

DESENVOLVIMENTO DE UM ALGORITMO DE CONTROLE EM TEMPO REAL DE UM SEMÁFORO DE DOIS ESTÁGIOS APLICANDO A REPROGRAMAÇÃO PELO MÉTODO OBSERVAÇÃO DE CAMPO

Tiago Ernani Postal Spassin, tiagoeps@yahoo.com.br
URI – Campus Erechim, Rua Dr. Sidney Guerra, 602

RESUMO: Este artigo descreve como foi desenvolvido e simulado um algoritmo de controle em tempo real de semáforos que calcula os tempos com base nos princípios da reprogramação de semáforos pelo método de observação de campo.

PALAVRAS-CHAVE: Algoritmo, Semáforo, SCOOT

ABSTRACT: This article describe show well designed and simulated an algorithm for real-time control of traffic light switch calculates the time based on the principles of reprogramming of traffic lights by the method of field observation.

KEYWORDS: Algorithm, Semaphore, SCOOT

INTRODUÇÃO

A metodologia denominada Reprogramação de Semáforos pelo Método de Observação de Campo foi exposta pelo Eng. Sergio Ejzenberg, enquanto engenheiro da CET-SP (Companhia de Engenharia de Tráfego – São Paulo), através da nota técnica N° 174/94 em maio de 1994, e visa determinar de forma simples os tempos ideais de operação de um semáforo. Tal simplicidade torna interessante sua utilização em um algoritmo para controle em tempo real de uma intersecção.

METODOLOGIA

Um algoritmo é uma instrução aplicada a um sistema de processamento de dados que, basicamente, define como o sistema tratará o fluxo de dados de entrada para responder com um certo fluxo de dados de saída que representem a melhor aproximação possível de um objetivo definido, portanto, para se desenvolver um algoritmo primeiramente deve-se saber qual é o seu objetivo principal, quais os possíveis dados de entrada e quais seriam os dados de saída admissíveis, tais informações poderiam ser chamadas de condições de contorno do algoritmo. O algoritmo desenvolvido neste trabalho tem as seguintes condições de contorno:

- O objetivo principal deve ser a minimização dos atrasos causados à uma intersecção semaforizada.
- Os dados de entrada disponíveis são os fluxos de veículos em cada uma das aproximações do semáforo.
- Os dados de saída que o algoritmo deve fornecer são os tempos de ciclo e de verdes que devem ser aplicados para minimizar os atrasos.

A linguagem usada foi C++ por ser bem desenvolvida, relativamente simples e a que o autor tem maior conhecimento. O compilador usado foi o DEV C++ Versão 4.9.9.2, ser uma versão recente e de código livre.

A metodologia de reprogramação de semáforos pelo método de observação de campo foi considerada adequada por ter a finalidade de calcular os tempos ideais de operação de um semáforo, que é o objetivo principal do

algoritmo, mas, ainda sendo o mais simples possível.

Para que se consiga calcular os tempos ideais de controle do semáforo pelo método escolhido é preciso se chegar a três dados de entrada, o tempo de verde não saturado, número de veículos que cruzam a intersecção durante o tempo de verde não saturado e o tamanho máximo da fila. Como um simples sistema de detecção não pode gerar todos os dados necessários, disponibilizando apenas o fluxo de veículos, foi necessário implementar um método de simulação dentro do algoritmo para que assim, o dado de entrada disponível fosse transformado nos dados de entrada que a metodologia de cálculo necessita. Portanto, o algoritmo pode ser dividido em duas partes distintas, uma que faz a simulação da progressão e gera os dados das condições do fluxo, e outra que, analisando as condições de fluxo, calcula os tempos ideais que devem ser aplicados ao controle da intersecção.

Simulação

O algoritmo faz com que cada veículo avance na fila uma posição (equivalente a 1 metro) a cada passo (equivalente a um décimo de segundo) até que seja impedido pelo veículo da frente ou por uma ordem de parada (sinal amarelo ou vermelho do semáforo). Cada vez que um veículo é parado é aumentado o marcador de paradas, e a cada intervalo de passo que um veículo permanecer parado é aumentado o marcador de atrasos. Caso a fila atinja seu tamanho máximo, isso impossibilitará que novos veículos detectados possam ingressar nas filas e caso isto ocorra, a cada passo que um veículo não puder entrar na fila o marcador de congestionamento será acrescido.

Calculo dos Tempos Ideais

Para poder fazer os cálculos propostos pela metodologia, o algoritmo precisa determinar se as vias estão ociosas ou congestionadas, medindo o tamanho da fila da via ao final do seu tempo de bloqueio e comparando com o tamanho de fila normal, que é uma função do fluxo de saturação e do atual tempo de verde. Resumidamente esta comparação determina se a fila máxima acumulada é

maior que a fila normal e, portanto, sobrar a fila para o próximo ciclo, ou se ela será desmanchada dentro do atual tempo de verde, sem que sobre fila para o próximo ciclo, podendo ainda existir um tempo sobrando (tempo de verde ocioso) ao final do tempo de verde. Os cálculos dos tamanhos das filas e dos tempos de verde ociosos são feitos em todos os ciclos e a cada 10 ciclos, com esta determinação, o algoritmo pode tomar dois caminhos.

Caso a fila seja maior do que a fila normal e, portanto, a via está congestionada, ele calculará qual seria o tempo de verde adicional na hora para que a fila congestionada seja desmontada, atribuindo para os próximos ciclos, um novo tempo mínimo de verde na hora formado pela soma do atual tempo com o tempo de verde adicional.

Caso a fila não seja maior que a fila normal, logo ela se desmancha antes do fim do tempo de verde atual, o algoritmo verifica se existe tempo de verde subutilizado. Nesse caso ele calcula quanto do tempo de verde está sendo subutilizado, quanto seria o suficiente e quanto é ocioso, depois, calcula um novo tempo de verde, formado pelo tempo atual deduzido o tempo de verde ocioso.

Posteriormente, com os tempos de verde mínimo de todas as vias, ele determina qual o tempo de verde mínimo de cada estágio como sendo o maior dentre os tempos de verde mínimo das vias que compõem o estágio e ainda a soma destes como sendo o tempo de verde mínimo da intersecção.

Com os tempos de verde mínimos o algoritmo calcula o grau de saturação da intersecção, depois o tempo de entreverdes máximo, o tempo de ciclo mínimo e o tempo de ciclo ótimo. Depois, com a proporção de verde necessário a cada estágio, e analisando as restrições, ele calcula os novos tempos de verde de cada estágio e o novo tempo de ciclo.

Variáveis De Controle

As variáveis de controle definidas são: $TV1$ e $TV2$ – Tempos de Verde da fase 1 e 2, respectivamente, e TC – Tempo de Ciclo.

Restrições Impostas

Tem-se como restrições impostas ao sistema: Tempos de Ciclo Máximo e Mínimo de 120 e 30 segundos, respectivamente, e Tempos de Verde Máximo e Mínimo de 100 e 10 segundos, respectivamente. Se alguma das variáveis infringir alguma das restrições, seu valor será atribuído igual ao da restrição e as outras serão recalculadas se necessário.

Determinação da Eficiência do Algoritmo

Para mensurar a eficiência do algoritmo foi considerado como base de comparação um sistema de tempos fixos, e foram simulados vários cenários, variando os fluxos nos estágios, dados em veículos por hora, e os tempos do sistema fixo, dados em segundos, num total de nove combinações diferentes.

RESULTADOS E DISCUSSÃO

Realizadas as nove simulações foram extraídos os dados dos marcadores, alocados em uma planilha eletrônica e

construídos gráficos para comparação da evolução alcançada pelo algoritmo frente ao sistema com tempo fixo. Com base nos gráficos de comparação, foi possível afirmar que o algoritmo tem um desempenho satisfatório e superior ao sistema de tempos fixos, acentuando-se tal superioridade em condições de desbalanceamento de demanda e principalmente em condição de fluxos mais elevados. O único cenário onde o algoritmo não conseguiu reduzir os atrasos foi onde se excedeu o fluxo máximo da intersecção, condição na qual, não existe uma programação que não cause congestionamento.

CONCLUSÃO

Com base nos resultados alcançados pode-se afirmar que o algoritmo tem um desempenho satisfatório e superior ao sistema de tempos fixos e é válido destacar que uma característica importante do algoritmo é sua adaptabilidade, de maneira que, ao substituir um sistema de tempos fixos, ele se adequaria constantemente a variação do tráfego.

Também seria extremamente interessante a continuação do desenvolvimento de tal trabalho com o intuito de melhorá-lo e ampliá-lo, principalmente no que diz respeito à coordenação de múltiplas intersecções, afim de se alcançar um desempenho global mais elevado.

Após concluir o algoritmo lhe foi dado nome de CSTRMOC – Controle Semáforo em Tempo Real pelo Método de Observação de Campo.

REFERÊNCIAS

- DENATRAN, 1985, “Manual de Semáforos”, 2ª ed., Coleção Serviços de Engenharia, Ministério da Justiça, Brasília – DF.
- EJZENBERG, Sergio, 1994, “Reprogramação de Semáforos: Método Baseado em Observação de Campo”, Nota Técnica 174/94 da CET - Companhia de Engenharia de Tráfego / SP, São Paulo – SP.
- EJZENBERG, Sergio, 2005 “Reprogramação de Semáforos: Método Baseado em Observação de Campo” Artigo do site www.sinaldetransito.com.br, disponível em www.sinaldetransito.com.br/artigos/metodo_de_cam po.pdf, acessado em 29/10/2011.

DECLARAÇÃO DE RESPONSABILIDADE

O autor é o único responsável pelo material impresso contido neste artigo.