

# **SIRI - UM NOVO SIMULADOR PARA REDES DE SEMÁFOROS**

**Luis Molist Vilanova**

Engenheiro de Trânsito

Companhia de Engenharia de Tráfego da cidade de São Paulo

## **RESUMO**

Siri é um simulador macroscópico e determinístico, em forma de programa de computador, voltado ao cálculo de programações de semáforos coordenados a tempos fixos. O programa otimiza o desempenho dos semáforos no que tange ao atraso e número de paradas. Seu desenvolvimento foi concluído no início de 2004 e encontra-se na fase de implementação das primeiras aplicações práticas. 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.

## **ABSTRACT**

Siri is a macroscopic deterministic traffic signal simulation and optimization computer program for networks operating on fixed times. The program is used to optimize the performance of urban signal system with respect to delay and number of intersection stops. Its development was finished in the beginning of 2004 and now its initial applications are being put in practice. This paper outlines the reasons of its creation, the traffic model, the algorithm used to calculate signal timings and the facts observed to get confidence in its general use as a traffic tool.

## **INTRODUÇÃO**

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, existe 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 não se aplicam aos projetos de programação de redes coordenadas, principalmente devido à explosão do tempo de processamento.

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

A fim de poder contar com uma ferramenta mais adequada à nossa realidade, criou-se o programa de computador Siri - Simulador de Redes de Semáforos -, que calcula o tempo de ciclo, tempos de verde e defasagens para uma rede de semáforos com o

objetivo de causar os menores tempos de espera e o menor número de paradas possível para os veículos.

O desenvolvimento do programa demorou quatro anos e sua primeira versão ficou disponível em janeiro de 2004. Recentemente, ocorreu sua primeira aplicação prática, numa rede de 9 semáforos coordenados e os resultados obtidos tem-se mostrado bastante promissores, como pode ser verificado no item 7 deste trabalho.

## **1. O QUE É O SIRI ?**

O Siri é um programa de computador de simulação de redes de semáforos a 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 semaforicos e os tempos de entreverdes.

O programa pode ser classificado como macrosimulador. Os veículos não são considerados como indivíduos discretos e com particularidades segundo algum critério de classificação, mas sim, componentes de um fluido contínuo que percorre as vias da rede. É um simulador do tipo determinístico: uma vez inseridos os dados de entrada, é único o caminho do algoritmo bem como o conseqüente resultado final. Uma de suas características mais interessantes é que se apoia tanto nas formulações teóricas como na bagagem adquirida pelos engenheiros de trânsito em sua atividade prática de programação semaforica.

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, simplesmente, copiar a realidade segundo a segundo, permite que novos módulos venham a ser facilmente integrados ao programa no futuro.

## **2. O MODELO DE TRÁFEGO**

A rede é formada por duas entidades: os nós, que representam as interseções semaforizadas e os links, que representam os fluxos unidirecionais de veículos entre dois nós.

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 nó

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.

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

A primeira etapa é igual a oito vezes o tempo de ciclo escolhido; 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 links de entrada, adotou-se um 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 mostrou pouquíssima tendência de desmanche dos pelotões de veículos.

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 antes da retenção. O 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 sua capacidade de armazenamento. Tal análise é fundamental quando lidamos 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 "estouro de caixa".

### **3. CÁLCULO DOS PARÂMETROS**

Existem três diferentes conjuntos de valores a serem determinados. 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. O terceiro é denominado conjunto das defasagens e tem a função de determinar em que

instante as programações individuais de cada nó ocorrerão, em relação a uma base de tempo única.

### 3.1 Determinação do tempo de ciclo

Para a escolha do tempo de ciclo da rede, utiliza-se o método do grau de saturação. O Siri calcula o tempo de ciclo de modo que os nós mais críticos operem com um grau de saturação por volta de 88%. O programa aceita que alguns nós trabalhem com valores superiores, mas nunca rompendo o valor limite de 95%. Evidentemente, em situações congestionadas, esses valores são ultrapassados por mais que se aumente o tempo de ciclo.

A escolha dos valores utilizados é um bom exemplo da contribuição da experiência dos técnicos da área. Deriva de aplicações práticas do programa Transyt, mas principalmente da observação do comportamento do trânsito durante a operação do Scoot, programa de controle de semáforos em tempo real.

### 3.2 Determinação dos tempos de verde

Em primeiro lugar, o programa identifica quais são os links críticos em cada estágio do ciclo. 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.

### 3.3 Determinação do jogo de defasagens

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

$$ID = \sum A_{link} + K_p * \sum NP_{link} \quad (1)$$

em que ID: Índice de Desempenho da rede;

$\sum A_{link}$ : somatório dos atrasos de todos os links da rede;

$K_p$ : peso da parada; e

$\sum NP_{link}$ : somatório do número de paradas de todos os links da rede.

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.

Utilizou-se o método "hill-climbing" para efetivar a busca do ponto de mínimo. 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.

Para construir o conjunto das defasagens da rede, o Siri se valeu da 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.

O programa calcula, então, as defasagens para a rota principal, numa seqüência de análise dos nós idêntica àquela seguida na construção da rota. Fixa-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 polêmico 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 situações possíveis. Mas essa escolha acarretaria numa proibitiva explosão de tempo de processamento.

## **4. ENTRADA DE DADOS**

### **4.1. Parâmetros obrigatórios**

Devem ser fornecidos, para o programa, os valores dos seguintes parâmetros:

*Para cada um dos links:*

- fluxo de veículos
- fluxo de saturação
- nó origem e nó destino
- tempo de verde de segurança
- tempo de percurso (se link interno)
- capacidade de armazenamento de veículos (se link interno)
- os links que recebem veículos do link em pauta e os correspondentes fluxos de contribuição

*Para cada um dos nós:*

- quantidade de grupos semaforicos

- diagrama de barras, onde é definida a correspondência entre links e grupos semaforicos bem como a seqüência dos estágios
- tempos de amarelo e de vermelho de limpeza
- quantidade de acionamentos de botoeiras de pedestres (para nós com este tipo de demanda)

#### **4.2. Parâmetros opcionais**

Podem ser impostos ao programa os seguintes parâmetros:

*Para a rede:*

- tempo de ciclo
- grau de saturação meta
- grau de saturação limite
- peso da parada

*Para cada um dos nós:*

- duração de estágios, para estabelecer, de antemão, defasagens internas em nós complexos
- defasagem fixa em relação a um nó configurado como mestre da rede
- proibição da utilização de ciclo duplo

### **5. PRODUTOS DO PROGRAMA**

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

*Relatório de saída:* listagem onde são repetidos os dados de entrada e são 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).

*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 subprodutos mais interessantes do Siri. Mostra na tela do monitor o desenho da rede, com seus links em verde, amarelo ou vermelho à medida que transcorre o período de simulação. Exibe, também, a formação e o desmanche das filas, como um traço proporcional à capacidade de armazenamento do link. 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.

## **6. VALIDAÇÃO**

A confirmação da boa qualidade do algoritmo 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 trabalhadas 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 aos elementos da programação dos tempos como no que se refere às filas, atraso e número de paradas. Nestes casos, o programa chegou aos mesmos valores apontados pela teoria.

*Comparação com Transyt na questão dos parâmetros calculados* – os resultados fornecidos pelos dois programas, Transyt e Siri, foram bastante consistentes entre si. Em termos de tempo de ciclo, houve uma variação em torno de 15% entre o valor alcançado pelo novo programa e o valor recomendado pela rotina Cyop do Transyt. 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 dos dois programas se apoiarem na técnica "hill-climbing", os passos utilizados são radicalmente distintos. Todas essas comparações foram feitas no regime de não-congestionamento pois o Transyt não chega a trabalhar com o fator bloqueio de caixas à frente.

*Comparação com Transyt na questão dos indicadores* - para testar a correção do algoritmo do Siri, os dois programas foram submetidos aos mesmos dados de entrada e a uma mesma programação imposta. Os resultados dos parâmetros número de paradas, atrasos e filas foram praticamente os mesmos, o que assegurou que o algoritmo estava construído corretamente, uma vez que o padrão de comparação - o Transyt - é o aplicativo mais utilizado mundialmente para efetuar o cálculo de programações de redes de semáforos.

*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. Nos primeiros testes, o programa não foi capaz de chegar a defasagens simétricas. O algoritmo foi modificado, então, para poder lidar adequadamente com tal situação.

*Aplicação em campo* – o item 7, a seguir, discorre sobre os resultados alcançados na primeira aplicação prática.

## 7. RESULTADOS OBTIDOS NA PRIMEIRA APLICAÇÃO PRÁTICA

A primeira implementação das programações calculadas pelo Siri ocorreu no início de junho de 2004 em 9 semáforos coordenados da Av. Brigadeiro Luiz Antônio. As principais características da rede são:

- avenida de mão dupla, com propriedades de via coletora, onde se encontram 7 semáforos
- o carregamento da avenida na hora-pico é de, aproximadamente, 1000 veículos-equivalente num sentido, 700 no outro e 500 em cada uma das 7 transversais
- o fluxo de ônibus em cada sentido, na hora-pico, é de aproximadamente 90 veículos
- existem dois semáforos fora do eixo, com volumes bem inferiores aos outros sete
- existem três semáforos com conversão à esquerda na própria avenida
- há seis travessias de pedestre semaforizadas na avenida, sendo que duas delas configuram um terceiro estágio específico no semáforo

Trabalhou-se o período compreendido entre 7:00 e 20:00 h dos dias úteis, no qual foram aplicados seis planos de tráfego.

Efetuuou-se uma pesquisa do tipo antes-depois, 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 cada um dos planos a menos do último que não foi verificado pela pesquisa. 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

Intervalo (H)	Ciclo		Tempo de percurso			Número de amostras		Confiança (%)
	antes	depois	antes (s)	depois (s)	var (%)	antes	depois	
06h00-09h20	80	80	323	286	-11	2	2	77
09h20-11h40	100	70	336	297	-12	8	8	98
11h40-13h20	120	85	310	287	-7	6	6	92
13h20-17h00	120	95	310	273	-12	10	10	98
17h00-18h00	90	75	323	283	-12	5	5	97
18h00-20h00	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 apontados 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.

## **8. 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 da primeira aplicação prática na Av. Brigadeiro Luiz Antônio 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.