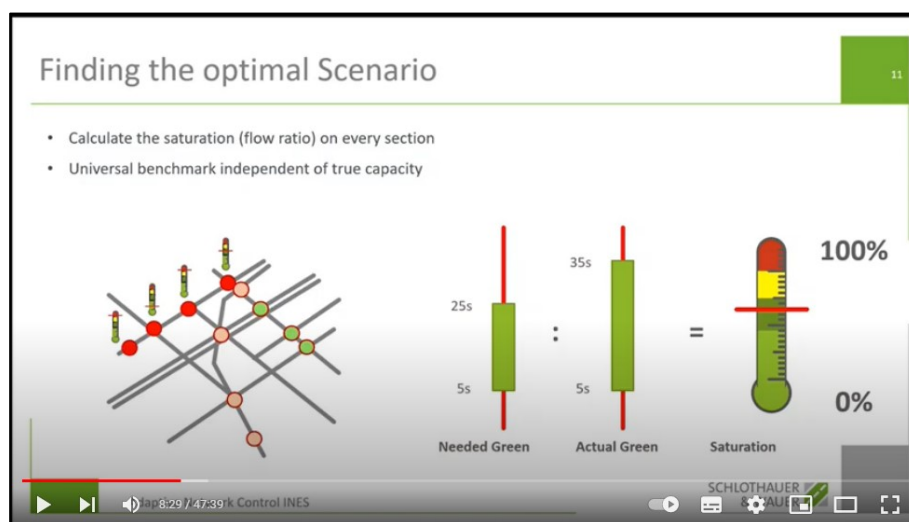


Figura 39a – Procurando o cenário ótimo
Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

No vídeo não é explicado o que é “saturação” ou razão de fluxos (*flow ratio*). No glossário do LISA Manual (s.d.) [72] não consta o termo “*saturation*”, mas consta a definição de “*flow ratio*”: razão entre o fluxo e a capacidade. Ora, a razão entre o fluxo e capacidade é o grau de saturação. De fato:

$$\text{Saturação} = \frac{F}{Cap} = \frac{F}{\frac{g}{C} F_s} = \frac{F \times C}{g \times F_s} = x$$

Onde:

F = fluxo;
 Cap = capacidade;
 g = tempo de verde;
 C = tempo de ciclo;
 F_s = fluxo de saturação; e
 x = grau de saturação.

Portanto, por “saturação” pode-se entender como grau de saturação.

No OCIT Outstations Version 3.0 – Glossary (2018) [25] não consta a definição de “saturation” e nem de “flow ratio”.

Não se conseguiu descobrir o que é “Universal benchmark” e tampouco o que é “true capacity”.

Também não fica claro o que significa “Needed green” e “Actual green”. Se “Actual green” for o tempo de verde disponível, dado pelo plano semafórico (STP – *Signal Timing Plan*) e “Needed green” for o tempo de verde efetivamente necessário, a saturação seria obtida pela razão “Needed green” / “Actual green”? No exemplo da Figura 36a, a saturação seria calculada como sendo a razão $(25 - 5)/(35 - 5) = 20/30 = 67\%$? Nesse caso, como seria obtido ou calculado o tempo de verde efetivamente necessário (“Needed green”) ? Seria o tempo de verde até atingir o *gap out* na atuação? Contudo, neste caso, a saturação não poderia ser maior que 1 uma vez que o “Needed green” não poderia ser maior que o “Actual green”. Tendo em vista que os detectores ficam próximos à linha de retenção e não se tem informação da demanda ou do fluxo de chegada, não fica claro como o sistema faz para saber quanto mais verde é necessário nos casos em que a fila não é totalmente escoada no ciclo corrente, havendo filas residuais, isto é, quando na atuação for usado todo o tempo de verde disponível, não havendo o *gap out*.

Já do vídeo original, em alemão²⁵, pode-se encontrar as expressões “freigabe benötigt” ou “liberação necessária” e “freigabe vorhanden” ou “lançamento disponível”. O que dá a entender que o “Actual green” é o verde máximo disponível e o “Needed green” é o tempo de verde demandado em determinado instante.

Agora, se o entendimento de que o “Needed green” é determinado pela atuação estiver equivocado, então não é possível cogitar o que seja, de fato, o “Needed green” e como ele é obtido.

Por outro lado, se a saturação for obtida pela razão “Needed green” / “Actual green”, então o conceito de saturação não é mais compatível com a definição de “flow ratio” dada pelo LISA Manual (s.d.) [72] (razão entre o fluxo e capacidade) e, portanto, com o conceito de grau de saturação.

A Figura 39b a seguir mostra que apenas a saturação não é suficiente para determinar o cenário ótimo. Saturação muito alta significa congestionamento, enquanto uma saturação muito baixa significa verde ocioso e, portanto, aumento do atraso. Esse raciocínio está compatível com o entendimento de que por saturação se entende como grau de saturação.

²⁵ Fonte: <https://www.youtube.com/watch?v=qxgPK5Fdws0> (Acesso em 18/11/2022).

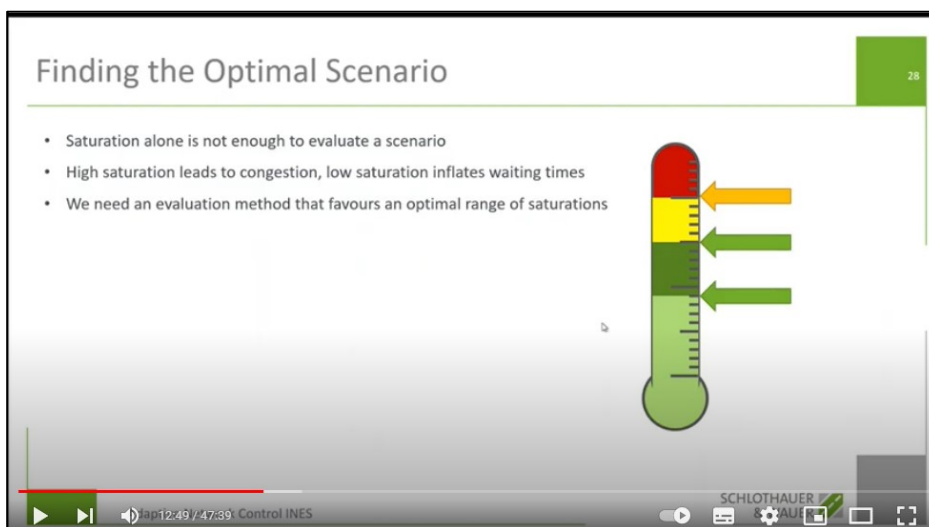


Figura 39b – Procurando o cenário ótimo
 Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Então, é necessário um método para achar uma faixa de valores de saturação que evite congestionamento e ociosidade.

11.2.5 Calculando o grau de saturação (Saturation)

Considere a Figura 40.

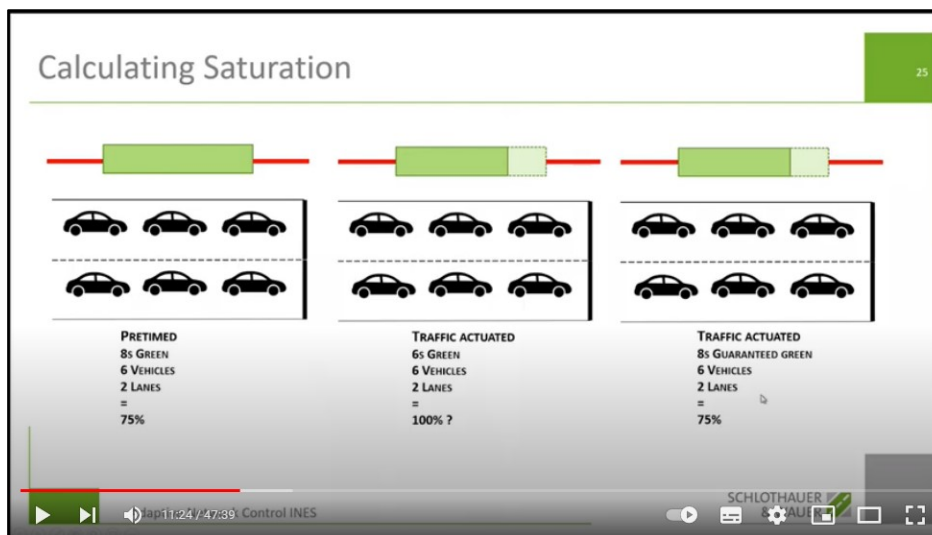


Figura 40 – Calculando saturação
 Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Acredita-se que nos exemplos da Figura 40, foi suposto um fluxo de saturação de 0,5 veículos/segundo por faixa. Então, no “Pretimed” ter-se-ia uma saturação de $(3/0,5)/8 = 75\%$. Na atuação sem tempo de ciclo fixo (atuação sem coordenação), ter-se-ia uma saturação de $(3/0,5)/6 = 100\%$. Na atuação com tempo de ciclo fixo, a saturação seria $(3/0,5)/8 = 75\%$. Nesse caso, a expressão “Guaranteed green” seria equivalente ao “Actual green”?

Note que essa forma de calcular a saturação é exatamente o cálculo do grau de saturação:

$$x = \frac{\frac{F}{F_S}}{\frac{g}{C}} = \frac{F \times C}{F_S \times g}$$

Onde:

x = grau de saturação;

F = fluxo;

g = tempo de verde;

C = tempo de ciclo; e

F_S = fluxo de saturação.

Neste caso, g seria o “*Actual green*” ou o “*Needed green*”?

Nesse ponto, fica a dúvida: qual é a relação entre essa forma de cálculo da saturação com a forma de cálculo dada pela razão entre o “*Needed green*” e o “*Actual green*”?

Outra dúvida é: no cálculo de saturação, são mostrados exemplos com valores de “*Needed green*”, “*Actual green*” ou “*Guaranteed green*” em segundos. Entretanto, o cálculo não deveria ser feito em termos de porcentagem de verde em relação ao tempo de ciclo? Afinal, 30 segundos em um ciclo de 80 é bem diferente de 30 segundos em um ciclo de 120. Eventualmente, foram usados valores em segundos (em vez de percentuais em relação ao ciclo) tão somente para ilustrar o raciocínio na apresentação do vídeo.

Contudo, aqui, a dúvida mais importante é: se o “*Needed green*” é determinado pela atuação, então o cálculo é feito ciclo a ciclo e, por consequência, também a saturação. Isso quer dizer que a seleção do plano é feita ciclo a ciclo? É implementado um novo plano a cada ciclo? Não fica claro qual é o intervalo de tempo em que é implementado um novo plano.

11.2.6 Função de Utilidade (*Utility Function*)

A faixa ótima de saturação (que corresponde à região entre o congestionamento e a ociosidade) é determinada por uma função denominada Função de Utilidade, cujo gráfico está mostrado na [Figura 41a](#), onde o eixo horizontal é a saturação.

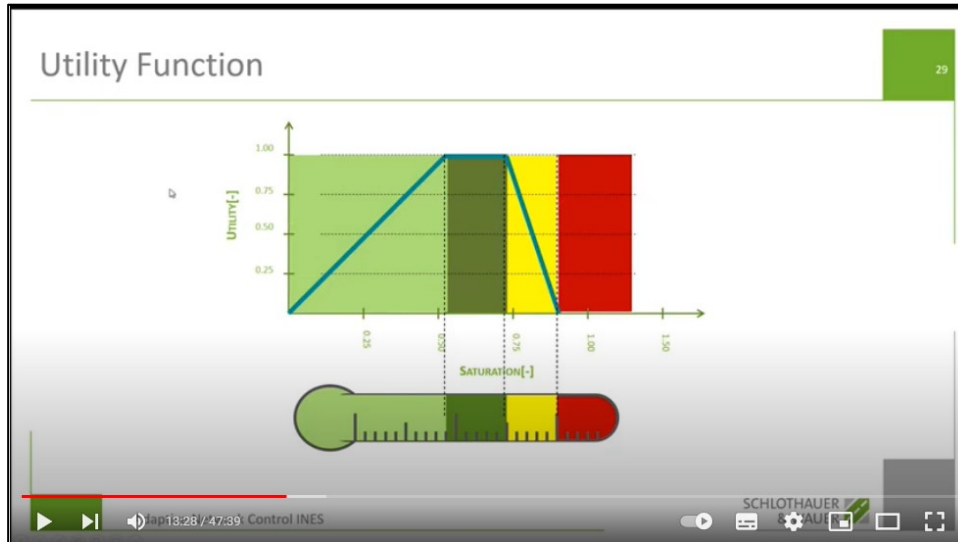


Figura 41a – Função de Utilidade (Utility Function)
 Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

O vídeo não esclarece como essa função é determinada e nem mostra a sua expressão analítica. O que se depreende é que a região verde claro corresponde à região de ociosidade, enquanto a região em amarelo corresponde à situação de congestionamento. A faixa ótima (entre a ociosidade e congestionamento) é a região verde escuro, onde o gráfico apresenta um patamar. Essa seria a faixa de valores ótima, onde a saturação pode aumentar mantendo-se constante o valor máximo da função de utilidade, até o ponto em que a função de utilidade começa a cair, indicando congestionamento. Assim, deve-se escolher um cenário onde a saturação fique nessa faixa de valores (onde os valores da função de utilidade fiquem no patamar horizontal do gráfico).

Um gráfico que tem o mesmo perfil é aquele apresentado no modelo de transmissão celular (*Cell Transmission Model*) conforme o artigo de Daganzo (1994) [73].

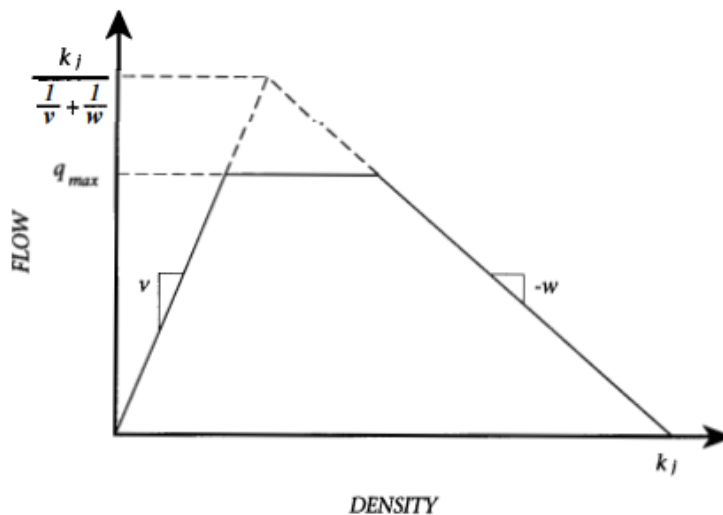


Figura 41b – Relação entre fluxo e densidade do modelo de transmissão celular (Cell Transmission Model)
 Fonte: Figura extraída de Daganzo (1994) [73]

No gráfico da Figura 41b, o eixo horizontal é a densidade, enquanto o eixo vertical é o fluxo. O segmento de reta à esquerda corresponde a uma região de baixa densidade (trânsito livre), enquanto o segmento de reta à direita corresponde a uma região de congestionamento (alta densidade). A região do patamar corresponde a fluxo máximo. Nessa região, a densidade pode aumentar mantendo-se o fluxo constante igual ao fluxo máximo, até o ponto em que o fluxo começa a cair, indicando situação de congestionamento.

Portanto, a interpretação dos dois gráficos (Figura 41a e Figura 41b) é bastante similar.

Existem vários artigos que utilizam funções denominadas Funções de Utilidade, definidas de formas as mais diversas. Por exemplo: no artigo de Tian et al (2008) [74] a Função de Utilidade foi escolhida como uma função do atraso médio dos veículos; no artigo de Le et al (2017) [75] a Função de Utilidade é uma função baseada no coeficiente de Gini ou *Gini Index*²⁶, limitada no intervalo [0, 1], usada para medir a desigualdade do tráfego dentro da rede urbana (heterogeneidade espacial); no artigo de Elouni et al (2021) [76], a Função de Utilidade é representada pela fila de veículos na linha de retenção; e no artigo de Guo et al (2018) [77] a Função de Utilidade foi assim definida: se o atraso médio é aumentado, então a Função de Utilidade é o atraso médio do intervalo anterior menos o atraso médio do intervalo atual (o resultado é negativo, indicando uma falha ou insucesso), se o atraso médio é reduzido ou igual, a Função de Utilidade é igual a 0 (indicando um sucesso).

No caso do INES, não foi possível saber a definição da Função de Utilidade.

Considere agora a Figura 41c.

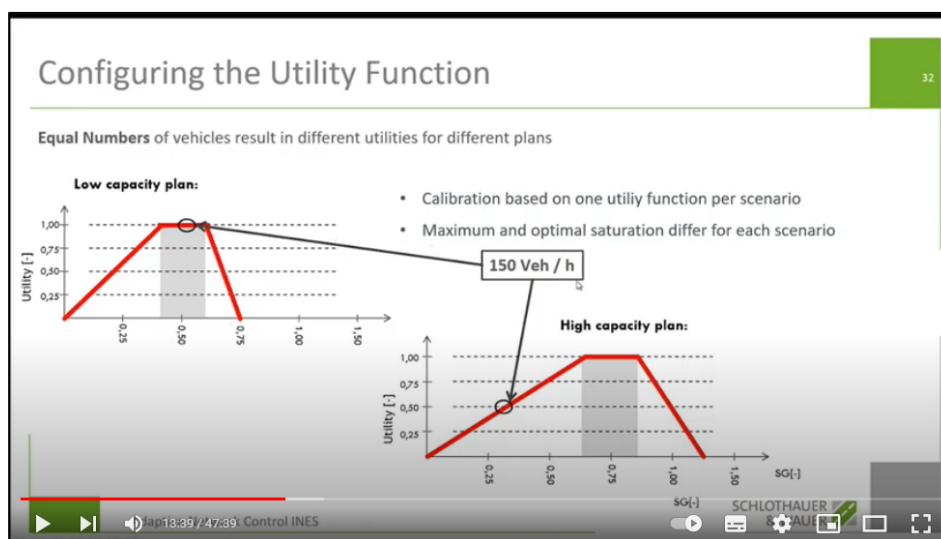


Figura 41c – Configuração da Função de Utilidade
Figura extraída de Schlothauer & Wauer (2022a) [69]

²⁶ Em Economia, o coeficiente de Gini, também conhecido como *Gini Index*, é uma medida da dispersão estatística para representar desigualdades entre valores de uma distribuição de frequência. Foi desenvolvido por Corrado Gini para medir a desigualdade de renda ou da saúde de uma nação. Um valor igual a 0 significa perfeita igualdade, enquanto um valor igual a 1 expressa máxima desigualdade. Fonte: https://en.wikipedia.org/wiki/Gini_coefficient (Acesso em 07/11/2022).

Na Figura 41c é possível ler que um fluxo igual pode resultar em diferentes valores da Função de Utilidade. No exemplo da Figura 41c, um fluxo de 150 veículos/hora corresponde ao valor 1 na Função de Utilidade para uma saturação de 0,50 em um plano de baixa capacidade, enquanto para o mesmo fluxo em um plano de alta capacidade representa um valor de 0,50 para Função de Utilidade e uma saturação entre 0,25 e 0,50.

Parece estranho fazer essa comparação com base apenas em fluxo, pois 150 veículos/hora em uma faixa é totalmente diversa que 150 veículos/hora em 3 faixas, por exemplo. Parece ser mais correto fazer a comparação com base na taxa de ocupação (razão entre fluxo e fluxo de saturação). No exemplo, em vez de 150 veículos/hora poderia ser 150 veículos/hora por faixa, por exemplo.

11.2.7 Atraso (*Loss Time*)

O atraso (*delay*) é designado aqui como *loss time*.

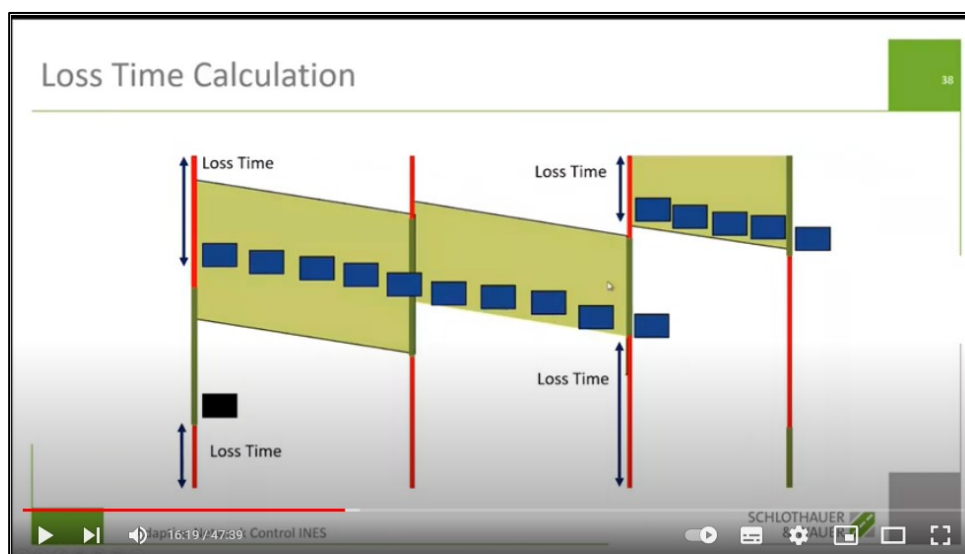


Figura 42 – Cálculo do atraso
Figura extraída de Schlothauer & Wauer (2022a) [69]

No vídeo, o apresentador explica que se a etapa de saturação for inconclusiva com relação ao plano a ser selecionado, então é feito o cálculo de atraso (*Loss Time*). Assim, o cálculo do atraso seria feito apenas se houver mais de um plano com tempo de verde que corresponde a uma saturação que caia dentro da faixa ótima (na parte do patamar do gráfico da Função de Utilidade). Dessa forma, o atraso teria apenas a função de “desempate” na escolha do melhor plano. Nesse caso de desempate, a escolha do plano recairia naquele plano cuja defasagem gerasse o menor atraso.

11.2.8 Detectores estratégicos

Os detectores estratégicos são posicionados em locais onde se espera que possa ocorrer congestionamentos (congestionamentos recorrentes).

O sistema usa a combinação de volume e velocidade para determinar congestionamento.

No gráfico da Figura 43, o eixo horizontal é velocidade e o eixo vertical é volume. O congestionamento seria detectado quando houvesse baixo volume e baixa velocidade.

O apresentador do vídeo não explica como a velocidade é medida, tampouco como o sistema reage para minimizar o congestionamento.

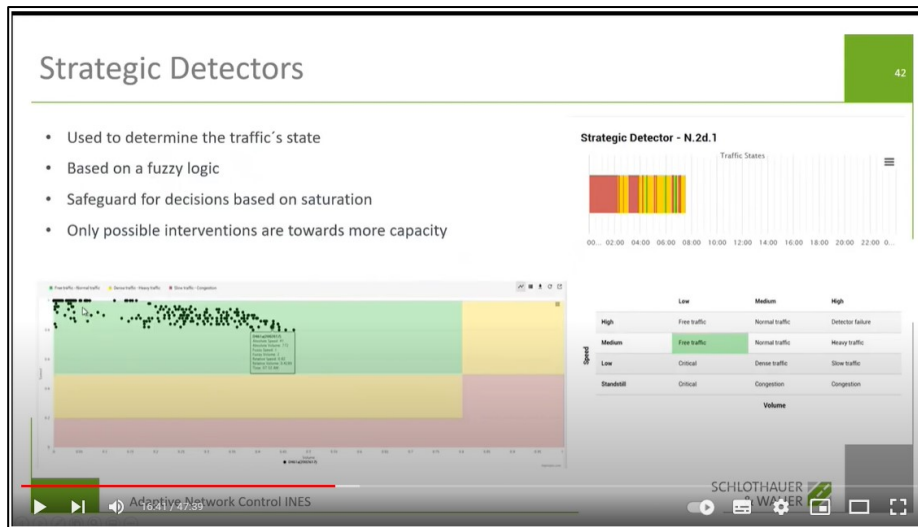


Figura 43 – Detectores estratégicos

Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Contudo, a Figura 43 traz alguns tópicos de forma genérica:

- ✓ Baseado em lógica fuzzy²⁷; e
- ✓ Intervenções para aumentar a capacidade.

²⁷ A lógica difusa ou lógica fuzzy é a forma de lógica multivalorada, na qual os valores verdade das variáveis podem ser qualquer número real entre 0 (correspondente ao valor falso) e 1 (correspondente ao valor verdadeiro), diferentemente do que se verifica na lógica booleana (também chamada, às vezes, lógica nítida), segundo a qual os valores lógicos podem ser apenas 0 ou 1.

A lógica difusa foi estendida para lidar com o conceito de verdade parcial, objetivando imitar o raciocínio humano em que o valor verdade situa-se entre o completamente verdadeiro e o completamente falso. Além disso, quando variáveis linguísticas são usadas, esses graus podem ser manipulados por funções específicas.

O termo lógica difusa foi introduzido em 1965 com a proposta da teoria de conjuntos difusos por Lotfi A. Zadeh. A lógica difusa tem sido aplicada em várias áreas, desde a teoria do controle à inteligência artificial. A lógica difusa tem sido, no entanto, estudada desde meados da década de 1920, como lógica infinito-valorada, por Lukasiewicz e Tarski.

Fonte:

[https://pt.wikipedia.org/wiki/L%C3%B3gica_difusa#:~:text=A%20l%C3%B3gica%20difusa%20ou%20l%C3%B3gica,%C3%A0s%20vezes%2C%20l%C3%B3gica%20n%C3%ADtida\)%2C](https://pt.wikipedia.org/wiki/L%C3%B3gica_difusa#:~:text=A%20l%C3%B3gica%20difusa%20ou%20l%C3%B3gica,%C3%A0s%20vezes%2C%20l%C3%B3gica%20n%C3%ADtida)%2C) (Acesso em 08/11/2022).

11.2.9 Redistribuição do tempo de verde e períodos de permissão (“*permission periods*”)

Considere a Figura 44a.

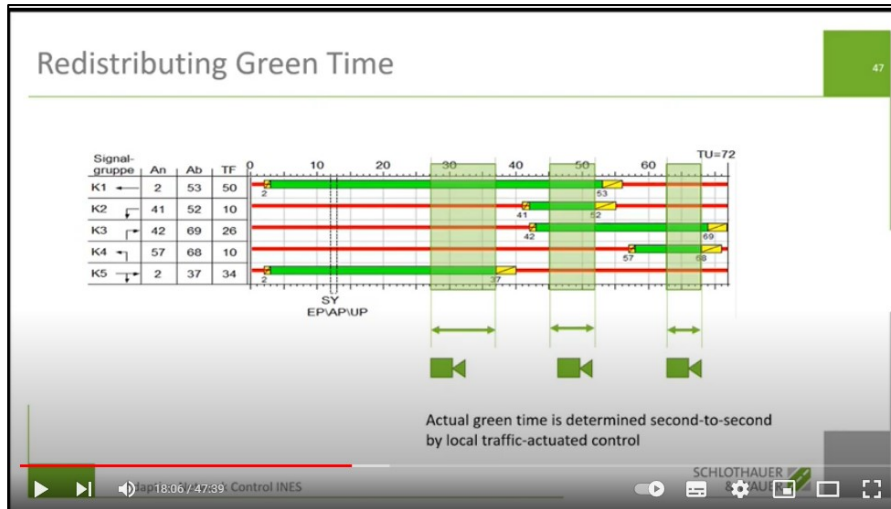


Figura 44a – Redistribuição do tempo de verde
 Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

Na Figura 44a, não foi possível entender a frase: ““Actual green” é determinado segundo a segundo pelo controle local atuado”.

Do vídeo original em alemão, a tradução literal seria: “o tempo verde real é determinado na unidade de controle local, levando em consideração a estrutura” (*Tatsächliche grünzeit wird im lokalen steuergerät unter berücksichtigung von rahmen bestimmt*). Aparentemente o verde real é o “*actual green*” do vídeo em inglês, sendo determinado na unidade de controle local (ou seja, no controlador), não havendo referência à expressão “segundo a segundo” e nem à atuação.

Na Figura 39a, tinha-se entendido que o “*Actual green*” seria o tempo de verde disponível, definido no plano semafórico (STP – *Signal Timing Plan*) e que o “*Needed green*” seria o tempo efetivamente necessário, determinado pela atuação em função de *gap out*.

Com base nesse entendimento, presume-se que o tempo de verde não utilizado (diferença entre “*Actual green*” e “*Needed green*”) é redistribuído entre os grupos semafóricos no próximo ciclo. O critério ou quais cálculos são feitos para essa redistribuição não estão claros.

Na Figura 44a, “An” significa instante de início de verde do grupo (1 segundo antes do início efetivo do verde); “Ab” é o instante final de verde do grupo; “Tf” é o tempo de verde do grupo (1 segundo a menos). Não se sabe o porquê de “An” indicar o instante do início do verde um segundo antes do início efetivo do verde.

Na Figura 44a estão indicados 3 períodos de permissão (*permission periods*).

Segundo o Manual (LISA Manual, s.d.) [72]:

Os “*permission periods*” são usados em coordenação.

Os períodos de permissão fornecem uma estrutura de tempo para controle de semáforos atuados pelo tráfego e garantem que a coordenação funcione bem.

Um período de permissão define um período de tempo dentro do ciclo em que uma determinada ação (por exemplo, estender um estágio, iniciar uma transição de estágio, iniciar um grupo semafórico) é permitida. Esses períodos de permissão são normalmente consistidos antes que qualquer ação seja iniciada para determinar se a respectiva ação é permitida.

Os planos de permissão são usados em controles coordenados atuados. Um plano de permissão geralmente consiste em vários períodos de permissão.

Existem três tipos de planos de permissão. Dependendo do tipo, podem ser definidas diferentes categorias de períodos de permissão e/ou diferentes subperíodos.

Ao usar vários planos semafóricos, cada plano geralmente é elaborado com seu próprio plano de permissão. Os períodos de permissão, no entanto, são definidos apenas uma vez e se aplicam a todos os planos de permissão. Apenas os inícios e finais dos períodos de permissão individuais diferem.

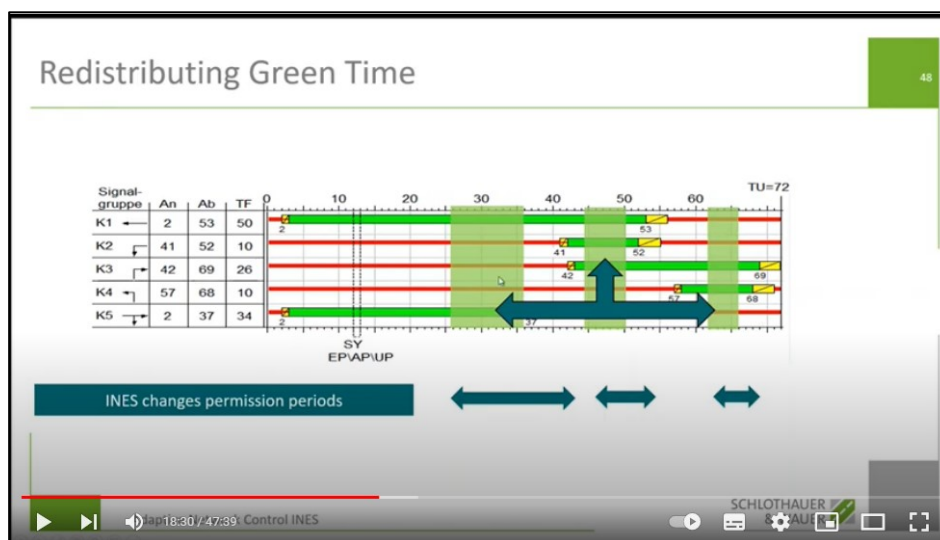


Figura 44b – Períodos de permissão

Fonte: Figura extraída de Schlothauer & Wauer (2022a) [69]

A Figura 44b mostra que INES pode mudar os períodos de permissão parametrizados no controlador. A mudança ocorre em intervalos de 5 minutos.

A mudança dos períodos de permissão e da redistribuição de verdes afetam a defasagem entre os cruzamentos.

11.2.10 Offset

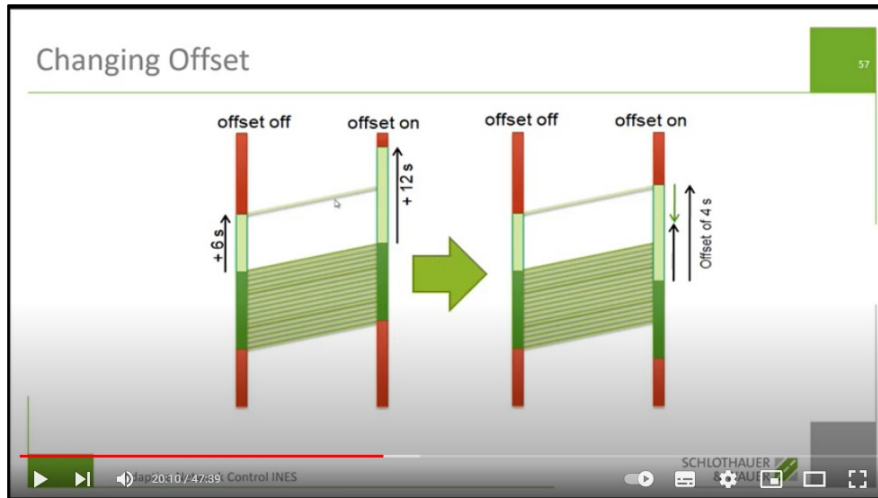


Figura 45 – Mudança de defasagem em função da mudança dos períodos de permissão e da redistribuição de verde
 Fonte: Figura extraída de Schlothauer & Wauer (2022) [69]

Com a mudança dos períodos de permissão, a região de atuação (verde mínimo / verde máximo) é alterada, mudando os inícios e finais de verde. A Figura 45 mostra como a defasagem entre os cruzamentos pode ser afetada pela mudança dos períodos de permissão e pela redistribuição de verdes.

Entretanto, como a palavra “offset” é designada no OCIT como atraso de grupo, não foi possível determinar com precisão se a palavra “offset” da Figura 45 se refere a atraso de grupo ou à defasagem. No vídeo, o apresentador fala em coordenação.

11.3 Outras informações obtidas do Schlothauer & Wauer (2022b) [70]

Este segundo vídeo da empresa Schlothauer & Wauer, disponibilizado em outubro de 2022, versa sobre os fatores que influenciam o processo de escolha do sistema em um projeto. Ao longo do vídeo, são feitas várias comparações com o SCOOT.

11.3.1 Posicionamento dos detectores

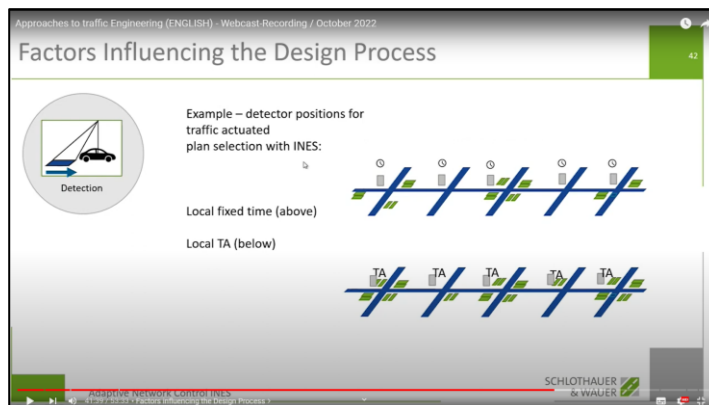


Figura 46a – Posição dos detectores
 Fonte: Figura extraída de Schlothauer & Wauer (2022b) [70]

A Figura 46a mostra que, para o INES, num controle local a tempo fixo, o detector é usado para ter uma medida da demanda (parte superior da Figura 46a).

Já para um controle local atuado com seleção de plano com INES (TA = *Traffic Actuated*), o posicionamento dos detectores seria como o indicado na parte inferior da Figura 46a.

No slide seguinte (Figura 46b), é informado que a posição preferível para posicionar o detector no controle atuado local é a 40 metros da linha de retenção (parte superior da Figura 46b).

O detector de congestionamento do INES (detector estratégico) é posicionado fora da fila normal e “dentro” da visualização do congestionamento. O apresentador afirma ainda que a posição do detector estratégico do INES é a mesma que se utiliza para os detectores SCOOT (parte inferior da Figura 46b).

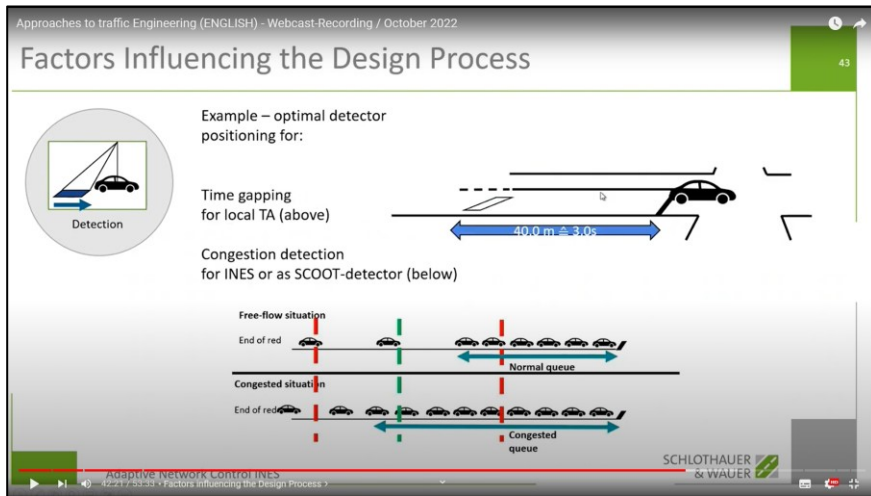


Figura 46b – Posição dos detectores
 Fonte: Figura extraída de Schlothauer & Wauer (2022b) [70]

11.3.2 Comparação de arquitetura INES e SCOOT

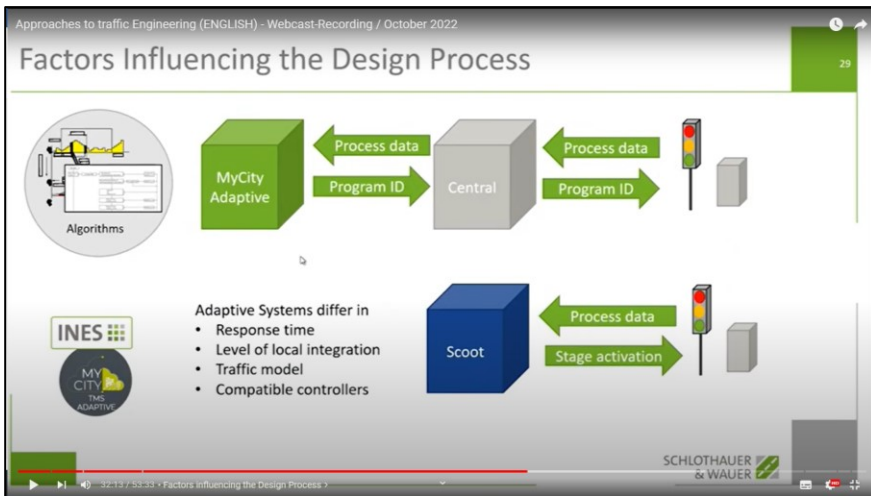


Figura 47 – Arquitetura SCOOT e INES
 Figura extraída de Schlothauer & Wauer (2022b) [70]

No SCOOT, o controlador tem apenas a função de executar os comandos de ativação dos estágios.

No INES, o controlador pode executar qualquer tipo de controle, tempo fixo, atuado etc. O controlador faz a maior parte dos cálculos segundo a segundo. No vídeo, o apresentador afirma na sequência que o LISA faz 80% das tarefas.

11.3.3 Otimização da defasagem

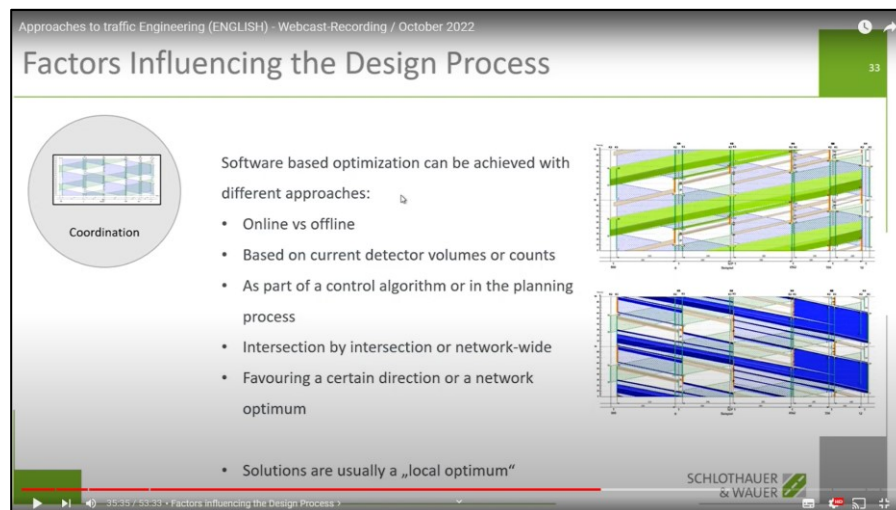


Figura 48 – Otimização da defasagem

Fonte: Figura extraída de Schlothauer & Wauer (2022b) [70]

Na Figura 48 são pontuadas algumas abordagens possíveis para a otimização da defasagem.

A otimização pode ser feita de forma *off-line* ou *on-line*.

Na abordagem *on-line* é levado em conta o volume detectado em tempo real e não baseado em contagens.

A vantagem da abordagem *off-line* é que a otimização é feita para a rede e não aos pares (interseção por interseção) como em sistemas *on-line*.

Observação: Neste vídeo é usado o termo “*offset*” para designar defasagem.

Cabe aqui fazer as seguintes considerações.

Nem todo sistema *on-line* faz a otimização da defasagem aos pares. O SCOOT, por exemplo, faz a otimização para uma miniárea, onde cada interseção está associada a uma miniárea, conforme descrito no item 5.5, em especial a Figura 20.

A abordagem *off-line* tem a capacidade de fazer a otimização de forma mais ampla. Rodando de forma *off-line*, tem-se mais tempo disponível para o processamento de algoritmos eficazes até achar o ponto ótimo. Rodar algoritmos *on-line* para achar o ponto ótimo é mais difícil por não haver tempo disponível para o processamento, uma vez que a solução tem que ser implementada em tempo real. Ver, por exemplo, o algoritmo de “*hill-climbing*”, adotado tanto no TRANSYT (*off-line*) como no SCOOT (*on-line*), ver o item 2.2.1.1

e o item 5. Enquanto no primeiro é possível rodar todos os passos do algoritmo, no segundo só é rodado um passo.

Entretanto, aplicar o resultado da otimização feita *off-line* em situações *on-line* não traz, necessariamente, resultados satisfatórios pois as condições sob as quais foi feita a otimização *off-line* podem não permanecer exatamente iguais às encontradas no instante da sua implementação.

Cada método tem as suas próprias vulnerabilidades. Nada garante que o ótimo obtido de forma *off-line* será o ótimo para a situação encontrada no instante da sua implementação, podendo, inclusive, estar bem distante da solução ótima. Por outro lado, a solução encontrada na otimização *on-line* provavelmente não será o ponto ótimo, podendo ser apenas uma solução aproximada.

11.4 Conclusões

Não foi possível localizar na Internet documentação sobre INES de fontes independentes ou acadêmicas. A análise se restringiu ao material disponibilizado pela própria empresa que desenvolveu o sistema.

Da análise do material disponível, pode-se pontuar o seguinte:

- Trata-se de um sistema de seleção dinâmica de planos, associado a um esquema de atuação.
- A seleção do melhor plano (ou o melhor cenário) se dá por meio do *split*, com base no qual se calcula a saturação (que é grau de saturação), como sendo a razão entre “*Needed green*” e “*Actual green*”, onde presume-se que “*Needed green*” é o tempo de verde efetivo, determinado pela atuação em função de *gap out* e “*Actual green*” é o tempo de verde disponível, dado pelo plano semafórico.
- Uma vez determinada a saturação propiciada por cada plano, verifica-se aquele(s) que recae(m) no trecho constante de uma Função de Utilidade (valor máximo da função).
- Se houver mais de um plano cuja saturação corresponda ao valor máximo da Função de Utilidade, é calculado o atraso correspondente a cada um dos planos e é selecionado aquele cuja defasagem propicie o menor atraso.
- Não foi possível determinar com exatidão o papel do LISA e do VISSIM no processo adaptativo.

Na análise do material disponível, foi empregado diversas vezes o verbo “presumir”, denotando haver muito mais dúvidas do que certezas.

Entre as dúvidas, além das já expostas ao longo desta análise, destaca-se a seguinte: do material disponível permite-se concluir que o sistema não otimiza o tempo de ciclo, adotando-se aquele do plano selecionado. Ora, não se sabe se o sistema possui algum mecanismo de transição de planos, para evitar que os tempos de ciclo (do plano anterior e o do novo plano) sejam muito diferentes, provocando distúrbios no tráfego.

Também não fica claro se o ganho obtido pela redistribuição dos tempos de verde supera o impacto negativo provocado pela perda da defasagem calculada para o plano.

Outra dúvida é inerente a sistemas de seleção dinâmica de planos: até que ponto a performance e o desempenho do sistema depende da qualidade e quantidade dos planos disponíveis para a seleção.

Com relação ao enfrentamento do congestionamento, entendeu-se que o sistema possui algum mecanismo em relação a congestionamentos recorrentes (nos locais onde se espera que possa haver congestionamento e onde foram implantados detectores estratégicos), embora não se tenha entendido exatamente como, uma vez que “intervenções para aumentar a capacidade” soa muito vago. Além disso, o sistema só irá reagir quando a fila atingir o detector estratégico, já com risco de propagação do congestionamento a montante.

Não há nenhuma menção com relação a congestionamentos não recorrentes, situações com obstruções a jusante e ações de *gating*. Com relação à obstrução a jusante (*exit blocking*) seria interessante saber o que acontece com a atuação, cálculo de saturação, seleção de plano (ou de cenário), redistribuição de verde etc.

Com relação à análise do sistema, o material disponível não foi suficiente para explicar o funcionamento do sistema. Infelizmente, não foi possível obter documentação técnica como Manuais, *Handbooks* ou *User Guides*.

Com relação à maturidade, o sistema foi implantado apenas em poucas cidades: Hamburg – população de 1,8 milhão; Augsburg – população de 295 mil e Heidelberg – população de 160 mil (dados de 2019). Mesmo assim, não se sabe em quantas interseções em cada cidade. A maior cidade é Bogotá, população de 7,2 milhões (dado de 2018), com 808 controladores conectados ao sistema.

Finalmente, deve-se destacar que o simples uso de controladores com protocolo OCIT não garante a compatibilidade com o sistema INES. É necessário que a lógica do controlador seja compatível com as funcionalidades do sistema. Por exemplo, o controlador deve ter funcionalidades como períodos de permissão e redistribuição de verdes.

12. TrafGo

Segundo o Relatório IDOM (2019 – D2) [20]²⁸:

TrafGo é da solução desenvolvida pela empresa SERTTEL e trata-se de uma solução centralizada que faz modificações dentro do tempo de ciclo. Os requisitos para sua implementação são:

- A sensorização de todas as faixas das vias de acesso, não considerando informações relacionadas com os movimentos dos pedestres. Tem a capacidade de integrar informações de sensorização através de serviços de informação de tráfego que ficam na internet, como o Google, com uma periodicidade de um (1) minuto. As mudanças são implementadas no ciclo seguinte;
- Uma latência de rede máxima de 1 segundo é necessária para sua implementação com informação dos dados de Google;
- Pode ser implementado usando o protocolo UTMC2, AENOR e NTCIP;
- Segundo a informação do fornecedor, os algoritmos de otimização do modo adaptativo são similares aos utilizados pela Kapsch (ITACA) no ciclo corrente. Neste caso o fornecedor informou que as latências necessárias devem ser inferiores a 200 ms.

TrafGo tem uma implementação alternativa que não usa sensores na via (nenhum sensor). Nesta implementação as informações do trafego são coletadas a cada minuto através de um aplicativo com uma comunicação dedicada com o Google (serviço pago com os servidores de informação de tráfego do Google). Como pode se deduzir a implementação deste sistema é bem mais econômica que a implementação do sistema usando sensores na via.

De acordo com o site da Serttel: <http://www.serttel.com.br/solucoes/transito-inteligente/> (Acesso em 20/07/2022):

TrafGo é uma ferramenta desenvolvida para fazer o controle e a otimização dos tempos dos semáforos utilizando as informações do Google Maps. O sistema disponibiliza uma interface gráfica para avaliar o desempenho do sistema em execução, bem como para monitorar remotamente, por meio de indicadores de tempo de deslocamento dos veículos, o tráfego na região controlada. As informações de trânsito na via são fornecidas pela Google Maps API, baseadas nos mesmos algoritmos que alimentam o Google Maps. O sistema TrafGo, por sua vez, analisa os indicadores e calcula os tempos dos semáforos e envia para os controladores semafóricos. Por fim, os semáforos executam os comandos da central, tornando o trânsito mais eficiente e, por consequência, reduzindo o tempo de deslocamento do motorista.

Benefícios:

- Baixo custo de implantação e manutenção, já que dispensa a instalação de sensores na via;
- Monitoramento e controle em tempo real;
- Otimiza ciclos, tempos de verde e defasagens dos semáforos; e
- Trânsito mais eficiente.

²⁸ Observe-se que as informações foram coletadas diretamente da empresa.

Não foi possível localizar fontes independentes sobre o sistema, bem como material com informações técnicas sobre os princípios de otimização.

13. BIBLIOGRAFIA

- [1] Os Custos do Congestionamento na Cidade de S. Paulo.
Disponível em:
<https://www.mobilize.org.br/midias/pesquisas/custos-do-congestionamento-em-sp.pdf>
(Acesso em 04/07/2022).
- [2] Cintra, 2014.
Os Custos dos Congestionamentos na Cidade de São Paulo.
Cintra, Marcos.
Texto para Discussão 356. Escola de Economia de São Paulo da Fundação Getúlio Vargas FGV-EESP. Abril de 2014.
- [3] Papageorgiou et al, 2003.
Papageorgiou, Markos; Diakaki, Christina; Dinopoulou, Vaya; Kotsialos, Apostolos; Wang, Yibing.
Review of Road Traffic Control Strategies.
Proceedings of the IEEE, Vol. 91, No. 12, December 2003.
- [4] Souza et al, 2017.
Souza, Marcello Victorino Junqueira de; Pina Filho, Armando Carlos de.
Automação Semafórica: Aplicando Tecnologia para Maximizar a Capacidade Viária Existente.
Revista dos Transportes Públicos – ANTP – Ano 40 – 3º quadrimestre – 2017.
- [5] IDOM, 2019 – D3.
IDOM
Consultancy Services for Alternatives Analysis and Other Studies for Traffic Lights Systems in São Paulo. D3 – Preliminary Economic Analysis.
Versão 1 – Revisão 1. World Bank. Edição 1. Junho de 2019.
- [6] Robertson et al, 1991.
Robertson, Dennis I.; Bretherton, R. David.
Optimizing Networks of Traffic Signals in Real Time – The SCOOT Method.
IEEE Transactions on Vehicular Technology, Vol. 40, No. 1, February 1991.
- [7] Bell et al, 1986.
Bell, M. C.; Bretherton, R. D.
Ageing of fixed time signal plans.
Presented at Inst. Elec. Eng. 2nd Int. Conf. Road Traffic Control, 1986.
- [8] Robertson et al, 1982.
Robertson, D. I.; Hunt, P. B.
A Method of Estimating the Benefits of Co-Ordinating Signals by TRANSYT and SCOOT.
Traffic Engineering and Control, 1982.
- [9] Hunt et al, 1981.
SCOOT: A Traffic-Responsive Method of Co-Ordinating Signals.
TRRL Laboratory Report LR 1014, Transport and Road Research Laboratory, Crowthorne, 1981.
- [10] Hunt et al, 1982.
Hunt, P. B.; Robertson, D. I.; Bretherton, R. D.; Royle, M. C.
The SCOOT On-Line Traffic Signal Optimisation Technique.
Traffic Engineering & Control Volume: 23 Issue Number: 4. 1982.

- [11] Adacher et al, 2020.
Adacher, Ludovica; Tiriolo, Marco.
Performance Analysis of Decentralized VS Centralized Control for the Traffic Signal Synchronization Problem.
Hindawi Journal of Advanced Transportation Volume 2020, Article ID 8873962, 19 pages. 9 November 2020.
- [12] Chow et al, 2019.
Chow, Andy H. F.; Sha, Rui; Li, Shuai.
Centralised and Decentralised Signal Timing Optimization Approaches for Network Traffic Control.
23rd International Symposium on Transportation and Traffic Theory, ISTTT 23, 24-26 July 2019, Lausanne, Switzerland. Transportation Research Procedia 38. Pg 222–241. 2019.
- [13] MOVA
MOVA – *Microprocessor Optimised Vehicle Actuation*.
TRL Software.
Disponível em:
<https://trlsoftware.com/products/traffic-control/mova/>
(Acesso em 15/06/2022).
- [14] SWARCO, 2020.
SWARCO – Smart Green – Intelligent solutions for Smart Cities.
Disponível em:
https://www.swarco.com/sites/default/files/public/downloads/2020-08/SMART%20GREEN%20BROCHURE%202020_EN%20VERSION%20WEBEDITION_2.pdf
(Acesso em 06/07/2022).
- [15] Fehon
Fehon, Kevin.
Adaptive Traffic Signals: Overview.
DKS Associates.
Apresentação disponível em:
https://mtc.ca.gov/sites/default/files/4-Adaptive_Signal_Control_-_How_Does_It_Work.pdf
(Acesso em 17/08/2022).
- [16] Work et al, 2009.
Work, Daniel B.; Tossavainen, Olli-Pekka; Jacobson, Quinn; Bayen, Alexandre M.
Lagrangian Sensing: Traffic Estimation with Mobile Devices.
2009 American Control Conference Hyatt Regency Riverfront, St. Louis, MO, USA, June 10-12, 2009.
- [17] Liu, 2019.
Liu, Henry X.
Data Driven Smart Transportation – 99/Didi.
Presentation at University of Sao Paulo, Brazil. May 31, 2019.
- [18] DASCOS, 2019.
DASCOS: Dynamic Area-Wide Signal Control Optimization System.
Didi Smart Transportation, White Paper. April 2019.
- [19] Wei et al, 2020.
Wei, Hua; Zheng, Guanjie; Gayah, Vikash; Li, Zhenhui.
A Survey on Traffic Signal Control Methods.
arXiv: 1904.08117. v3 16. Jan 2020.

- [20] IDOM, 2019 – D2.
IDOM
Consultancy Services for alternatives Analysis and Other Studies for Traffic Lights Systems in São Paulo. D2 – Analysis of Technological Alternatives.
Versão 1 – Revisão 1. The World Bank. Junho de 2019.
- [21] Matheus, 2018.
Matheus, Yuri.
O Modelo OSI e suas Camadas. 24/01/2018.
Disponível em:
<https://www.alura.com.br/artigos/conhecendo-o-modelo-osi>
(Acesso em 01/07/2022).
- [22] Kraus et al, 2011.
Kraus Junior, Werner; Melgarejo, Luiz Fernando Bier; Pieri, Giovanni; Polônia, Pablo; Tobaldini, Ricardo Ghisi; Montez, Carlos Barros.
Proposta de Implantação do Padrão NTCIP no Contexto Brasileiro.
XXV ANPET – Congresso de Pesquisa e Ensino em Transportes – Belo Horizonte – MG – 7 a 11 de Novembro de 2011.
- [23] OCIT Outstations – Introduction, 2018.
OCIT Outstations – Introduction to the System.
- [24] OCIT-O Version 3.0 – Mirror, 2018.
OCIT Outstations – Function Mirror.
OCIT-O Version 3.0 for Traffic Signal Controllers.
- [25] OCIT Outstations Version 3.0 – Glossary, 2018.
OCIT-O Version 3.0 for Traffic Signal Controllers – Glossary.
- [26] OCIT Outstation, 2012.
OCIT Outstations – Traffic Signal Controllers.
- [27] OCIT Goes Cooperative, 2018.
OCIT-O V3.0 / OCIT-O Car – Consortium for Standardization of Interfaces of Road Traffic Control – Prospect.
Disponível em: https://www.ocit.org/media/ocit-o_v3.0_prospekt_en.pdf
(Acesso em 05/07/2022).
- [28] Aavani et al, 2017.
Aavani, P.; Mithun, K. Sawant; Sneha, Sawant; Rohit, S. Deshmukh.
A Review on Adaptive Traffic Controls Systems
International Journal of Latest Engineering and Management Research (IJLEMR), pp. 52-57. January 2017.
- [29] TRL Software.
SCOOT
Disponível em:
<https://trlsoftware.com/products/traffic-control/scoot/>
(Acesso em 05/07/2022).
- [30] Yunex Traffic.
Manage Traffic Control in Real Time.

- Disponível em:
<https://www.yunextraffice.com/us/en/portfolio/adaptive-technologies/scoot>
(Acesso em 05/01/2022).
- [31] Bell, 2006.
Bell, M. G. H.
Transport Planning and Traffic Engineering Chapter 27 – Signal Control in Networks.
Published by Elsevier Ltd. (First Edition: 1997; Reprinted: 2001, 2002, 2004, 2005, 2006). Edited by C. A. O’Flaherty.
- [32] Hill Climbing, Wikipedia
Hill Climbing.
Disponível em:
https://en.wikipedia.org/wiki/Hill_climbing
(Acesso em 25/06/2022).
- [33] Shelby et al, 2008.
Shelby, V; Bullock, Darcy M.; Gettman, Doug; Ghaman, Raj S.; Sabra, Ziad A.; Soyke, Nils.
An Overview and Performance Evaluation of ACS Lite – A Low Cost Adaptive Signal Control System.
Submitted to the 87th TRB Annual Meeting in Washington, DC, January 2008.
- [34] Wang et al, 2019.
Wang, Hong; Zhu, Meixin; Hong, Wanshi; Wang, Chieh (Ross); Tao, Gang; Wang, Yin Hai.
Optimizing Signal Timing Control for Large Urban Traffic Networks Using an Adaptive Linear Quadratic Regulator Control Strategy.
IEEE Transactions on Intelligent Transportation Systems (2019).
- [35] SCOOT
User Guide – 666/HF/16940/000 – Issue 35.
- [36] TRL – Version History.
Disponível em:
<https://trlsoftware.com/products/traffic-control/scoot/scoot-version-history/>
(Acesso em 15/07/2022).
- [37] Ming, 1997.
Ming, Sun Hsien.
Uma breve descrição do Sistema SCOOT – Nota Técnica 201/97.
Companhia de Engenharia de Tráfego – CET SP.
Disponível em:
<http://www.cetsp.com.br/media/20764/nt%20201%20revisado.pdf>
(Acesso em 11/07/2022)
- [38] SCOOT 0474.
SCOOT Traffic Handbook, SCOOT 0474: Operational Guide Customising. Issue C 29-Feb-2000.
- [39] SCOOT 0414
SCOOT Traffic Handbook, SCOOT 0414 Executive Summary: Description of SCOOT. Issue B 29-Feb-2000.
- [40] Transport for NSW.
SCATS is a traffic control system and keep traffic moving.
Disponível em:
<https://www.scats.nsw.gov.au/#:~:text=SCATS%20is%20a%20traffic%20control,patterns%20and%20keep%20traffic%20moving>
(Acesso em 24/08/2022).

- [41] Smardi, 2001.
Smardi, Mohammad Ghaleb.
A Knowledge-Based Traffic Signal Control Application.
Paper Submitted to the Graduate Faculty of the North Dakota State University of Agriculture and Applied Science.
December 2001.
- [42] SCATS 6
The New Generation: SCATS 6 – Functional Description.
Aldridge Traffic Controllers.
[file:///C:/Users/sunm/Downloads/Introduction_To_New_Generation_Scats_6_5%20\(1\).pdf](file:///C:/Users/sunm/Downloads/Introduction_To_New_Generation_Scats_6_5%20(1).pdf)
(Acesso em 26/08/2022).
- [43] Wikipedia
Sydney Coordinated Adaptive Traffic System, 22 May 2022.
Disponível em: https://en.wikipedia.org/wiki/Sydney_Coordinated_Adaptive_Traffic_System
(Acesso em 24/08/2022).
- [44] Lee et al, 2002.
Lee, Sang Soo; Lee, Seung Hwan; Oh, Young Tae; Choi, Kee Choo.
Development of Degree of Saturation Estimation Models for Adaptive Signal Systems.
Transportation Engineering Vol. 6, No. 3, pp 337-345. September 2002.
- [45] Day et al, 2010.
Day, Christopher M.; Bullock, Darcy M.
Performance Based Management of Arterial Traffic Signal Systems.
NCHRP Project 3-79a. Arterial Performance Measures, Final Report – Volume 1. 2010.
- [46] Fellendorf et al, 1997.
Fellendorf, Martin; Aongusa, Conall Mac; Pierre, Manual.
LRT Priority within the SCATS Environment in Dublin – A Traffic Flow Simulation Study.
Urban Transport and the Environment for the 21st Century.
Transactions on the Built Environment vol 30, 1997.
- [47] Stevanovic et al, 2009.
Stevanovic, Aleksandar; Kergaye, Cameron; Martin, Peter T.
SCOOT and SCATS: A Closer Look into Their Operations.
Transportation Research Board 88th Annual Meeting, Washington January 2009.
- [48] McCann.
McCann, Barry.
A Review of SCATS Operation and Deployment in Dublin.
Intelligent Transportation Systems, Dublin City Council, Wood Quay, Dublin. 2014.
- [49] Lin et al, 2010.
Lin, Liang-Tay; Huang, Hung-Jen.
An Effective Interval of Traffic Signal Coordination under Safety and Efficiency Considerations.
Journal of the Chinese Institute of Engineers, Vol. 33, No. 2, pp. 271-285. 2010.
- [50] Yagoda et al, 1973.
Yagoda, H. N.; Principe, E. H.; Vick, C. E.; Leonard, B.
Subdivision of Signal Systems into Control Areas.

- Traffic Engineering, Vol. 43, No. 12, pp. 42–45, 1973.
- [51] Chang, 1985.
Chang, Edmond Chin-Ping.
How to Decide the Interconnection of Isolated Traffic Signals.
Proceedings of the 1985 Winter Simulation Conference, Gantz D., Blais, G., and Solomon, S., eds., pp. 445-453.
1985.
- [52] Szazs, 1991.
Szazs, Pedro A.
Necessidade de Sincronismo para Evitar Perda de Capacidade nas Interseções.
Nota Técnica 138. CET. Dezembro de 1991.
- [53] Hamilton et al, 2013.
Hamilton, Andrew; Waterson, Ben; Cherrett, Tom; Robinson, Andrew; Snell, Ian.
The Evolution of Urban Traffic Control: Changing Policy and Technology.
Transportation Planning and Technology, Vol. 36, No. 1, 2443. 2013.
- [54] Malcontenti-Wilson et al, 2013.
Malcontenti-Wilson, Adam; Vu, Hai L.; Cadusch, Peter; Branch, Philip; Lint, Hans van; Hoogendoorn, Serge.
Exploring Urban Traffic Flow Dynamics.
October, 2013.
- [55] López et al, 1996.
López, J.; Peck, C.
ITACA Adaptive Traffic Control.
Computer Science pp. 184-5, 187-8
- [56] Srivastava, 2022
Srivastava, Anil.
ITS Deployment & Enforcement Challenges and Opportunities (28/04/2022)
Disponível em : <https://unece.org/transport/documents/2022/04/presentations/anil-srivastava-its-deployment-enforcement-challenges-and>
- [57] Sahri, 2005.
Sahri, Nurul Huda Mohd.
The Effectiveness of the Implementation of ITACA System to Traffic Light Junction in Putrajaya.
Dissertation submitted in partial fulfillment of the requirements for the Bachelor of Engineering (Hons) (Civil Engineering). Universiti Teknologi PETRONAS. June 2005.
- [58] Dubey, 2019.
Dubey, Kajal.
Adaptive Traffic Control System: The Smart and Imperative Traffic Monitoring System for India.
International Journal for Research in Applied Science & Engineering Technology (IJRASET) Volume 7 Issue XII, Dec 2019.
- [59] Revitalização CTA-3, 2006.
Manual do Software ITACA Optimus 5.0
Serviços de Revitalização da CTA-3 – Telvent.
Prefeitura do Município de São Paulo.

- 31-10-2006.
- [60] Manolis et al, 2018.
Manolis, Diamantis; Diakaki, Christina; Papamichail, Ioannis; Papageorgiou, Markos.
Simulation Investigations of the Coordinated Traffic-Responsive Signal Control Strategy TUC with Actuation at the Local Junction Level.
European Transport Research Review. 2018.
- [61] Papageorgiou.
Papageorgiou, Markus.
Network-Wide Traffic Signal Control Strategy TUC: Methodology and Applications.
Apresentação enviada por e-mail em 14/09/2022.
- [62] Gazis et al, 1963.
Gazis, D. C.; Potts, R. B.
The Oversaturated Intersection.
Proceedings of the 2nd International Symposium on Traffic Theory, London, UK, pp. 221–237, 1963.
- [63] Aboudolas et al, 2009.
Aboudolas, K.; Papageorgiou, M.; Kosmatopoulos, E.
Store-and-Forward Based Methods for the Signal Control Problem in Large-Scale Congested Urban Road Networks.
Transportation Research Part C 17, pp. 163–174. 2009.
- [64] Smaragdis et al, 2003.
Smaragdis, Emmanouil; Dinopoulou, Vaya; Aboudolas, Kostas; Diakaki, Christina; Papageorgiou, Markos.
Application of the Extended Traffic Signal Control Strategy TUC to the Southampton Urban Road Network.
IFAC Control in Transportation Systems, Tokyo, Japan, 2003.
- [65] Kouvelas et al, 2011.
Kouvelas, Anastasios; Aboudolas, Konstantinos; Kosmatopoulos, Elias B.; Papageorgiou, Markos.
Adaptive Performance Optimization for Large-Scale Traffic Control Systems.
Dezembro 2011.
- [66] SMART INTERSECTION, 2017.
Kaminski, Piotr; Kaminski, Tomasz.
SWARCO Technology Smart Intersection Manual.
Revision: 1.0. August 18, 2017.
- [67] SMART CORRIDOR, 2019.
Kaminski, Piotr; Kaminski, Tomasz.
SWARCO Technology Smart Corridor Programmers Manual.
Revision: 1.0. October 29, 2019.
- [68] Schlothauer & Wauer
INES – The Intelligent Adaptive Traffic Control System.
<https://www.schlothauer.de/en/software-systems/ines/>
(Acesso em 11/10/2022).
- [69] Schlothauer & Wauer, 2022a.
Schlothauer & Wauer.
Presentation INES – Intelligent Network Control (ENGLISH) – Webcast Recording/June 2022 – video.
Disponível em: <https://www.youtube.com/watch?v=nW7UIYGZX7M>
(Acesso em 21/10/22).

- [70] Schlothauer & Wauer, 2022b.
Schlothauer & Wauer.
Approaches to Traffic Engineering (ENGLISH) – Webcast-Recording/October 2022 – video.
Disponível em: <https://www.youtube.com/watch?v=YlXevG7C00>
(Acesso em 21/11/2022).
- [71] Bogotá, SWARCO.
Striving for a Better Bogotá.
Disponível em: <https://www.swarco.com/stories/striving-better-bogota>
(Acesso em 28/10/2022).
- [72] LISA Manual
LISA+ Manual Version 5.2
- [73] Daganzo, 1994.
Daganzo, Carlos.
The Cell Transmission Model: Network Traffic.
Working Papers, California Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies, UC Berkeley. 1994.
- [74] Tian et al, 2008.
Tian, Yi; Zhao, Dongbin; Yi, Jianqiang.
A Fuzzy Logic Controller with Adaptive Dynamic Programming Optimization for Traffic Signals.
Fifth International Conference on Fuzzy Systems and Knowledge Discovery 2008.
- [75] Le et al, 2017.
Le, Tung; Vu, Hai L.; Walton, Neil; Hoogendoorn, Serge P.; Kovács, Péter; Queija, Rudesindo N.
Utility Optimization Framework for a Distributed Traffic Control of Urban Road Networks.
Transportation Research Part B 105 (2017) 539–558
- [76] Elouni et al, 2021.
Elouni, Maha; Abdelghaffar, Hossam M.; Rakha, Hesham A.
Adaptive Traffic Signal Control: Game-Theoretic Decentralized vs. Centralized Perimeter Control.
Sensors 21, 274. 2021.
- [77] Guo et al, 2018.
Guo, Xinlan; Li, Tao.
Traffic Signal Optimization Control Based on Fuzzy Logic Controller.
2nd International Conference on Applied Mathematics, Modelling and Statistics Application – AMMSA. 2018.
- [78] Zhao et al, 2012.
Zhao, Dongbin; Dai, Yujie; Zhang, Zhen.
Computational Intelligence in Urban Traffic Signal Control: A Survey.
IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews, Vol. 42, No. 4, July 2012.