

## **SIRI - Um simulador mesoscópico para redes de semáforos**

*Luis Molist Vilanova*

### **RESUMO**

Siri é um simulador mesoscópico, na forma de programa de computador, cuja função é calcular programações de semáforos coordenados operando na estratégia de tempos fixos. O programa otimiza o desempenho dos semáforos no que tange ao atraso e ao número de paradas. Foi desenvolvido pela CET de São Paulo e seu lançamento ocorreu no início de 2004. Atualmente, cerca de 250 semáforos paulistanos operam segundo os tempos calculados pelo Siri, que se encontra na versão 2.2. O texto delinea os motivos que levaram à sua criação, o modelo de tráfego e o algoritmo utilizados, bem como a medida de sua confiabilidade até o presente momento.

### **HISTÓRICO**

A programação das redes de semáforos coordenados sempre foi uma questão não totalmente resolvida no Brasil. Além das dificuldades inerentes a este tipo de trabalho, como, por exemplo, o levantamento, tratamento e aplicação dos dados, constata-se a inadequação dos simuladores estrangeiros a algumas das nossas peculiaridades. Pode-se citar como a deficiência mais sentida o fato de que tais simuladores desconsideram o tamanho físico das caixas entre dois semáforos. É como se pudessem caber infinitos veículos entre eles. Quando é necessário programar uma rede numa situação congestionada, tão comum em muitas de nossas cidades, a questão passa a ser fundamental. Existem simuladores microscópicos que consideram o tamanho finito das caixas, mas que não se aplicam aos projetos de programação de redes coordenadas, principalmente devido à explosão do tempo de processamento.

Outros pontos onde se sente ausência de um tratamento mais próximo à nossa realidade são os casos dos ciclos duplos assimétricos, das botoeiras de pedestres chamando estágios específicos em cruzamentos e dos tempos de verde de segurança configurados por grupo semafórico e não por estágio.

São extremamente raros os esforços para criação de programas de software brasileiros na área de Engenharia de Trânsito e, em especial, no campo semafórico. Nosso país, internacionalmente reconhecido como desenvolvedor de aplicativos de software e possuidor de importante acervo técnico na área de Engenharia de Trânsito está deixando de aproveitar ótimas oportunidades para avançar neste mercado. Pode-se citar como notável exceção o trabalho desenvolvido no Centro Tecnológico da Universidade Federal de Santa Catarina por um grupo de pesquisa, coordenado pelo professor Werner Kraus Júnior, que está trabalhando num sistema nacional de TI para controle de tráfego.

A fim de poder contar com uma ferramenta mais adequada à nossa realidade, a CET de São Paulo desenvolveu o programa de computador Siri - Simulador de Redes de Semáforos, cuja função é calcular o tempo de ciclo, tempos de verde e defasagens de uma rede de semáforos

com o objetivo de causar os menores tempos de espera e o menor número de paradas possível dos veículos, respeitadas as condições da segurança viária.

O desenvolvimento do programa demorou quatro anos e sua primeira versão ficou disponível no início de 2004. Atualmente, encontra-se na versão 2.2, que incorpora vários recursos apontados pelos técnicos que acompanharam em campo a implementação das programações da primeira versão. A experiência prática foi importante tanto para melhorar a interface com o usuário como para promover alterações significativas no próprio algoritmo. Atualmente, existem cerca de 250 semáforos, em São Paulo, que operam segundo as programações semafóricas calculadas pelo Siri.

## **O ALGORITMO E O MODELO DE TRÁFEGO**

Siri é um programa de computador de simulação de redes de semáforos de tempos fixos. Seu objetivo é calcular os melhores tempos de verde, defasagens e tempo de ciclo, adotando-se o termo "melhores" no sentido de que os veículos venham a ter o menor tempo de espera e o menor número de paradas possível. O algoritmo deve respeitar determinadas condições de contorno impostas pelo usuário, notadamente o tempo de ciclo máximo, os tempos de verde de segurança dos grupos semafóricos e os tempos de entreverdes.

O programa pode ser classificado como simulador mesoscópico, segundo a definição do Prof. Licínio da Silva Portugal em seu livro "Simulação de Tráfego: Conceitos e Técnicas de Modelagem". Segundo ele: "Os modelos (mesoscópicos) formam uma classe intermediária quanto ao realismo e detalhamento. Os veículos são agrupados em pelotões e tratados desta forma quanto a tamanho, localização, velocidade e aceleração". Desta forma, os veículos não são considerados como elementos discretos e com particularidades individuais, mas sim, componentes de um fluido contínuo que percorre as vias da rede.

Uma de suas características mais interessantes, reflexo da orientação prática que norteou o desenvolvimento do programa, é que o algoritmo se apóia tanto nas formulações teóricas como na bagagem adquirida pelos engenheiros de trânsito em sua atividade diária de programação semafórica.

A estratégia fundamental da operação do Siri é a de construir uma fotografia da rede a cada segundo. Tal procedimento exaustivo tem o inconveniente de incrementar significativamente o tempo de processamento, mas, em contrapartida, permite representar direta e objetivamente a situação do trânsito na rede estudada. O respeito ao tamanho finito das caixas entre semáforos fica extremamente simples de tratar e, conseqüentemente, pode-se dispensar a utilização de algumas expressões analíticas bastante questionáveis. Um subproduto interessante é que tal abordagem permitiu apresentar o resultado final ao usuário de uma forma bastante amigável. Outro ponto favorável que merece ser citado é que a abordagem pragmática de representar a circulação dos veículos a cada segundo facilita a incorporação de futuros módulos, como, por exemplo, a consideração de perfis de demanda variáveis ao longo do tempo.

A rede de semáforos é representada por duas entidades: os nós, que representam as interseções semaforizadas e os links, que representam os movimentos dos veículos chegando num semáforo.

O link pode ser do tipo "de entrada" ou "interno". O primeiro é aquele que provém de um local externo à rede e desemboca num de seus nós. O link interno parte de um denominado nó origem e chega a outro, denominado nó destino. Os veículos percorrem os links numa velocidade informada e se detêm quando encontram o nó destino numa situação de semáforo vermelho, ou então quando alcançam uma fila de veículos parados.

A figura 1 apresenta a tela principal do programa, com o desenho dos nós e links que esquematizam a rede de semáforos da Av. Nove de Julho.

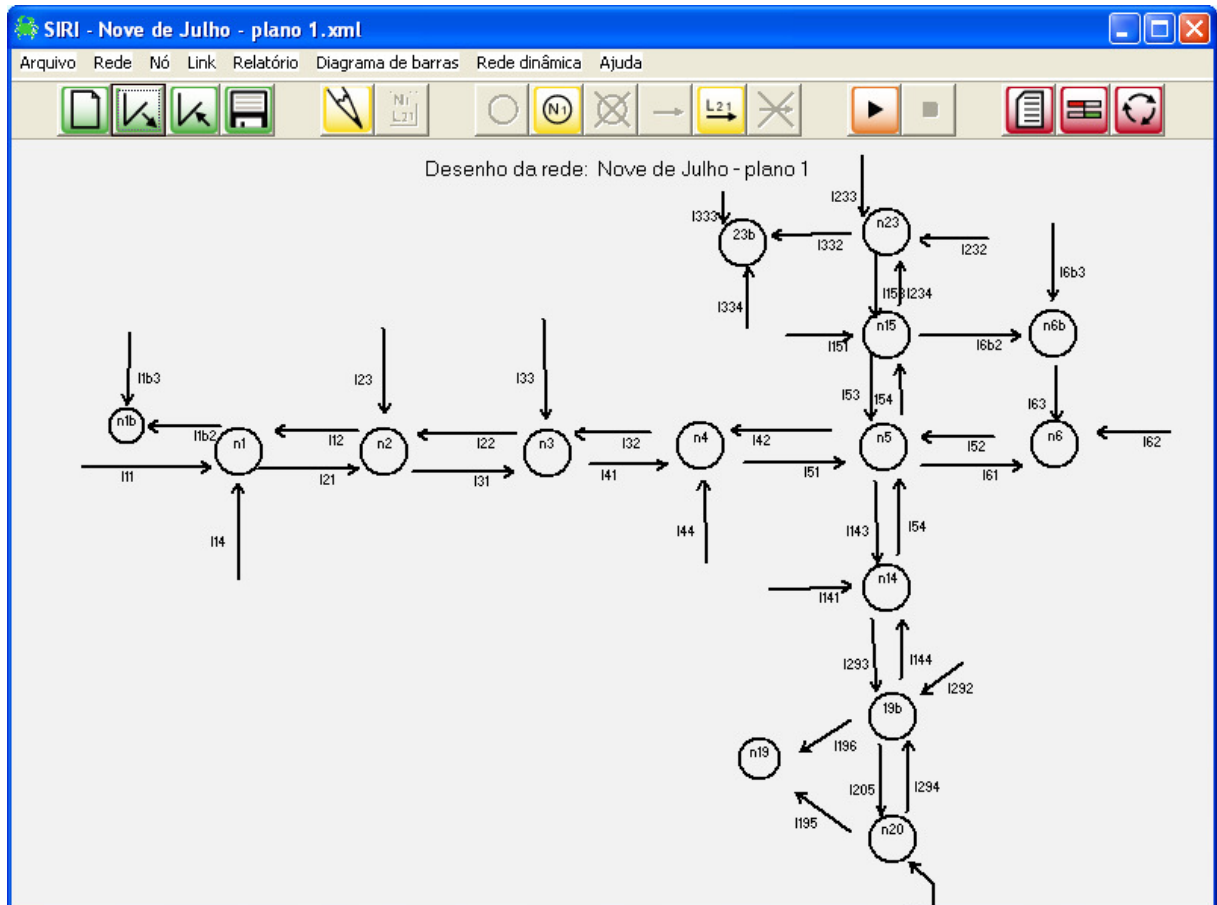


Figura 1

O processo é simulado durante um período de tempo variável, composto por duas etapas.

A primeira tem duração igual a oito vezes o tempo de ciclo calculado e tem a função de permitir que todos os veículos que vem de fora da rede possam preencher todos os links internos de forma a assegurar que o processo entrou num regime estável de distribuição de fluxos. A segunda etapa é, efetivamente, aquela na qual se contabilizam os atrasos e as paradas. Para situações não congestionadas, é igual ao número inteiro de ciclos que couber em 600 segundos. Para situações congestionadas, é igual ao número inteiro de ciclos que couber em 3600 segundos. Nos casos não congestionados, não é necessário ir além dos 600 segundos, pois os resultados passam a se repetir após este período. Entretanto, nos exemplos em que há supersaturação, é imprescindível processar a simulação durante um período maior a fim de

poder quantificar corretamente, durante a hora-pico, o crescimento constante das filas; o usuário pode, através de uma simples mudança de unidades, analisar situações em que o congestionamento ultrapasse o intervalo de uma hora.

Nos links de entrada, o programa adota o perfil de chegada uniforme. Tal distribuição mostrou-se muito mais realista do que outras que consideram fatores aleatórios. Nos links internos existem duas seções notáveis: uma seção a montante, logo no início do link e outra a jusante, coincidente com a faixa de retenção. O programa constrói o perfil do trânsito nessas duas seções, para todos os links da rede, durante o período de simulação, em passos de um segundo.

O perfil da seção a montante de um determinado link deriva do fluxo fornecido pelos links que contribuem para o link em pauta. O perfil da seção a jusante é conseguido, simplesmente, deslocando-se no tempo o perfil da correspondente seção a montante de um valor igual ao tempo de percurso daquele link. Optou-se pela não utilização do processo de dispersão veicular, pois a evolução dos perfis de tráfego observada nas vias urbanas indicou que a tendência de desmanche dos pelotões de veículos só ocorre a partir de distâncias superiores a 300 metros, quando o fator coordenação já perde muito de sua importância devido às entradas e saídas intermediárias.

O programa vai contabilizando a fila dos veículos em cada link, à medida que eles vão se acumulando, devido ao semáforo em vermelho ou devido à existência de veículos parados à frente. Siri analisa, também, a possibilidade de que os veículos de um link não consigam passar para o link seguinte devido à existência neste de uma fila que já ocupe toda a sua capacidade de armazenamento. Tal análise é fundamental quando se lida com situações de congestionamento.

A informação contínua da situação das filas permite calcular o número de paradas e o atraso em cada link, bem como a fila máxima que ocorreu durante o período de simulação e detectar um eventual perigo de "estouro de caixa".

A programação de uma rede de semáforos operando no modo "Tempos fixos" exige o cálculo de três diferentes conjuntos de valores. O primeiro é o tempo de ciclo, válido para todos os nós da rede. O segundo é o conjunto das distribuições, para cada nó, desse tempo de ciclo em tempos de verde / vermelho; os tempos de amarelo e vermelho de limpeza são fornecidos pelo usuário, pois são grandezas que implicam em análise de segurança viária. O terceiro conjunto é composto pelas defasagens entre os semáforos e tem a função de determinar em que instante as programações individuais de cada nó ocorrerão, em relação a uma base de tempo única para toda a rede.

O programa calcula o tempo de ciclo da rede através do método "Grau de saturação". Esta técnica permite definir diretamente a reserva de capacidade da rede em função de sua criticidade; é o método utilizado pelo sistema SCOOT de controle de semáforos em tempo real. Siri calcula o tempo de ciclo de modo que os nós mais críticos operem com um grau de saturação por volta de 88%. A escolha deste valor é um bom exemplo da contribuição da experiência dos técnicos da área. Deriva de aplicações práticas do programa Transyt e da observação do comportamento do trânsito durante a operação do programa Scoot. A adoção de um grau de saturação superior mostrou-se demasiadamente sensível à flutuação do trânsito

e à ocorrência de incidentes que restrinjam a capacidade, originando intervalos congestionados sem necessidade. Por outro lado, a escolha de um grau de saturação muito inferior a 88% implica na programação desnecessária de tempos de ciclo muito elevados, provocando incremento proporcional da espera dos veículos nos semáforos, tecnicamente denominada pelo termo “atraso”.

Em situações supersaturadas, não é possível obedecer ao patamar padrão do programa de 88%; nestes casos, a saturação pode chegar a ser maior, inclusive, do que 100%, configurando as situações congestionadas. Em tais casos, o tempo de ciclo é dimensionado no seu valor máximo, que é muito mais função de aspectos de conforto e de segurança do que da abordagem da fluidez. O valor padrão adotado pelo Siri para ciclo máximo é de 120 segundos, mas pode ser modificado, para cima ou para baixo, pelo usuário.

Em relação à determinação dos tempos de verde, o programa identifica, inicialmente, quais são os links críticos em cada um dos semáforos. Pode-se chegar a situações bastante complexas, quando não é biunívoca a correspondência entre links críticos e estágios. Em seguida, os tempos de verde são calculados segundo o critério da equalização dos graus de saturação entre os links críticos. Finalmente, os tempos de verde obtidos são confrontados com os tempos de verde de segurança parametrizados pelo usuário. Como não é permitido aceitar valores menores do que os de segurança, o programa recalcula as programações se valendo de outras expressões, caso isso venha a ocorrer.

As defasagens dos nós são escolhidas com o intuito de minimizar o parâmetro Índice de Desempenho, a seguir explicitado:

$$ID = \sum_{l=1}^{nl} A_l + K_p * \sum_{l=1}^{nl} NP_l \quad (1)$$

em que:

$ID$  - Índice de Desempenho da rede;

$l$  - cada um dos links da rede;

$nl$  - número total de links da rede;

$A_l$  - Atraso do link  $l$ ;

$NP_l$  - Número de paradas do link  $l$ .

Procura-se descobrir o valor mínimo da função  $ID$ , o que representa procurar os menores valores possíveis para os atrasos e para o número de paradas. Foi adotado o valor de 30 segundos para o parâmetro peso da parada, parâmetro este que traduz a relação de importância entre o prejuízo causado por uma parada e o prejuízo causado por um segundo de atraso.

Na versão 1.0, utilizou-se o método matemático denominado "hill-climbing" para efetuar a busca do ponto de mínimo da função. O grande problema a ser superado é o de não ser enganado pela escolha indevida de pontos de mínimo locais, que podem vir a apresentar desempenhos bem piores do que os correspondentes ao ponto de mínimo global. As primeiras etapas utilizam passos proporcionais ao tempo de ciclo. No fim, recorre-se a passos pequenos, de valores absolutos, para atingir um refinamento em torno do ponto escolhido.

A partir da versão 2.0, optou-se por substituir o processo "hill-climbing" por um procedimento de varredura mais detalhada a intervalos regulares. Tal alteração conduziu a melhores resultados, principalmente nas situações mais difíceis de pesquisar, como, por exemplo, a coordenação entre um semáforo de dois estágios com outro de três estágios.

Para construir o conjunto das defasagens da rede, Siri recorre à técnica utilizada pelos engenheiros em suas regulagens de campo. Em primeiro lugar, identifica-se a chamada rota principal da rede. O primeiro link desta rota é o link externo de maior fluxo. A partir daí, os maiores fluxos de contribuição vão configurando a seqüência dos links internos até que se chega ao último link, isto é, a um link que não contribui para mais ninguém. Uma vez conseguida a rota principal da rede, vão se construindo rotas secundárias, vinculadas à rota principal por, geralmente, um ou dois links. A idéia é a de linearizar uma malha fechada e estabelecer uma prioridade entre os nós, na definição de suas defasagens. Este processo reflete o encaminhamento, geralmente adotado, pelos técnicos em seu trabalho prático de regulação semafórica.

O programa calcula as defasagens para a rota principal, numa seqüência de análise dos nós idêntica àquela seguida na construção da rota. Fixam-se, então, os valores obtidos para esse conjunto de nós e parte-se para a determinação das defasagens de cada uma das rotas secundárias, utilizando-se procedimentos análogos.

O cálculo das defasagens é, sem dúvida, o ponto mais complexo de qualquer simulador de redes de semáforos coordenados, seja para aplicações do tipo "on-line" ou "off-line". O ideal seria poder calcular as defasagens, exaustivamente, para todas as combinações possíveis. Porém, tal escolha acarretaria numa proibitiva explosão de tempo de processamento. O busfílis da questão é, portanto, descobrir um método de busca inteligente que ache a melhor solução possível em, relativamente, poucas tentativas.

## ENTRADA E SAÍDAS DE DADOS

É interessante comentar, neste ponto, que a dificuldade em levantar dados confiáveis acaba, muitas vezes, prejudicando a imagem dos programas simuladores. Quando a programação calculada pelo simulador é ruim, a tendência é culpar a sua qualidade e não a dos dados inseridos. Quando a CET terminou o desenvolvimento do programa SIRI, a questão foi retomada. Lançar um simulador novo, sem cuidar do aspecto de levantamento de dados, seria, com certeza, condená-lo à execração. Procurou-se, então, criar um método de levantamento de dados que produzisse *inputs* confiáveis e que, simultaneamente, não exigisse muito esforço na pesquisa, o que não seria viável, dado os outros trabalhos exigidos. Este método pode ser encontrado no endereço [www.sinaldetransito.com.br/artigos/semáforo](http://www.sinaldetransito.com.br/artigos/semáforo).

É importante sublinhar a necessidade de que o programador acompanhe, pelo menos parcialmente, o desenrolar das pesquisas em campo, pois desta forma poderá aquilatar muito melhor tanto a qualidade como o significado dos dados registrados. É conveniente, inclusive, que o responsável pela programação também realize pessoalmente contagens e levantamentos, pois é uma ótima maneira de ganhar sensibilidade a respeito da rede que está encarregado de programar. Um programador experiente, numa vistoria, percebe uma série de fatores importantes que devem ser considerados. Mas se, além disso, ele toma uma atitude mais ativa

e participa das contagens, dos levantamentos dos tempos de percurso, das medições do fluxo de saturação, etc., ele vai conseguir refinar a percepção das características relevantes do local. Uma coisa é perceber algo qualitativamente. Mas quando você é obrigado a quantificar o que está vendo, a própria percepção qualitativa passa a ganhar outra dimensão.

Descrevem-se, a seguir, os dados que devem ser informados ao Siri, seja sob a forma de atributos dos nós ou dos links.

Em relação aos nós, devem-se configurar os grupos semafóricos, definindo a estrutura dos intervalos luminosos que compõem o diagrama de barras, a relação dos links que são controlados por cada grupo semafórico e os valores dos entreverdes e dos intervalos de limpeza. Informa-se, também, se o semáforo em questão pode operar com ciclo duplo, se é um nó escravo ou se possui botoeira de pedestres. Nestes dois últimos casos, é necessário preencher os respectivos parâmetros.

A figura 2 mostra a tela de entrada dos dados do nó que representa o semáforo da Av. Cruzeiro do Sul x Av. do Estado.

The screenshot shows the SIRI software interface for configuring a traffic node. The window title is "SIRI - Companhia de Engenharia de Tráfego CET / SP". The interface includes several input fields and checkboxes. At the top, there are fields for "Código do nó" (N7), "Quantidade de grupos semafóricos" (2), and a checkbox for "Ciclo duplo permitido". Below these are fields for "Nó escravo" (checked), "Código do nó mestre" (n6), "Intervalo escravo" (1), "Intervalo mestre" (1), and "Defasagem" (114). There is also a checkbox for "Nó com demanda de pedestres". The central part of the screen is a "Diagrama de barras" (bar chart) showing two horizontal bars with colored segments (green, yellow, red) and vertical dashed lines indicating time intervals. Below the diagram are two checkboxes and two input fields with the value "6". At the bottom right, there are "Cancelar" and "Inserir Dados" buttons.

Figura 2

No que tange às características dos links, pode-se identificar dois grupos de informações. O primeiro refere-se às condições fixas do local e engloba o fluxo de saturação, o tempo de percurso, seu comprimento, a capacidade de armazenamento de veículos e o menor tempo de

vede aceitável, sob o rótulo de verde de segurança. No segundo grupo, encontram-se o fluxo que chega ao semáforo e as contribuições de fluxo do link que está sendo editado para os links a jusante.

Existe, também, uma classe especial de dados de entrada que são as condições de contorno que o usuário pode impor ao programa, opcionalmente. Trata-se do tempo de ciclo, do tempo de ciclo máximo, do grau de saturação meta, do grau de saturação limite e do peso da parada.

Ao final do processamento, Siri disponibiliza os resultados nas seguintes formas de apresentação:

*Relatório de saída:* listagem onde são repetidos os dados de entrada e apresentados os resultados a que se chegou. Constam do relatório:

- a) para a rede: tempo de ciclo, número de paradas, atraso e índice de desempenho global;
- b) para cada um dos nós: programações calculadas (tempo de ciclo, tempos dos intervalos e defasagens);
- c) para cada um dos links: indicadores de trânsito resultantes (taxa de ocupação, grau de saturação, número de paradas, atraso, fila máxima e aviso de eventual "estouro" de caixa).

A figura 3 mostra parte do relatório de saída, com os resultados obtidos para os nós e links da rede.

**Informações dos nós**

Nome	Defas. (s)	Ciclo duplo	Duração dos intervalos (s)																	
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
n3	56	não	16	5	7	5	22	4	6	-	-	-	-	-	-	-	-	-	-	
n4	0	não	40	5	15	5	-	-	-	-	-	-	-	-	-	-	-	-	-	
n2	11	não	42	6	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
n5	64	não	37	6	17	5	-	-	-	-	-	-	-	-	-	-	-	-	-	
n1	28	não	20	5	17	5	14	4	-	-	-	-	-	-	-	-	-	-	-	

**Informações dos links**

Nome	Taxa de ocupação (%)	Grau de saturação (%)	Períodos de verde (s)				Fila máxima (veic)	Estouro de caixa	Atraso (veic*h/h)	Paradas (veic/h)
			Primeiro	Segundo	Terceiro	Quarto				
I32	21	48	28	-	-	-	10	não	3,6	632
I34	14	42	22	-	-	-	9	não	3,5	542
I42	40	64	40	-	-	-	5	não	1,7	619
I44	13	58	15	-	-	-	9	sim	1,8	359
I31	19	56	22	-	-	-	11	não	2,8	635
I52	41	71	37	-	-	-	17	não	4,7	908
I53	17	65	17	-	-	-	5	não	2,4	339
I13	17	79	14	-	-	-	14	não	6,4	823
I11	23	76	20	-	-	-	13	não	5,8	844
I21	54	83	42	-	-	-	14	não	3,2	878
I14	22	84	17	-	-	-	9	não	4	562

Figura 3



*Diagrama de barras final:* diagrama de barras resultante para cada nó; tem a função de auxiliar a análise dos resultados, de apoiar os serviços de campo e de ser utilizado para efeito de documentação.

*Formação dinâmica de filas:* É um dos produtos mais interessantes do Siri. O desenho da rede na tela vai sendo alterado, à medida que o tempo passa, representando o que ocorrerá na situação prática. Os links vão sendo coloridos de verde, amarelo ou vermelho e as filas vão sendo formadas e desmanchadas de acordo com os cálculos efetuados pelo programa. Tal visão dinâmica da rede facilita ao usuário uma visualização rápida do que será implementado em campo, contribuindo, principalmente, na questão da análise do jogo de defasagens.

## VALIDAÇÃO DO SIRI E SUAS APLICAÇÕES PRÁTICAS

Como ocorre com todo novo projeto, foi preciso confirmar a qualidade do programa. Esta etapa foi conseguida através de um processo de testes de simulação no computador, confrontando os resultados do Siri com os resultados obtidos por outras técnicas. As principais comparações levadas a cabo foram:

*Verificação contra situações teóricas* – em situações mais simples, como links de entrada ou primeiro link interno, é possível comparar os resultados do programa com a teoria, tanto no que se refere à programação dos tempos como no que se refere às filas, atraso e número de paradas decorrente. Nestes casos, o programa chegou aos mesmos valores apontados pela teoria.

*Comparação com Transyt na questão dos parâmetros calculados* – Transyt, programa lançado na Inglaterra há cerca de quarenta anos, é o software universalmente mais utilizado para calcular os tempos de redes semaforizadas; comparou-se os resultados fornecidos pelos dois programas, Transyt e Siri, e concluiu-se que, em situações não congestionadas, os dois foram bastante consistentes entre si. Em termos de tempo de ciclo, o módulo Cyop, do Transyt, recomendou valores 15%, em média, superiores aos do Siri. A diferença se deve, evidentemente, às diferentes metodologias utilizadas pelos dois algoritmos. A repartição dos tempos de verde foi exatamente igual; não poderia ser de outra forma já que a metodologia é exatamente a mesma. Na questão das defasagens, a comparação já é bem mais complexa. Os resultados variam bastante e fica difícil aquilatar qual é a melhor solução. A diferença provém do fato de que, apesar de tanto Transyt como a versão primeira do Siri utilizarem a técnica "hill-climbing", os passos utilizados eram radicalmente distintos. Todas essas comparações foram feitas no regime de não-congestionamento, pois o Transyt não incorpora o fator bloqueio de caixas à frente, o que impede o cotejo entre as duas soluções.

*Comparação com Transyt na questão dos indicadores* - para testar a correção do algoritmo do Siri no que concerne ao dimensionamento de filas, paradas e atraso, os dois programas foram submetidos aos mesmos dados de entrada e a uma mesma programação imposta. Os resultados obtidos foram praticamente os mesmos.

*Rede com características simétricas* – se a rede apresentar configuração e carregamento simétricos, os tempos de verdes e as defasagens também tem de apresentar tal peculiaridade. Esta propriedade foi uma das utilizadas para assegurar a exatidão do algoritmo utilizado.

As confrontações teóricas até aqui relatadas foram essenciais para ganhar confiança no novo programa, mas o teste de fogo sempre é, evidentemente, o desempenho na aplicação prática. Após quatro anos de utilização, pode-se dizer que a principal prova da qualidade do Siri é a sua aplicação em 250 semáforos paulistanos. Apresentam-se, a seguir, algumas experiências relevantes.

A primeira rede que recebeu a programação do Siri foi a da Av. Brigadeiro Luiz Antônio, próxima à Rótula, no início de junho de 2004. Nesta área foi realizada pesquisa do tipo antes-depois o que possibilitou constatar os ganhos alcançados.

As principais características desta rede são:

- Nove semáforos numa avenida de mão dupla, com propriedades de via coletora;
- Carregamento da avenida na hora-pico de, aproximadamente, 1000 veículos-equivalente num sentido, 700 no outro e 500 em cada uma das 7 transversais;
- Fluxo de ônibus em cada sentido, na hora-pico, de aproximadamente 90 veículos;
- Operação em regime não congestionado;
- Dois semáforos fora do eixo, com volumes bem inferiores aos outros sete;
- Três semáforos com conversão à esquerda na própria avenida;
- Seis travessias de pedestre semaforizadas na avenida, sendo que duas delas configuram um terceiro estágio específico no semáforo.

O trabalho foi realizado no período compreendido entre 7:00 e 20:00 h dos dias úteis, no qual foram aplicados seis planos de tráfego.

A pesquisa do tipo antes-depois foi executada com veículos percorrendo e medindo os tempos de percurso em três diferentes rotas da rede. Cada um dos sentidos da avenida constituiu uma rota; a terceira foi definida de modo a poder medir os tempos de espera das transversais. A programação que existia na situação "antes" tinha sido elaborada de forma empírica pelos técnicos de trânsito que trabalhavam na região. A Tabela 1 apresenta os resultados obtidos para cinco dos seis planos implementados. Lançou-se mão do teste de diferença de médias em distribuições tipo Student para efetuar a comparação entre os dos dados da situação "antes" com os da situação "depois".

**Tabela 1:** Rede da Av. Brig. Luiz Antônio - comparação entre as situações antes e depois

Confiança (%)	Intervalo (h)	Ciclo (s)		Tempo de percurso			Número de amostras	
		antes	depois	antes (s)	depois (s)	var (%)	antes	depois
77	06:00-09:20	80	80	323	286	-11	2	2
98	09:20-11:40	100	70	336	297	-12	8	8
92	11:40-13:20	120	85	310	287	-7	6	6
98	13:20-17:00	120	95	310	273	-12	10	10
97	17:00-18:00	90	75	323	283	-12	5	5
-	18:00-20:00	90	95	-	-	-	-	-

A leitura da Tabela 1 mostra resultados interessantes. Houve uma tendência para a redução dos tempos de ciclo, o que foi, sem dúvida, o fator mais influente na diminuição dos tempos

de percurso. Estes apresentaram uma queda persistente em torno de 10% em todos os planos. As variações relativas entre as medições das duas pesquisas estão apontadas na coluna "var". O grau de confiança obtido pelo método da diferença entre as médias para a distribuição de Student foi bastante elevado. Portanto, é altamente significativo afirmar que os conjuntos "antes" e "depois" são distintos entre si e que ocorreram ações que causaram tal variação. Como as pesquisas foram feitas em dias com características similares e com um intervalo entre elas de apenas duas semanas, pode-se afirmar, com alto grau de confiança, que a nova programação foi o fator responsável pela redução dos tempos de percurso.

Outro caso que merece ser citado é o da rede da Av. Cruzeiro do Sul / Av. do Estado. Trata-se de um eixo congestionado, com forte presença de caminhões. Antes da aplicação do programa, os links de entrada do eixo e algumas das transversais chegavam a ter filas de mais de 500 metros durante várias horas por dia. A nova programação reduziu o período supersaturado para cerca de três horas no total do dia e, mesmo assim, com filas inferiores a 150 metros. Análise posterior mostrou que redução do congestionamento tinha sido devida a uma interessante combinação entre a divisão dos tempos de verde e as defasagens. Defasagens mais adequadas nos link críticos do eixo permitiram reduzir seu percentual de verde que foi repassado às transversais. O próprio eixo teve um ganho significativo na eficiência do aproveitamento do verde devido ao jogo de defasagens implementado; isso foi muito importante devido à forte presença de caminhões pesado, situação extremamente susceptível à variação do parâmetro defasagem.

Em geral, o que se percebe em todas as redes estudadas é a redução do tempo de ciclo. Em média, as programações calculadas pelo Siri apresentaram tempos de ciclo 12% inferiores aos antigos. Como o atraso é diretamente proporcional ao tempo de ciclo, pode-se dizer, grosso modo, que este foi o benefício obtido em termos de espera semafórica.

## **CONCLUSÃO**

A disponibilidade de um simulador adaptado às nossas características abre as portas para um tratamento bem mais consistente das programações semafóricas do que aquele a que estamos habituados.

Com toda a certeza, o programa ainda é muito novo para poder ser considerado detentor da mesma solidez de outros que já operam há décadas. Existe um trabalho futuro de observação e, sem dúvida, de adequações tanto do algoritmo como da interface com o usuário. Apenas a aplicação prática numa grande quantidade de locais poderá transformar o Siri num programa plenamente adaptado às nossas necessidades. Aliás, é justamente essa a maior vantagem de desenvolver um programa próprio: a oportunidade de poder executar as adequações e melhorias requeridas.

Apesar de reconhecer a necessidade dessas futuras melhorias, os resultados até agora obtidos são extremamente animadores e permitem antever que, com a colaboração tanto dos profissionais da área teórica como dos engenheiros que lidam com a prática da temporização semafórica em seu dia a dia, será possível contar, num futuro próximo, com uma ferramenta eficiente e de fácil utilização para os trabalhos de programação de semáforos.

## REFERÊNCIAS BIBLIOGRÁFICAS

- Wohl, Martin e Martin, Brian. *Traffic System Analysis for Engineers and Plannings*. USA: McGraw-Hill Series in Transportation, 1967.
- Robertson, D. I. *Transyt: a traffic network study tool*. Crowthorne: RRL Report LR 253, Road Research Laboratory, 1969.
- Hunt, P. B., Robertson, D. I. e Winston, R. I. *SCOOT – a traffic responsive method of coordinating signals*. Crowthorne: TRRL Report LR 1014, Transport and Road Research laboratory, 1981.
- Akçelik, R. *Road traffic signaling*. London: Australian Road Research Board - internal report, 1982.
- Valdes, Antonio. *Ingenieria de Trafico*. Madrid: Libreria Editorial Bellisco, 1988.
- Salter, R. J. *Highway Traffic Analysis and Design*. London: MacMillan Education LTD, 1989.
- PORTUGAL, Licinio da Silva. *Simulação de Tráfego: Conceitos e Técnicas de Modelagem*. Rio de Janeiro: Editora Interciência, 2006.

## REFERÊNCIAS DE SITES

- Biblioteca da Federal Highway Administration - FHWA: [www.tfhrc.gov/library/library.htm](http://www.tfhrc.gov/library/library.htm)
- Center for Transportation Analysis ORNL: <http://cta.ornl.gov/cta/>
- Sinal de Trânsito: [www.sinaldetransito.com.br](http://www.sinaldetransito.com.br)
- Traffic Flow Theory - A State of the Art Report (FHWA): <http://www.tfhrc.gov/its/tft/tft.htm>