

SP 05/97

NT 201/97

Uma breve descrição do Sistema SCOOT

Eng^o Sun Hsien Ming

1. Introdução

Este trabalho tem por objetivo divulgar ao corpo técnico da CET não diretamente envolvido com a implantação das CTAs (ou dos “semáforos inteligentes”) os princípios básicos do sistema SCOOT, ora implantado na CTAs 1, 2 e 5.

O sistema SCOOT em implantação na CTA 1 é fornecido pela firma inglesa *Siemens-Plessey*, enquanto que o implantado nas CTAs 2 e 5 é fornecido pela também inglesa *Peek Traffic*. Embora fornecido por empresas diferentes, existindo, portanto, várias diferenças entre si, os princípios básicos dos dois sistemas SCOOT são semelhantes.

Neste trabalho, será feita uma descrição do SCOOT baseada no sistema da *Peek Traffic*, ora em implantação nas CTAs 2 e 5.

Nas CTAs 2 e 5 está previsto, até o final da implantação do projeto, o controle em SCOOT de 279 interseções (controladas por 233 controladores) na CTA 2 e 180 interseções (controladas por 143 controladores) na CTA 5.

2. Controle de tráfego em tempo real

A sigla SCOOT significa “*Split Cycle Offset Optimization Technique*”, onde *Split* significa porcentagem ou fração de verde (note que não é a duração de tempo de verde em segundos), *Cycle* é o tempo do ciclo semafórico e *Offset* significa defasagem. Assim, SCOOT significa “Técnica de Otimização da Porcentagem de Verde, Ciclo e Defasagem”.

SCOOT é um sistema de controle de tráfego em tempo real, na medida em que variam os tempos semafóricos (porcentagem de verde, ciclo e defasagem) em função do fluxo de tráfego a cada momento. Ao contrário do controle de tráfego em tempos fixos, onde os tempos semafóricos são invariáveis, o sistema de controle em tempo real

reage às variações de tráfego de forma a adequar os tempos semafóricos ao tráfego em cada instante.

Convém lembrar aqui a diferença entre o sistema de controle em tempo real com sistemas atuados e com sistemas de seleção dinâmica de planos.

Nos sistemas atuados, os tempos semafóricos também não são fixos, variando de um mínimo a um máximo. Porém, neste caso, não existe um cálculo de otimização visando minimizar os atrasos e o número de paradas na área sob controle. O sistema atuado simplesmente, a partir de um tempo de verde mínimo, prolonga o verde à medida que detecta a aproximação de mais veículos até atingir o verde máximo.

Os sistemas de seleção dinâmica de planos são apenas uma variação do sistema de controle em tempos fixos. Ao invés dos planos serem selecionados por uma tabela horária, eles são selecionados por uma tabela de decisão baseada em parâmetros de tráfego. Assim, esses sistemas continuam baseados em planos de tempo fixo.

3. Estrutura básica do SCOOT

O sistema SCOOT é constituído basicamente por 5 partes:

- Sistema de detecção de veículos;
- Sistema de transmissão de dados;
- Computador central;
- Controladores
- Grupos focais.

O funcionamento básico está ilustrado na Figura 1.

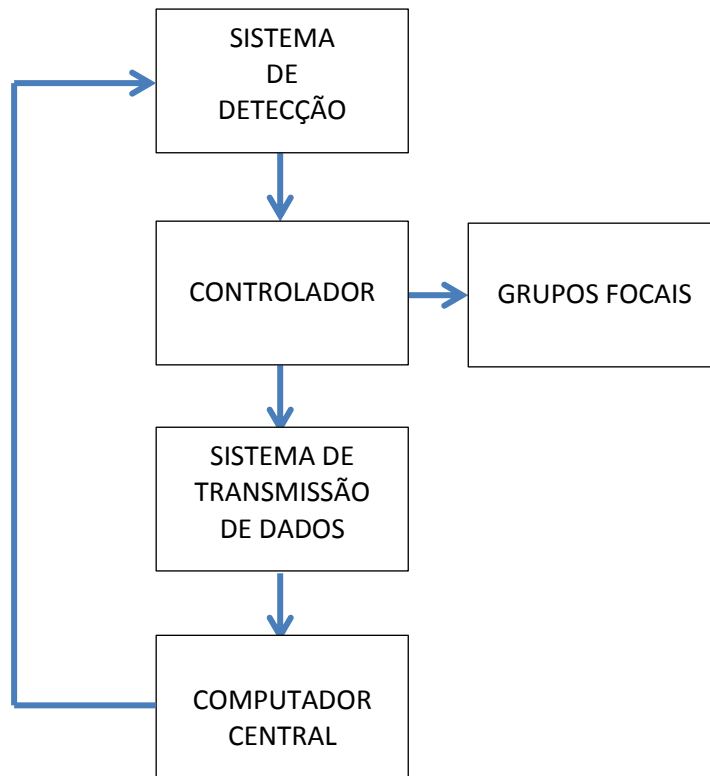


Figura 1

O sistema de detecção detecta o fluxo de veículos, os dados são recolhidos pelo controlador que os envia através do sistema de transmissão de dados ao computador central. O computador central processa as informações e calcula os tempos semafóricos e envia ao controlador, que os implementa nos grupos focais.

4. Hierarquia de controle

A hierarquia de controle do sistema SCOOT está representada na Figura 2.

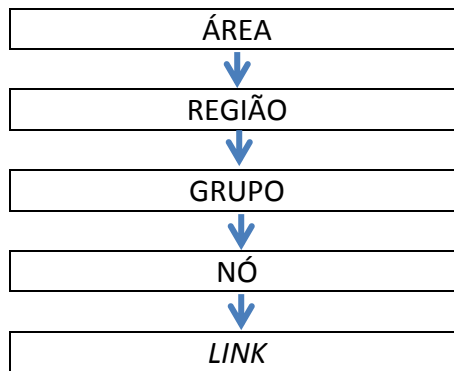


Figura 2

A *ÁREA* é toda a área sob controle do SCOOT.

A *ÁREA* é dividida em *REGIÕES*. Uma região é um grupo de semáforos (chamados de *NÓS*), que operam com o mesmo tempo de ciclo e são sincronizados ou coordenados entre si através de defasagens.

Uma *REGIÃO* pode conter um ou mais *GRUPOS*. Um *GRUPO* é composto por 2 ou mais *NÓS*. Cada *NÓ* componente de um *GRUPO* pode ser designado de “mestre” ou “escravo”. Quando um “mestre” cai para o modo local, automaticamente todos os *NÓS* do *GRUPO* caem junto com ele. Quando um “escravo” cai para o modo local, este fato não interfere com a operação de outros *NÓS*.

O *NÓ* é a menor unidade de controle. Normalmente ele corresponde a uma interseção semaforizada.

Um *NÓ* é composto de *LINKS*. Um *LINK* corresponde a uma aproximação do cruzamento semaforizado.

5. Modelo de tráfego

A parte nevrálgica do sistema, onde está alojada a sua “inteligência”, é baseada num modelo de tráfego. Em termos simplificados, o modelo baseado no fluxo de veículos detectado 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 detectado pelo sistema de detecção e outro que é projetado pelo modelo de tráfego na linha de retenção.

O perfil detectado pelo sistema de detecção é denominado de “*Cyclic Flow Profile*”.

Um exemplo de *Cyclic Flow Profile*, obtido a partir do sistema de detecção, está ilustrado na Figura 3.

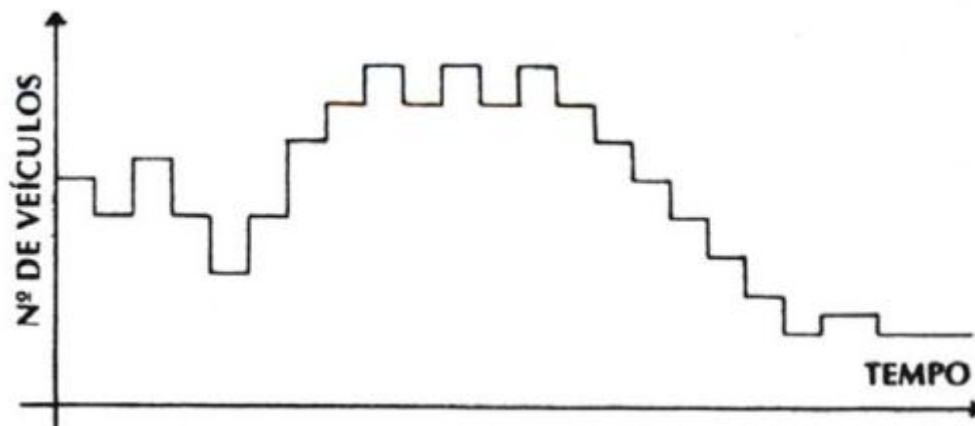


Figura 3

Baseados no *Cyclic Flow Profile* de cada *link*, é possível ao sistema SCOOT calcular os atrasos e o número de paradas para diversos conjuntos de valores de tempos semafóricos, escolhendo aquele que minimize o atraso e o número de paradas. Esses tempos semafóricos, assim obtidos, são implementados imediatamente, a tempo de favorecer os veículos que foram detectados e que compuseram o *Cyclic Flow Profile*. Daí o sistema SCOOT ser totalmente responsivo, pois os tempos semafóricos são calculados para escoar os veículos que estão sendo detectados no momento e não para os veículos que chegam no ciclo seguinte, por exemplo. A Figura 4 ilustra este fato.

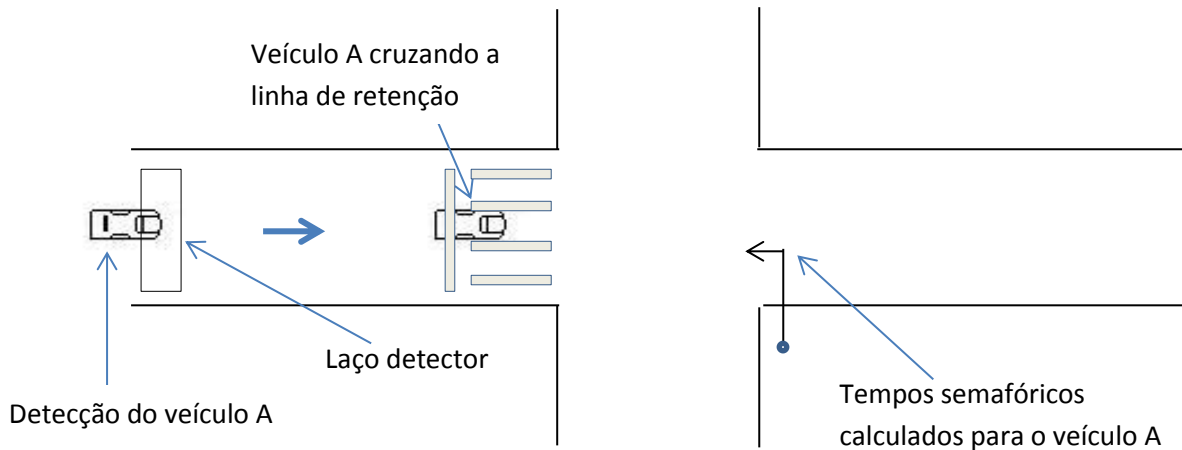


Figura 4

Assim, a detecção dos veículos deve ser feita obrigatoriamente antes de o veículo chegar à linha de retenção, para que haja tempo suficiente para o sistema SCOOT poder fazer todo o processamento necessário. Normalmente, a detecção deve ser feita 8 a 12 segundos antes de o veículo chegar na linha de retenção.

Com base no *Cyclic Flow Profile* é feito o processo de otimização. O processo de otimização do sistema SCOOT é feito através de três otimizadores: otimizador de *Split*, otimizador de *Offset* e o otimizador de *Cycle*. O funcionamento desses otimizadores está descrito na Seção 8 deste trabalho.

O sistema 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”. 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 o otimizador de *Split* usa o “*Long Term Cycle Profile*”), enquanto que os otimizadores de *Offset* e de *Cycle* usam o “*Long Term Cycle Profile*”.

6. Sistema de detecção

Normalmente, a detecção dos veículos é feita através de laços detectores colocados no pavimento.

A cada $\frac{1}{4}$ de segundo o sistema de detecção verifica se o laço está ocupado ou não.

Exemplo:

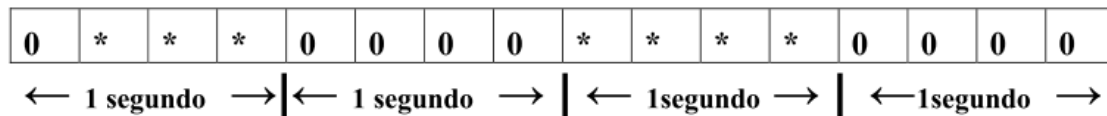


Figura 5

onde:

0 = laço não ocupado

* = laço ocupado

Assim, a detecção é feita em intervalos de $\frac{1}{4}$ de segundo (250 milissegundos).

O que se detecta não é exatamente o fluxo ou o volume de veículos (em veículos por segundo ou por hora) e nem a porcentagem de tempo de ocupação do laço.

O que se detecta é uma mistura de ambos, cuja unidade é expressa em LPU (*Link Profile Unit*).

O número de LPUs é obtido da seguinte forma:

- O 1º intervalo de $\frac{1}{4}$ de segundo com “*” (ocupado) após um intervalo com “0” (não ocupado) equivale a 7 LPUs.
- O 2º equivale a 6 LPUs.
- O 3º equivale a 5 LPUs.
- O 4º equivale a 4 LPUs.
- O 5º equivale a 3 LPUs.
- O 6º equivale a 2 LPUs.
- O 7º equivale a 1 LPU.
- Do 8º em diante cada intervalo vale 1 LPU.

Para melhor compreensão deste cálculo, considere-se um exemplo de detecção de 3 veículos em um período de 6 segundos (24 intervalos de $\frac{1}{4}$ de segundo), conforme mostra a Figura 6.

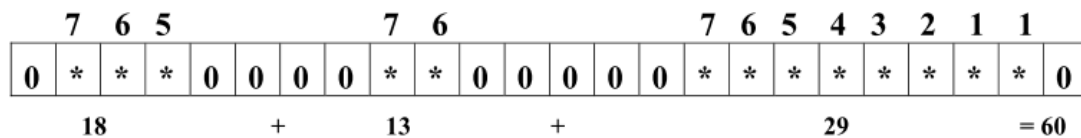


Figura 6

No exemplo da Figura 6, os 3 veículos representam 60 LPUs, o que dá uma média de 20 LPUs por veículo.

No exemplo:

- Em termos de fluxo de tráfego:

Fluxo = 3 veículos/6 segundos = 0,5 veíc/seg = 1800 veíc/hora.

- Em termos de porcentagem de tempo de ocupação:

Ocupação = 13 intervalos / 24 intervalos = 54%

- Em termos de LPU:

60 LPUs / 6 segundos = 10 LPU/s

Desta forma, a medida em LPUs é função do fluxo de veículos, da velocidade, do comprimento dos veículos e da disposição física do laço na via. Assim, não existe uma equivalência fixa e matemática entre o número de veículos e o número de LPUs, uma vez que, a rigor, a equivalência entre as duas medidas varia de local e de momento para momento, pois depende do comportamento e composição do tráfego em cada local e em cada momento. Em termos médios, pode-se considerar que um veículo é equivalente a aproximadamente 17 LPUs.

Tendo em vista não haver uma equivalência fixa entre o número de veículos e a medida em LPUs, os detectores SCOOT não são apropriados para efetuar contagem de veículos. Entretanto, é possível obter uma contagem aproximada utilizando-se dos detectores SCOOT desde que se aplique um fator de correção. Esse fator pode ser obtido fazendo-se uma contagem confiável por outro meio (por exemplo, contagem manual).

Todos os parâmetros de tráfego utilizados pelo sistema SCOOT são expressos em LPU.

Por exemplo, o fluxo de saturação (denominado *Saturation Occupancy* – SATO no sistema SCOOT da *Peek Traffic*) é medido em LPU/s, a fila (*Queue*) é expresso em LPU, etc.

Situações de congestionamentos podem ser identificadas diretamente através de detectores SCOOT.

Para tanto, deve-se locar o laço em uma posição tal que o fim da fila normal do semáforo (fila de espera semaforica quando o semáforo está vermelho) não esteja sobre o laço. Dessa forma, se por determinado período o laço estiver permanentemente ocupado, isto significará que a aproximação está congestionada. Pode-se definir um índice de congestionamento conforme descrito abaixo.

Define-se intervalo congestionado como sendo um intervalo de 4 segundos (16 intervalos de ¼ de segundo) em que o laço esteja permanentemente ocupado, conforme Figura 7.

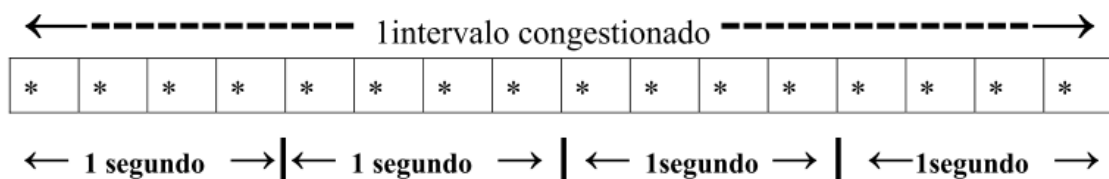


Figura 7

Define-se o índice de congestionamento como sendo:

$$\text{Índice de congestionamento} = \frac{4 \times N}{C}$$

onde:

N = número de intervalos congestionados durante o ciclo

C = tempo de ciclo

Em caso de falha de laço, o sistema SCOOT passa a operar com *Split* fixo para o *link* cujo laço está em falha. O valor do *Split* é tirado do plano de tempos fixos configurado na tabela horária.

Os detectores SCOOT podem apresentar três estados de operação: “limpo” (normal), “suspeito” e “falha”.

As otimizações do SCOOT só ocorrem quando o detector está “limpo”. Se após um determinado tempo não houver detecção (“vazio”) ou houver detecção contínua (“cheio”), o detector irá para “suspeito”, as otimizações param e, se a situação “cheio” ou “vazio” continuar, ele vai para “falha”. Se os veículos começarem a passar novamente sobre o laço normalmente, ele volta automaticamente (após 5 minutos) de “suspeito” para “limpo”. Quando o detector já estiver na condição de “falha”, ele somente sairá deste estado com um comando do operador (ou com um comando configurado em tabela horária). Os tempos de mudança de estado dos detectores SCOOT estão mostrados na Figura 8 a seguir:



Figura 8

Esses tempos podem ser alterados no sistema.

7. Modelo de fila

Para explicar como o sistema SCOOT faz o modelamento de fila, apresenta-se um exemplo conforme o Quadro 1 abaixo:

Quadro 1:

IVL	OCC	Q	Fluxo de chegada	Fluxo de saída	R / G
1	16	22	22	0	RRRR
2	18	40	40	0	RRRR
3	2	42	42	0	RRRR
4	16	58	58	0	RRRR
5	2	60	60	0	RRRR
6	16	66	76	10	RRRG
7	2	28	78	50	GGGG
8	18	6	96	90	GGGG
9	18	- 16	114	114	GGGG
10	2	- 44	116	114	GGGR

onde:

IVL = intervalo de 4 segundos;

OCC = ocupação medida pelos detectores em LPU, já projetado pelo perfil cíclico para a linha de retenção, considerando-se o platô e a dispersão;

Q = Fila acumulada em LPU ($Q = Q$ do intervalo anterior + *OCC* do intervalo em questão, quando o semáforo está no vermelho).

Por exemplo:

Para o intervalo 2: $Q = 22 + 18 = 40$ LPUs

Fluxo de chegada = fluxo acumulado de veículos que chegam na linha de retenção em LPU;

Fluxo de saída = fluxo acumulado de veículos que saem quando o sinal está em verde em LPU.

R/G = R (*Red*) vermelho e G (*Green*) verde:

Cada letra (R ou G) indica 1 segundo.

Supondo que o fluxo de saturação (*Saturation Occupancy*) seja 10 LPU/seg, no intervalo 6, quando o semáforo abre o verde, a fila fica:

$$Q = 60 + 16 - 10 = 66$$

No intervalo 7, a fila é:

$$Q = 66 + 2 - 4 \times 10 = 28$$

No intervalo 9, a fila fica negativa, indicando uma sobra de verde (quem chega passa):

$$Q = 6 + 18 - 4 \times 10 = -16$$

Número de paradas (em LPUs) = soma dos números da coluna *OCC* quando o semáforo está vermelho.

Com base nesse modelo de fila, o sistema SCOOT constrói o *Vega Profile*. A Figura 9 representa o *Vega Profile* do exemplo mostrado no Quadro 1:

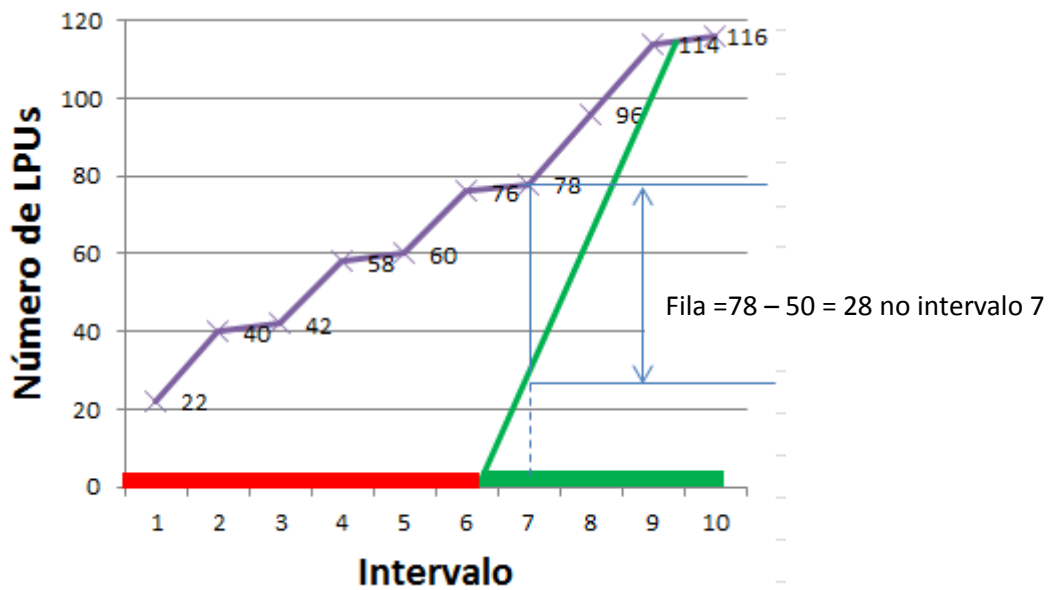


Figura 9

8. Otimizadores

As características básicas dos otimizadores estão resumidas no Quadro 2 abaixo:

Quadro 2

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

8.1. Otimizador de Split

O otimizador de *Split* atua 5 segundos antes do instante previsto de mudança de estágio. Cinco segundos antes do instante previsto para a mudança de estágio, uma decisão deverá ser tomada pelo otimizador: se encurta o estágio em 4 segundos (- 4), se mantém a mesma duração do estágio (0) ou se prolonga o estágio em 4 segundos (+ 4). A Figura 10 mostra um exemplo onde a duração prevista do estágio é de 26

segundos. A decisão é tomada no instante 21. Se a decisão for encurtar o estágio, ele terminará no instante 22 ($26 - 4 = 22$); se a decisão for manter a mesma duração, o estágio terminará no instante 26 (não há mudança); e se a decisão for prolongar o estágio, ele terminará no instante 30 ($26 + 4 = 30$).

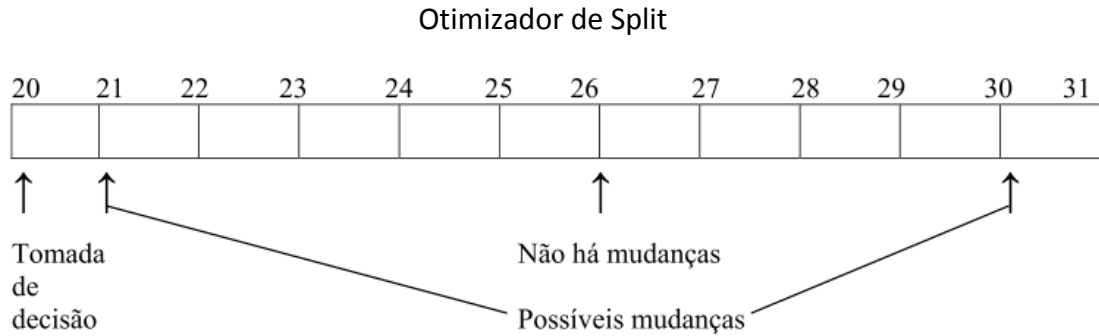


Figura 10

Se a decisão tomada foi a de encurtar o estágio, no ciclo seguinte o instante previsto para a mudança do estágio será 25 ($26 - 1 = 25$) e a nova decisão deverá ser tomada no instante 20, podendo o estágio, nesse novo ciclo, terminar no instante 21, 25 ou 29, conforme se a nova decisão for encurtar, manter ou prolongar o estágio. Ver a Figura 11.

Otimização do *Split* no ciclo seguinte se a decisão no ciclo anterior foi encurtar o verde



Figura 11

Se a decisão no primeiro ciclo foi prolongar o estágio (e não encurtar), no ciclo seguinte o instante previsto para a mudança de estágio será 27 ($26 + 1 = 27$) e a nova decisão deverá ser tomada no instante 22, podendo o estágio, nesse novo ciclo, terminar no instante 23, 27 ou 31, conforme se a nova decisão for encurtar, manter ou prolongar o estágio. Ver a Figura 12.

Otimização do *Split* no ciclo seguinte se a decisão no ciclo anterior foi prolongar o verde

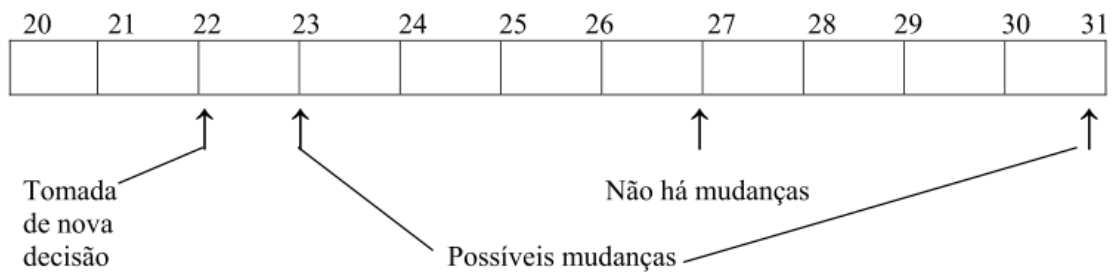


Figura 12

A decisão de encurtar, manter ou prolongar o estágio é feita visando minimizar o quadrado do maior grau de saturação nos *links* do nó. A seguir, é dado um exemplo de um nó com 3 *links* e 2 estágios, conforme Quadro 3, abaixo:

Quadro 3

LINK	ESTÁGIO
A	1
B	2
C	3

No exemplo, vamos supor que o estágio que está sendo otimizado é o estágio 2. Para otimização, o sistema calcula o grau de saturação e o seu quadrado. O Quadro 4 mostra os valores calculados para cada *link*, nas 3 hipóteses: prolongando, mantendo e encurtando o verde do estágio 2.

Quadro 4

LINK	PROLONGA		MANTÉM		ENCURTA	
	x	x^2	x	x^2	x	x^2
A	0,80	0,64	0,70	0,49	0,60	0,36
B	0,85	0,72	0,90	0,81	0,95	0,90
C	0,45	0,20	0,55	0,30	0,65	0,42
MÁXIMO		0,72		0,81		0,90

x = Grau de saturação

DECISÃO: o menor dos máximos (0,72; 0,81; 0,90) é 0,72 da coluna “PROLONGA”. Assim, a decisão a ser tomada é de prolongar o verde do estágio 2 em 4 segundos.

Supondo que inicialmente o estágio 2 terminaria no instante 10 do ciclo, com a decisão de prolongar o verde do estágio 2, o mesmo terminará no instante 14.

8.2. Otimizador de *Offset*

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.

A decisão do otimizador de *Offset* é tomada com base no menor valor de PI (*Performance Index*):

$$PI = \text{atraso} + \text{índice de paradas} + \text{índice de congestionamento}$$

O índice de paradas é definido como sendo o número de paradas multiplicado por um fator chamado *Stop Penalty*.

$$\text{Índice de paradas} = \text{número de paradas} \times \text{Stop Penalty}$$

O Quadro 5 mostra exemplo de um nó com 4 *links* onde são mostrados os PIs dos *links* calculados para as três hipóteses: aumentar, manter ou diminuir o *Offset*.

Quadro 5

LINK	AUMENTA	MANTÉM	DIMINUI
A	126	121	120
B	82	78	8
C	143	139	141
D	96	95	96
SOMA	447	432	447

O menor PI é 432 na coluna MANTÉM. Logo, a decisão do otimizador deve ser: manter o atual valor de *Offset*.

8.3. Otimizador de Ciclo

Normalmente o otimizador de ciclo roda 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.

Passos de incremento/decremento:

- 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 nó é calculado um ciclo chamado “ciclo ideal” (*Ideal Node Cycle Time*). O ciclo ideal do nó é calculado em função do nível de saturação desejado para o *link* mais saturado do nó. O nível de saturação desejado é denominado *Target Saturation*.

Normalmente, o *Target 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 Saturation* = 90%

C = tempo do ciclo corrente

LT = tempo morto (*Lost Time*)

NS = nível de saturação do *link* mais saturado do nó.

Se qualquer *link* do nó 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* de nó 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 abaixo um exemplo:

TS = 90%

C = 120

LT = 15

NS = 80%

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

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 – MPCY*) de um nó 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:

INCT = 67,5

MPCY = 68

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

Exemplo:

Seja uma região com 4 nós.

- Nó 1 → $MPCY = 88$
- Nó 2 → $MPCY = 80$
- Nó 3 → $MPCY = 48$
- Nó 4 → $MPCY = 48$

Ciclo da Região = 88

Se houver mudança de ciclo, o sistema repete pelo menos uma vez o ciclo anterior, antes da mudança.

Exemplo:

Sequência de ciclos numa região: 80, 80, 88, 88, 80, 80, 88.

Ciclo Duplo

O otimizador pode implementar ciclo duplo em alguns nós da Região. A implementação do ciclo duplo é explicada através de dois exemplos:

Exemplo 1:

Quadro 6

NÓ	<i>MPCY</i>	CICLO SIMPLES/DUPLO
1	88	Simple
2	80	Simple
3	48 (*)	Simple
4	48 (*)	Simple

(*) = O otimizador não adota o ciclo duplo neste exemplo, pois precisa diminuir o ciclo desses dois nós para 44, o que acarretaria uma saturação maior de 90% no *link* mais saturado desses dois nós.

Exemplo 2:

Quadro 7

NÓ	<i>MPCY</i>	CICLO SIMPLES/DUPLO
1	88	Simples
2	80	Simples
3	40 (**)	Ciclo duplo de 44 seg.
4	40 (**)	Ciclo duplo de 44 seg.

(**) = Neste exemplo, o otimizador irá adotar ciclo duplo de 44 s (pois $44 > 40$).

Engº SUN HSIEN MING
(GSC/CTAs 2 e 5)