

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
CURSO DE SISTEMAS DE INFORMAÇÃO

**CONTROLADORES VIRTUAIS EM AMBIENTE DE MICRO-
SIMULAÇÃO DE TRÁFEGO URBANO**

Michel Pires de Araujo

Trabalho de conclusão de curso
apresentado como parte dos requisitos
para obtenção do grau de Bacharel em
Sistemas de Informação

Florianópolis - SC

2005 / 1
Michel Pires de Araujo

**CONTROLADORES VIRTUAIS EM AMBIENTE DE MICRO-
SIMULAÇÃO DE TRÁFEGO URBANO**

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Sistemas de Informação

Orientador: Dr. João Bosco Manguiera Sobral

Co-orientador: Dr. Werner Kraus Junior

Banca examinadora

Dr. João Bosco Manguiera Sobral

Dr. Werner Kraus Junior

Dr. Bernardo Gonçalves Riso

DEDICATÓRIA

Aos meus pais, Maura Ana Pires de Araujo e Ronaldo Marques de Araujo pela educação que agora completa mais uma etapa;

Aos meus irmãos, Marcelo Pires de Araujo, Michelle Pires de Araujo e Ronaldo Marques de Araujo Júnior, sempre muito pacientes comigo;

A Ana Cristina de Oliveira Kerber por estar ao meu lado em todos os momentos difíceis ao longo deste caminho;

Aos meus eternos amigos: Eduardo Martins Gonçalves, Jason Schreiner dos Santos e Mariana Bittencourt.

AGRADECIMENTOS

A todos meus colegas do projeto SincMobil que participaram do desenvolvimento deste sistema no qual meu trabalho fez parte;

Ao Ricardo Schmidt e Carlos Eduardo Gesser pela ajuda nos momentos de desespero;

Ao meu cunhado Fábio Barbosa Almeida pelas correções, sem o qual esta monografia não teria a mesma qualidade;

E, por fim, ao Flavio Henrique Cuareli, pelas ricas reflexões conjuntas sobre o tema, revisão geral e auxílio na modelagem e implantação do sistema alvo deste trabalho.

Sumário

| | |
|---|-----------|
| Lista de Figuras | 6 |
| Resumo | 7 |
| 1. Introdução | 8 |
| 1.1 <i>Visão Geral</i> | 8 |
| 1.2 <i>Organização</i> | 10 |
| 2. Fundamentação Teórica | 12 |
| 2.1 <i>Objetivos do Controle Semafórico</i> | 12 |
| 2.2.1 <i>Controle Semafórico de Tempo Fixo</i> | 15 |
| 2.2.2 <i>Controle Semafórico em Tempo Real</i> | 17 |
| 2.2 <i>Micro simulação</i> | 20 |
| 2.3 <i>Instrumentação Virtual</i> | 21 |
| 2.4 <i>Sistemas em Tempo Real</i> | 23 |
| 2.5 <i>Linguagem XML</i> | 25 |
| 3. Descrição do Sistema..... | 27 |
| 3.1 <i>Simulador Sitra B+</i> | 30 |
| 3.2 <i>Sincronizador</i> | 32 |
| 3.3 <i>Central de Controle de Tráfego</i> | 34 |
| 3.4 <i>Controlador Virtual</i> | 37 |
| 3.4.1 <i>Desenvolvimento do Controlador</i> | 38 |
| 3.4.2 <i>Estrutura do controlador</i> | 40 |
| 3.4.3 <i>Controlador Virtual - Tempo Fixo</i> | 47 |
| 3.4.4 <i>Controlador Virtual - Tempo Real</i> | 50 |
| 3.4.5 <i>Controlador Virtual - Modo Passivo</i> | 53 |
| 4. Resultados | 55 |
| 5. Conclusões e Perspectivas..... | 57 |
| 6. Referências Bibliográficas | 58 |
| 7. Anexos..... | 61 |
| 7.1 <i>Interfaces do Controlador Virtual</i> | 61 |
| 7.2 <i>Mensagens XML</i> | 64 |
| 7.3 <i>Artigo</i> | 68 |
| 7.4 <i>Código-fonte</i> | 72 |

Lista de Figuras

| | |
|--|----|
| FIGURA 3-1 TROCA DE INFORMAÇÕES DENTRO DO SISTEMA. | 29 |
| FIGURA 3-2 SIMULAÇÃO DO <i>SOFTWARE</i> SITRA B+. | 30 |
| FIGURA 3-3 INTERFACE DO SINCRONIZADOR V1.4. | 32 |
| FIGURA 3-4 INTERFACE DA CENTRAL DE CONTROLE. | 36 |
| FIGURA 3-5 DIAGRAMA DE CLASSES DO CONTROLADOR. | 41 |
| FIGURA 3-6 FLUXOGRAMA DO MODO EM TEMPO FIXO | 48 |
| FIGURA 3-7 FUNCIONAMENTO DA PRIMEIRA VERSÃO DO CONTROLADOR. | 49 |
| FIGURA 3-8 FUNCIONAMENTO DO CONTROLE EM TEMPO REAL. | 51 |
| FIGURA 3-9 FUNCIONAMENTO DA SEGUNDA VERSÃO DO CONTROLADOR. | 52 |
| FIGURA 3-10 FUNCIONAMENTO DA TERCEIRA VERSÃO DO CONTROLADOR. | 54 |
| FIGURA 7-1 TELA INICIAL DE CONFIGURAÇÃO | 61 |
| FIGURA 7-2 TROCA DE MENSAGENS ENTRE ENTIDADES | 61 |
| FIGURA 7-3 CONTROLE EM TEMPO FIXO | 62 |
| FIGURA 7-4 CONTROLE EM TEMPO REAL | 62 |
| FIGURA 7-5 CONTROLE EM MODO PASSIVO. | 63 |
| FIGURA 7-6 XML DE CONFIGURAÇÃO. | 64 |
| FIGURA 7-7 XML DE SINCRONISMO. | 64 |
| FIGURA 7-8 XML DE CONFIGURAÇÃO ESTÁGIOS. | 64 |
| FIGURA 7-9 XML DE CONFIGURAÇÃO DE PLANOS. | 65 |
| FIGURA 7-10 XML DE ATIVAÇÃO. | 65 |
| FIGURA 7-11 XML TIPO DE OPERAÇÃO. | 65 |
| FIGURA 7-12 XML CONFIGURAÇÃO TEMPO REAL. | 66 |

Resumo

CONTROLADORES VIRTUAIS EM AMBIENTE DE MICRO-SIMULAÇÃO DE TRÁFEGO URBANO

Apresenta-se neste trabalho a implementação de um instrumento virtual para ambiente de simulação do controle de tráfego veicular urbano. Os principais elementos desenvolvidos são um controlador semafórico que se comunica com uma central de controle de tráfego e um detector veicular associado a este controlador.

O estudo visa facilitar o desenvolvimento de estratégias de controle de tráfego urbano através da implementação de plataforma de testes de algoritmos de controle. Tais testes abrangem a comunicação de dados e o desempenho do controle. Um programa de micro-simulação de tráfego é usado para gerar cenários urbanos necessários para estudo das estratégias.

Neste trabalho foi realizado um estudo de como implementar um controlador semafórico virtual com a capacidade de atuar sobre uma simulação. Além da análise, foi realizado o desenvolvimento desta instrumentação virtual que auxiliará engenheiros de tráfego a construir algoritmos de controle em tempo real com desempenho maximizado.

Palavras-chave

Controlador semafórico, instrumentação virtual, tráfego urbano, micro-simulação.

1. Introdução

1.1 Visão Geral

O trânsito nos grandes centros urbanos causa muitos problemas para seus habitantes. Ele atinge desde a economia, com o desperdício de recursos, até a saúde da população, sendo esse efeito sobre a saúde não apenas ligado à poluição gerada pelos veículos, mas também ao estresse relacionado aos congestionamentos e à lentidão do trânsito.

Nas cidades grandes, metade do tempo que se gasta em deslocamentos utilizando veículos automotores ocorre em filas nos cruzamentos. Esse tempo de espera representa um acréscimo de cerca de 30% no consumo de combustível (DENATRAN, 1979), o que é um desperdício que se reflete na economia local, uma vez que nos preços de todos os produtos está embutido o custo do frete. Este atraso nos deslocamentos afeta também a emissão de poluentes pelos veículos: quanto mais tempo os veículos trafegam mais poluentes eles emitem.

As pessoas que enfrentam congestionamentos, o ápice dos atrasos veiculares, estão sujeitas ao calor, ao barulho, à irritação, à sensação de perda de tempo, etc. Estes fatores provocam estresse em quem trafega por regiões onde existem estes problemas. Este estresse agrava ainda mais o problema do congestionamento, uma vez que alguns motoristas tentam encontrar atalhos onde esses não existem. Além disso, esse estresse gerado no trânsito atrapalha a vida

dos motoristas mesmo quando estes não estão dirigindo, prejudicando seu relacionamento pessoal, bem como seu rendimento profissional.

Do que foi exposto pode-se perceber que os atrasos veiculares são um grande problema para a Engenharia de Tráfego, e que uma redução desses atrasos pode trazer grandes melhorias para a fluidez do trânsito.

Uma forma de reduzir os atrasos é otimizar o controle dos cruzamentos semaforizados, que são tradicionalmente controlados por tabelas de tempo fixo. Algoritmos de tempo real, por exemplo, são mais eficientes que as tabelas de tempo fixo por se adaptarem às variações repentinas da demanda, o que as tabelas não são capazes de fazer.

Ao se desenvolver um sistema de controle de tráfego em tempo real é necessário realizar testes para analisar seu desempenho e funcionamento. Como é inviável realizar esses testes nas ruas, utiliza-se um simulador de trânsito, de onde surge a necessidade de construir uma entidade virtual capaz de emular parte dos dispositivos físicos presentes em um sistema de controle de tráfego.

O objetivo do trabalho descrito neste relatório é implementar um controlador de tráfego virtual para auxiliar no desenvolvimento de sistemas de controle de semafórico em tempo real.

1.2 Organização

A seguir, será apresentada a estrutura deste documento:

- o presente capítulo trata-se da Introdução onde é estabelecido o primeiro contato com o leitor sobre o tema abordado;
- o capítulo 2 é um apanhado das diversas teorias utilizadas para o desenvolvimento deste trabalhos. Tem como objetivo, portanto, realizar uma fundamentação teórica do tema proposto;
- o capítulo 3 trata da descrição do Sistema. Neste capítulo é demonstrado o sistema como um todo e como foi realizada a construção do controlador semafórico virtual;
- o capítulo 4 traz os resultados obtidos com a implantação deste instrumento virtual no sistema;
- o capítulo 5 demonstra as conclusões e perspectivas obtidas com a construção deste trabalho. Neste capítulo são demonstradas as potencialidades e limitações do sistema proposto, bem como seu provável auxílio para estudos futuros;

- o capítulo 6 apresenta as referências bibliográficas utilizadas para o desenvolvimento deste trabalho.

2. Fundamentação Teórica

2.1 Objetivos do Controle Semafórico

De acordo com GAZIS (1930), existem dois objetivos principais a serem alcançados com o controle de tráfego urbano: o primeiro diz respeito à segurança dos transientes e o segundo sobre a liberdade de escolha do motorista. A segurança é um objetivo óbvio, consiste em separar os fluxos veiculares para evitar colisões e atropelamentos. Liberdade de escolha é a opção que o motorista deve ter em escolher o caminho que ele julgue ser mais rápido e econômico para alcançar um destino qualquer.

Modernamente, pode-se atribuir também ao controle de tráfego o objetivo de otimizar os recursos viários dividindo o tempo de verde entre os fluxos conflitantes, buscando a redução máxima do atraso médio dos veículos que trafegam pela malha viária controlada. Muitas vezes, porém a redução dos atrasos entra em conflito com a segurança dos usuários. Nestes casos, a prioridade sempre será a segurança, mas deverá ser alcançado um meio de se resolver este problema de tal forma que se agrave o mínimo possível o atraso.

O controle de tráfego em cruzamentos semaforizados é realizado por controladores semafóricos. Os controladores mais utilizados atualmente são aqueles baseados em tabelas de tempo fixo, isto é, que utilizam tempos de verde e vermelho fixos. Estes tempos são dimensionados e armazenados, podendo variar conforme um cronograma de horário, não possuindo flexibilidade acima disso. Estes controladores que utilizam tabela de tempo fixo são chamados de “controladores de tempo fixo”.

Com o avanço no estudo de automação, controle e informática, um novo conceito de controle semafórico foi desenvolvido. Estes novos controladores semafóricos têm a capacidade de captar dados dos fluxos que passam dentro de seu domínio e tomar a decisão, em tempo real, do melhor tempo de verde para cada via, no sentido de ajudar o escoamento mais rápido dos veículos que ali trafegam. Estes controladores são conhecidos como “controladores semafóricos em tempo real”.

O termo “controlador semafórico inteligente” poderá ser utilizado para os controladores que apresentam algum tipo de aprendizado durante sua execução. Um sistema que é inteligente não pode ter sua decisão prevista em uma situação futura, pois sua percepção muda com o tempo. WINSTON (1992) define a Inteligência Artificial como sendo o estudo da computação que torna possível perceber, raciocinar e agir.

O presente trabalho tratará apenas dos controladores semafóricos de tempo fixo e controladores semafóricos em tempo real.

2.2.1 Controle Semafórico de Tempo Fixo

Os controladores semafóricos de tempo fixo possuem uma tabela de tempos previamente estabelecida que não varia de acordo com a demanda de veículos. O tempo dos estágios pode variar no máximo de acordo com uma configuração baseado em um horário, ou seja, pode haver diferentes planos pré-estabelecidos para entrarem em funcionamento em determinados horários do dia.

Os controladores semafóricos de tempo fixo não levam em conta as informações obtidas de detectores veiculares sobre a situação do trânsito local. Este tipo de controle pode ser classificado como malha aberta por não utilizar informações sobre o estado corrente do sistema controlado.

Existem várias formas para calcular o melhor tempo a ser aplicado por um controlador semafórico de tempo fixo. De acordo com BRITO (1998), temporizar um sistema de semáforo que opera com ciclo fixo significa determinar basicamente: o ciclo que fornece o mínimo atraso e a divisão deste ciclo entre diversas fases (*split*).

Uma forma de calcular o *split* é dividir o valor do fluxo observado pelo fluxo de saturação de cada faixa. Posteriormente, selecionar o maior valor para cada faixa e distribuir o tempo de verde efetivo entre os grupos de movimentos que serão separados na proporção dos valores encontrados. De acordo com Webster

apud BRITO (1998), o *split* ótimo difere muito pouco do encontrado por este método, não sendo necessário maior esforço em utilizar uma técnica mais elaborada.

Em interseções onde o fluxo de veículos se altera muito durante certos horários, torna-se interessante a implementação de outros planos. Estes teriam valores de *split* diferenciados, que poderão ser colocados em funcionamento em determinados horários pré-programados durante o dia.

2.2.2 Controle Semafórico em Tempo Real

Os controladores semafóricos em tempo real executam algoritmos de controle realimentado, utilizando detectores veiculares para verificar o estado corrente do tráfego.

Os sensores extraem informações sobre a situação do tráfego e as encaminham ao controlador semafórico, este processa as informações e toma a decisão de alterar ou não o estado dos grupos semafóricos. Os veículos obedecem ao sinal e a situação volta a ser detectada pelos sensores. Outra entidade capaz de realizar a tarefa de processar as informações extraídas pelos sensores é a central de controle de tráfego.

Justificando o uso do controle em tempo real, SOUZA et al (2004) sugere:

[...] apesar de terem custo maior de implantação, são preferíveis, pois conseguem ajustar-se automaticamente às variações de fluxo e eliminam o custo de atualização dos planos de tempos fixos. Também, apresentam melhor desempenho, pois podem aproveitar-se de variações de demanda ciclo-a-ciclo, algo que não pode ser feito pelas estratégias de cálculo off-line de tempos semafóricos. Estes sistemas de controle capazes de reagir às variações de demanda de tráfego são objeto de grande interesse comercial e acadêmico.[...]

O processamento de controle em tempo real é realizado através de algoritmos de controle. A metodologia de construção destes algoritmos não será abordada neste trabalho.

Ao contrário dos controladores semafóricos de tempo fixo, os controladores de tempo real se adaptam a condições inesperadas. Um controlador de tempo fixo, quando trabalha em uma condição fora da qual foi programada, acaba tendo um desempenho muito baixo, enquanto os de tempo real tentam minimizar o atraso médio independentemente da situação a que estão sujeitos.

De acordo com BRITO (1998) existem três formas de realizar o controle de malha fechada, cada uma específica para um caso diferente:

-Controle semi-realimentado – É apropriado para cruzamentos de vias de baixo tráfego com vias de fluxo intenso. A principal característica deste controle é a existência de detectores veiculares apenas nas vias secundarias (de menor fluxo). Desta forma, o controle pode fornecer verde constante a via de maior fluxo e apenas mudar seu estágio caso algum veículo seja detectado nas vias secundarias.

- Controle completamente realimentado – É indicado para cruzamento de vias que possuem fluxos semelhantes de veículos. Seu trabalho é semelhante ao controle semi-realimentado, com a diferença que não existe uma via principal, todas são caracterizadas como secundárias. Os sensores estão presentes em todas as vias.

- Controle com característica de volume-densidade – Avalia de forma mais elaborada as condições dos fluxos existentes em seu domínio. Tem a capacidade de estimar tamanho das filas e alterar parâmetros do controlador.

2.2 Micro simulação

Uma simulação é uma abstração da realidade onde se busca imitar determinado sistema em funcionamento. Villamil (2003) sugere dois tipos de abordagens para classificar uma simulação: microscópica e macroscópica.

Na abordagem macroscópica, são utilizadas informações sobre o comportamento de um grupo de elementos para se chegar em um sistema simulado. Na abordagem microscópica, são utilizados os comportamentos individuais de cada envolvido da simulação para se gerar o comportamento do grupo.

Muitas vezes, a representação de um sistema através da abordagem macroscópica é suficiente para que uma simulação alcance seu objetivo. Existem casos onde há a necessidade de se representar uma simulação de forma que cada elemento participante seja descrito individualmente (Claramunt & Jiang 2001).

Para criar uma simulação de tráfego, optou-se por utilizar uma micro-simulação, onde o comportamento de cada veículo é tratado individualmente. Para realizar a micro-simulação de tráfego urbano, foi utilizado o software Sitra B+. Esta ferramenta proporciona liberdade de configurar diversos detalhes sobre cada veículo envolvido na simulação, na tentativa de alcançar o máximo de desempenho.

2.3 Instrumentação Virtual

Buscando o significado da palavra “instrumento” em um dicionário da língua portuguesa (Michaelis, 1998), encontramos a seguinte definição: “Aparelho, objeto ou utensílio que serve para executar uma obra ou levar a efeito uma operação mecânica em qualquer arte, ciência ou ofício”. No mesmo dicionário encontra-se, para palavra virtual, o seguinte significado: “Que não existe como realidade, mas sim como potência ou faculdade”.

Um instrumento virtual seria, portanto, um programa que simule um objeto real, tendo como finalidade uma aplicação específica. De acordo com JOHNSON (1997), “A instrumentação virtual tem como objetivo usar um computador comum para criar um aplicativo simulando instrumentos reais com controles e ambiente personalizados, mas com a versatilidade que acompanha o programa de aquisição”.

O presente relatório descreve o desenvolvimento de um simulador de controlador semafórico a ser utilizado em conjunto com simulações de trânsito. O controlador virtual tem a finalidade de controlar o fluxo de veículos que passam sob determinado cruzamento da simulação, assim como o controlador real tem a finalidade de controlar o fluxo de veículos que passam sob determinado cruzamento da malha viária real.

Uma das principais vantagens do uso de controladores semafóricos virtuais em conjunto com simulações de trânsito é a não realização de testes para medida de desempenho do sistema de controle na malha viária real, o que implicaria em transtornos e riscos para a população. Outra vantagem proporcionada é o baixo custo para realizar estes testes.

Além disso, novas funcionalidades podem ser implementadas rapidamente, sem a necessidade de ir a campo ou modificar partes físicas dos equipamentos, bastando apenas fazer alterações no código fonte dos programas locais.

2.4 Sistemas em Tempo Real

No decorrer do progresso tecnológico de tratamento da informação, um requisito que se tornou indispensável para certas aplicações foi o conceito de tempo real. Um sistema é dito em tempo real quando existe a necessidade de realizar determinado processamento em um tempo específico. Isto não significa que o processamento deva ser extremamente rápido, mas sim que ele deve terminar em determinado intervalo de tempo. MARQUES (1990) descreve: “Os sistemas em que a noção de tempo é relevante são designados sistemas em tempo real e têm como principal objetivo garantir que a resposta a um estímulo externo seja dada ao fim de um intervalo de tempo especificado”.

Existem dois tipos de sistemas em tempo real. São eles:

- *soft real-time* – Sistemas em tempo real que são tolerantes a falhas no requisito de tempo. Isto é, caso a aplicação não consiga cumprir os requisitos de tempo real, o sistema continuará funcionando sem grandes perdas. É evidente que essas falhas, apesar de serem toleráveis, não são desejadas.

- *hard real-time* – São conhecidos como sistemas de tempo real crítico e não aceitam nenhum tipo de falha no requisito de tempo. Quando uma falha desse tipo acontece, suas conseqüências são catastróficas, comprometendo o funcionamento do sistema como um todo.

O controlador virtual é um sistema em tempo real, uma vez que as trocas de mensagens entre os processos possuem restrições de tempo. Ele não é um sistema *hard-real time* porque, ainda que não se tratasse de uma simulação, as condições das vias não se alteram de forma brusca em um curto intervalo de tempo, de modo que caso alguma mensagem se perca ou sofra um atraso, o funcionamento do sistema não será comprometido. Trataremos, então, apenas dos sistemas *soft real-time*.

As aplicações de *soft real-time* devem, portanto, fornecer a possibilidade de se perder um determinado número de prazos e permitir que este sistema seja operacionalmente aceitável. De acordo com OLIVEIRA (1998):

Quando os requisitos temporais não são críticos (soft real-time) é possível que determinadas tarefas não consigam executar sequer suas respectivas partes obrigatórias. Partes obrigatória e opcional passam a ser, na verdade, partes com precisão mínima e máxima, respectivamente. A motivação para usar Computação Imprecisa em sistemas deste tipo reside no fato dela permitir que a aplicação adapte a qualidade de seus resultados dinamicamente, conforme a disponibilidade dos recursos computacionais.

Caso o controlador não se tratasse de uma aplicação *soft real-time*, não haveria a possibilidade de se obter um bom resultado caso houvesse, por exemplo, uma falha de comunicação por poucos instantes ou uma mensagem corrompida.

2.5 Linguagem XML

XML é uma linguagem de marcação que tem por objetivo a integração entre dados e usuários. Tudo começou com o HTML que também é uma linguagem de marcação. O HTML (*Hypertext Markup Language*) foi desenvolvido para publicação de documentos em meios eletrônicos, e com passar do tempo, tornou-se o padrão da Internet. Com o progresso da Internet, suas funcionalidades foram crescendo, deixando de ser apenas um meio de publicação de documentos, para se tornar um meio de acesso a aplicações, banco de dados, etc. O HTML passou a ser uma limitação, pois não dava suporte a estas novas funcionalidades da Internet. Uma forma de suprir estas necessidades foi o desenvolvimento da *Extended Markup Language* (XML).

HTML e XML são ditas linguagem de marcação por apresentarem elementos em seu corpo que indicam a maneira como determinada parte deve ser representada. Esses elementos são chamados de “marcação” (*tags*). ALMEIDA (2002) define linguagem de marcação como sendo “um conjunto de convenções utilizadas para a codificação de textos. Uma linguagem de marcação deve especificar que marcas são permitidas, quais são exigidas, como se deve fazer distinção entre as marcas e o texto e qual o significado da marcação”.

A linguagem HTML tem um determinado número de elementos com objetivo bem especificado. Para utilizar o HTML, ter-se-á de ficar limitado as “*tags*” existentes. Já o XML permite ao autor do documento definir suas próprias

marcações, possibilitando melhorias significativas em processos de disseminação da informação. Diz-se que XML não se trata de uma linguagem de marcação pré-definida pois, ao contrario do HTML, não apresentam elementos definidos. O autor de determinado documento terá a liberdade de criar suas próprias *tags*. Além de conter dados, um arquivo XML pode conter os chamados DTDs (*Data Type Definitions*) que se trata de uma gramática que confere se o próprio arquivo encontra-se no seu formato esperado.

Dentre as principais características do XML, podemos destacar: Pode ser diretamente utilizada na Internet; é legível ao homem, pois ele mesmo quem cria suas próprias marcações; possibilita meio independente de publicação eletrônica; permite definir protocolos para troca de dados independente de plataforma lógica ou física.

Por facilitar a troca de mensagens entre sistemas heterogêneos e permitir rápidas alterações em sua estrutura, optou-se por utilizar o XML neste trabalho. O XML será utilizado na troca de mensagens entre as entidades que compõem o sistema de controle de tráfego.

3. Descrição do Sistema

O sistema de controle de tráfego em tempo real proposto é dividido em três partes. Uma delas é um simulador de tráfego, o *software* Sitra B+. É através da simulação realizada por ele que será mensurado o resultado da utilização do controlador virtual. O Sitra B+ se comunica com outros aplicativos através de arquivos de texto gravados em disco. Ele faz a leitura do estado em que deverão estar os grupos semaforicos no próximo passo da simulação e escreve nos arquivos o resultado dessa escolha.

Outra parte deste sistema é o controlador semaforico virtual, um instrumento virtual que emula um controlador de grupos semaforicos existente em interseções de vias reais. O controlador virtual opera, neste sistema, através da leitura dos dados das condições das vias fornecidos pela simulação. A leitura dos arquivos do simulador é realizada por outro *software*, chamado Sincronizador que, posteriormente, passa os dados extraídos para o controlador. Após receber estes dados, o controlador executa o algoritmo de controle, e retorna à simulação qual deve ser o estado dos semaforos no próximo passo.

Por último, temos a Central de Controle, responsável por gerenciar macroscopicamente a execução dos controladores semaforicos presentes na simulação. A Central de Controle é um programa que recebe informações de

todos os elementos do sistema e as repassa, quando necessário, para os controladores, como um auxílio para a tomada de decisão destes.

Além dos três módulos apresentados, o sistema possui outras partes que dão suporte ao seu funcionamento. Estas partes são: o *software* Sincronizador, que fornece a comunicação entre os controladores virtuais e a simulação e, o Sistema de Informação de Tráfego que disponibiliza, através da Internet, informações sobre as condições das vias. Cada parte do sistema será abordada detalhadamente adiante.

A figura 3-1 apresenta a troca de informações entre os diversos módulos que compõe o sistema.

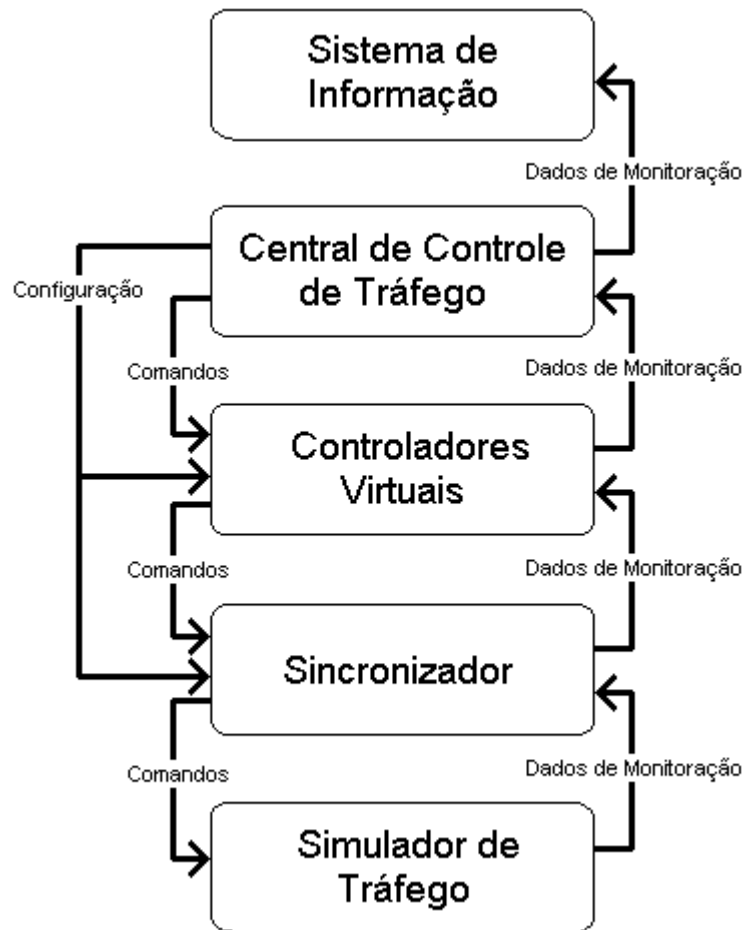


Figura 3-1 Troca de informações dentro do sistema.

3.1 Simulador Sitra B+

O Sitra B+ é um programa desenvolvido na França com o objetivo de simular o tráfego de veículos em vias pré-estabelecidas. Os veículos que participam da simulação podem ser de três tipos: carros, motos e ônibus. Fornece, portanto, um grau de realidade bem satisfatório para o propósito deste trabalho.

Durante a utilização do Sitra B+ é possível ver o estado da simulação através de uma interface gráfica como mostra a figura 3-2.

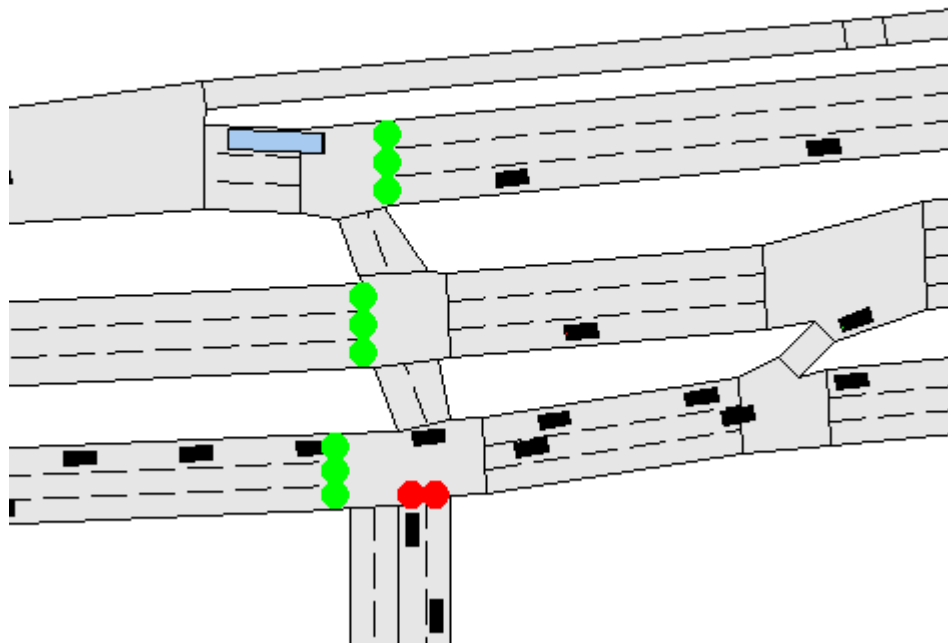


Figura 3-2 Simulação do *software* Sitra B+.

É possível configurar o Sitra B+ de modo a simular qualquer malha viária com controle interno ou externo, isto é, os semáforos da simulação podem ser controlados pelo próprio simulador ou por um algoritmo externo.

Para fazer o controle externo em tempo real é necessário que a simulação possua sensores veiculares, da mesma forma que são instalados laços indutivos nas vias reais. Desta forma, o simulador fornece, como parâmetros de saída, dados sobre a condição das vias, e é com base nesses dados que o controlador virtual toma as decisões para atuar sobre os grupos semaforicos simulados, controlando os fluxos de veículos da simulação.

O controle dos semáforos é feito através de parâmetros de entrada aceitos pelo Sitra B+. Após o controlador tomar as decisões, é especificado ao simulador como deverá estar a situação dos semáforos presentes em seu ambiente virtual. Desta forma, a situação dos grupos semaforicos que foi estabelecido externamente à simulação vai controlar os fluxos presentes internamente no Sitra B+.

O Sitra B+ comunica-se com outros *softwares* através da leitura e escrita em arquivos de texto. Para interagir com a simulação, foi utilizado um *software*, chamado Sincronizador, que abstrai esta tarefa. Para o desenvolvimento do controlador virtual, não será necessário conhecimento profundo do *software* Sitra B+. Por este motivo, detalhes do seu funcionamento não serão abordados neste trabalho.

3.2 Sincronizador

Uma vez que o simulador SITRA B+ se comunica com outros *softwares* através de arquivos de texto e uma vez que se desejava que o Controlador Virtual fosse independente do simulador, foi necessário implementar um *software* que estabelecesse a ligação entre eles. Esta ligação é o Sincronizador.

Ele é responsável por ler os arquivos de saída da simulação e repassar as informações relevantes para os Controladores Virtuais bem como por receber destes os comandos de atuação e escrever nos arquivos de entrada da simulação para que esses comandos se realizem.

A comunicação entre o Sincronizador e os Controladores Virtuais é feita através de conexões *socket*, e as mensagens respeitam o formato XML.

Na figura 3-3 está apresentada a interface do Sincronizador.

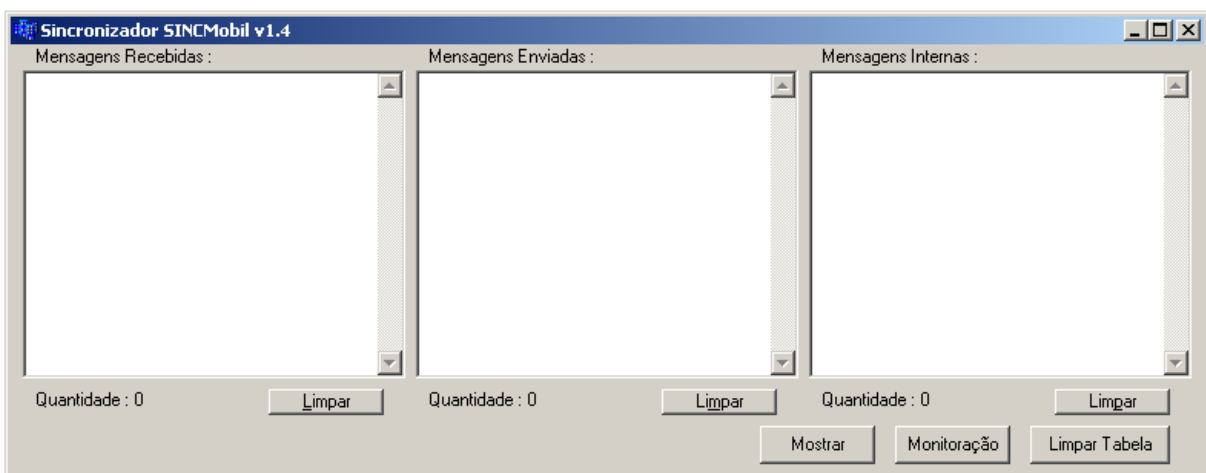


Figura 3-3 Interface do Sincronizador v1.4.

O Sincronizador foi implementado por outro pesquisador. Detalhes sobre seu funcionamento não serão abordados neste trabalho.

3.3 Central de Controle de Tráfego

Uma Central de Controle de Tráfego é um *software* utilizado para administrar o trânsito de uma determinada malha viária. Ela sempre faz parte de um sistema maior, composto ao menos por sensores e atuadores além da própria Central. Um sistema desse tipo é chamado de Sistema Inteligente de Transportes, ou ITS (da sigla em inglês). Um exemplo de padronização que define a arquitetura e as funções desse tipo de sistema é a NIA (*National ITS Architecture*), sendo usada para nos EUA para desenvolvimento dos ITS. O grau de controle que uma central pode oferecer aos operadores pode variar muito conforme sua complexidade. O fator econômico associado às necessidades do domínio a ser controlado pela central, determina o quão complexo este sistema será.

A Central de Controle utilizada neste trabalho foi desenvolvida para o sistema operacional Linux, na distribuição Mandrake 9.2. De acordo com o padrão da NIA, esta Central foi planejada para operar sobre a malha viária de uma cidade de médio porte. Sua principal função é executar algoritmos de controle em tempo real: ela recebe dados dos sensores, executa o algoritmo de controle escolhido tendo esses dados como entrada e então repassa as decisões do algoritmo para os dispositivos atuadores. No caso da simulação em laboratório, os sensores já são simulados pelo simulador SITRA B+, e os dispositivos atuadores são substituídos pelos Controladores Virtuais.

Outra função da Central de Controle é alimentar o Sistema de Informação com as informações de interesse que chegam da simulação. Este sistema disponibiliza informações sobre as condições das vias aos usuários através de uma interface web.

Como a Central de Controle opera recebendo informações de todos os Controladores a ela ligados, ela possui uma visão geral das condições de todas as vias da malha viária sob sua administração. Dessa forma, ela consegue executar algoritmos que tomam decisões de âmbito mais geral, que abrangem toda a malha viária, ao contrário dos algoritmos executados nos Controladores, que tomam decisões locais.

A comunicação entre a Central de Controle e os vários Controladores Virtuais foi implementada através conexões *socket*. As mensagens que eles trocam são todas em formato XML.

A figura 3-4 mostra a interface da Central de Controle.

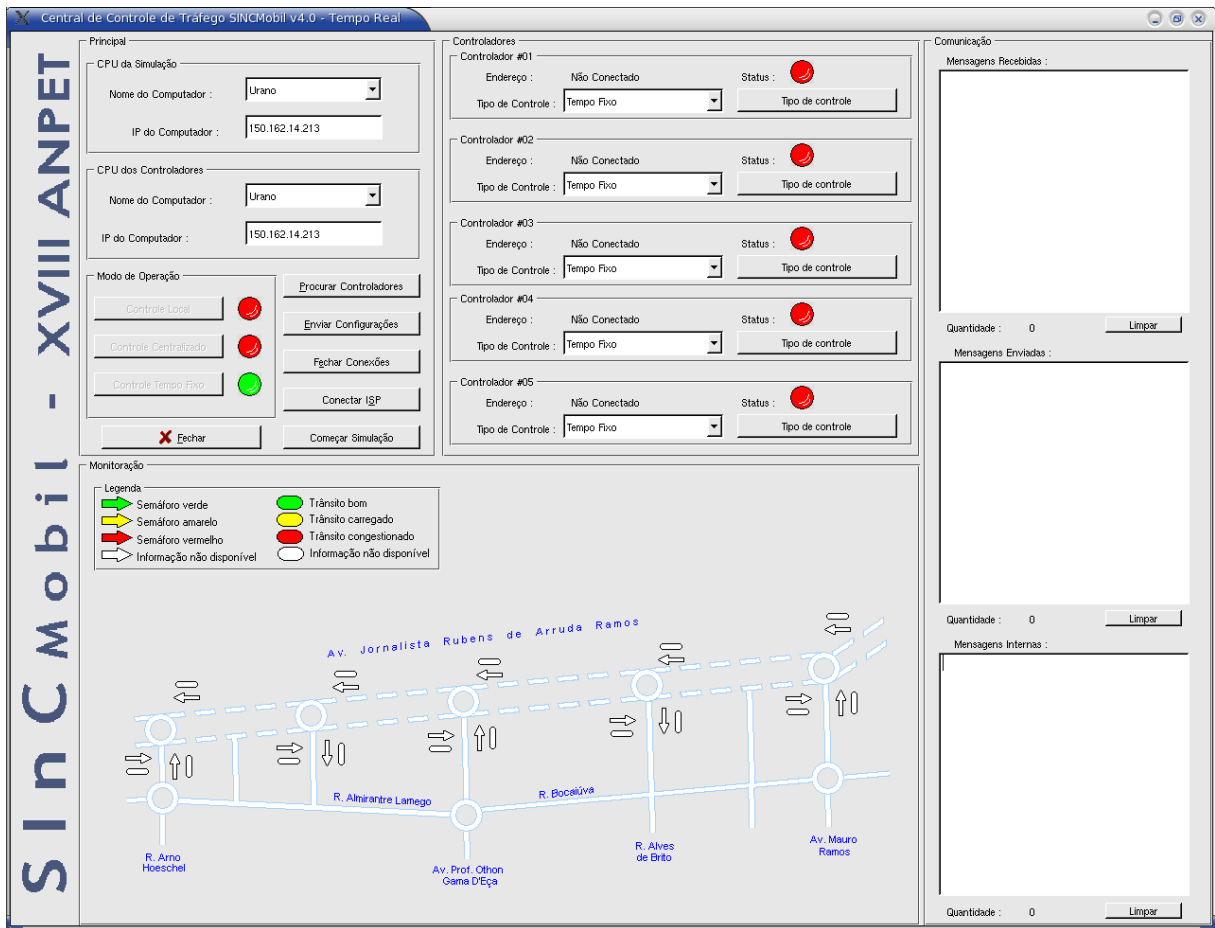


Figura 3-4 Interface da Central de Controle.

3.4 Controlador Virtual

O controlador virtual é um artefato computacional capaz de realizar tomada de decisão com base em dados de entrada e atuar sobre grupos semafóricos de simulações de trânsito. Durante o processo de desenvolvimento deste elemento foi dada atenção, primeiramente, ao controle em tempo fixo e, posteriormente, ao controle em tempo real. Apesar do foco deste trabalho ser no controle de tempo real, o desenvolvimento do controle em tempo fixo é necessário para que se possa fazer comparações e analisar o quanto o controle em tempo real proporcionou melhorias no tráfego de veículos da simulação.

O controlador desenvolvido realiza o controle tanto em tempo real quanto em tempo fixo, podendo alternar o modo de controle em tempo de execução. Esta capacidade é necessária para que, em caso de falhas de algum elemento do sistema, o controle que opera em tempo real possa ser convertido para tempo fixo para minimizar as consequências da falha. Um exemplo de falha é a perda de conexão com um detector veicular onde não haveria a possibilidade de estimar a fila de veículos em determinado trecho. Tal falha comprometeria todo o fluxo de veículos, caso o controle continuasse a trabalhar em tempo real.

3.4.1 Desenvolvimento do Controlador

De acordo com PRESSMAN (1995), a Engenharia de *Software* é composta por métodos, ferramentas e procedimentos. Neste contexto, os métodos englobariam o planejamento, estimativas, análise de requisitos, projetos da estrutura, fluxos de processos, geração de código e manutenção do sistema. Ferramentas seriam os *softwares* que auxiliam nas etapas dos métodos, conhecidas como ferramentas CASE (*Computer-Aided Software Engineering*). Os procedimentos são a forma como serão organizadas e executadas as outras etapas que compõem a Engenharia de *Software*.

Dos procedimentos existentes na Engenharia de *Software*, optou-se por utilizar neste trabalho a técnica de desenvolvimento em espiral. Esta técnica tem o objetivo de implantar, primeiramente, um protótipo executável com funcionalidades básicas do sistema e, posteriormente, incrementar seu processo de desenvolvimento. A cada iteração, novas funcionalidades são adicionadas ao sistema até se chegar à versão final do *software*.

A utilização do modelo em espiral proporcionou o entendimento de cada etapa do projeto com máximo de controle de risco. Primeiramente foi construído um Controlador Virtual de tempo fixo e, após outros ciclos de desenvolvimento, foram sendo adicionadas funções de controle em tempo real.

Para a geração de código foi utilizado a ferramenta Borland C++ Builder 6 em ambiente Microsoft Windows 2000. A escolha deste ambiente se deve ao fato de este ser o mesmo sistema operacional utilizado pelo simulador Sitra B+. A linguagem C++ foi utilizada por apresentar a robustez herdada da linguagem C com a vantagem de utilizar os conceitos da orientação a objetos. Outro motivo de se utilizar a linguagem C++ é o fato de outros módulos que fazem parte do sistema também utilizarem esta linguagem.

Para a troca de informações entre os diversos módulos que se comunicam com o controlador, foram utilizadas conexões *socket*. As conexões *socket* utilizam o protocolo orientado a conexão TCP/IP. Este protocolo permite o controle das mensagens, dando garantia de que elas alcançaram o destino sem erros. Além de permitir a comunicação interprocessos que são executados em um mesmo dispositivo, as conexões *socket* permitem integrar processos de diferentes plataformas através de uma rede.

As mensagens trocadas através das conexões *socket* apresentam-se no formato XML. O formato XML forneceu flexibilidade na troca de mensagens, o que foi bastante útil visto que elas foram alteradas constantemente.

3.4.2 Estrutura do controlador

O Controlador Virtual é uma abstração de um controlador semafórico real para aplicação em um ambiente de simulação. Para simplificar o seu desenvolvimento, o código do controlador virtual foi dividido em três classes, a saber: Comunicador, Controlador e XML Parser. Cada classe ficou responsável por ações bem específicas sendo elas responsáveis por encapsular os dados pelos quais são responsáveis. Foi construído um diagrama de classes para expressar com detalhes as conexões necessárias entre os objetos e métodos que cada classe deve definir (LARMAN, 2002).

Posteriormente, será apresentada a organização, de forma mais detalhada, sobre cada uma das classes que formam o Controlador Virtual. A figura 3-5 mostra organização final destas classes que compuseram o Controlador Virtual.

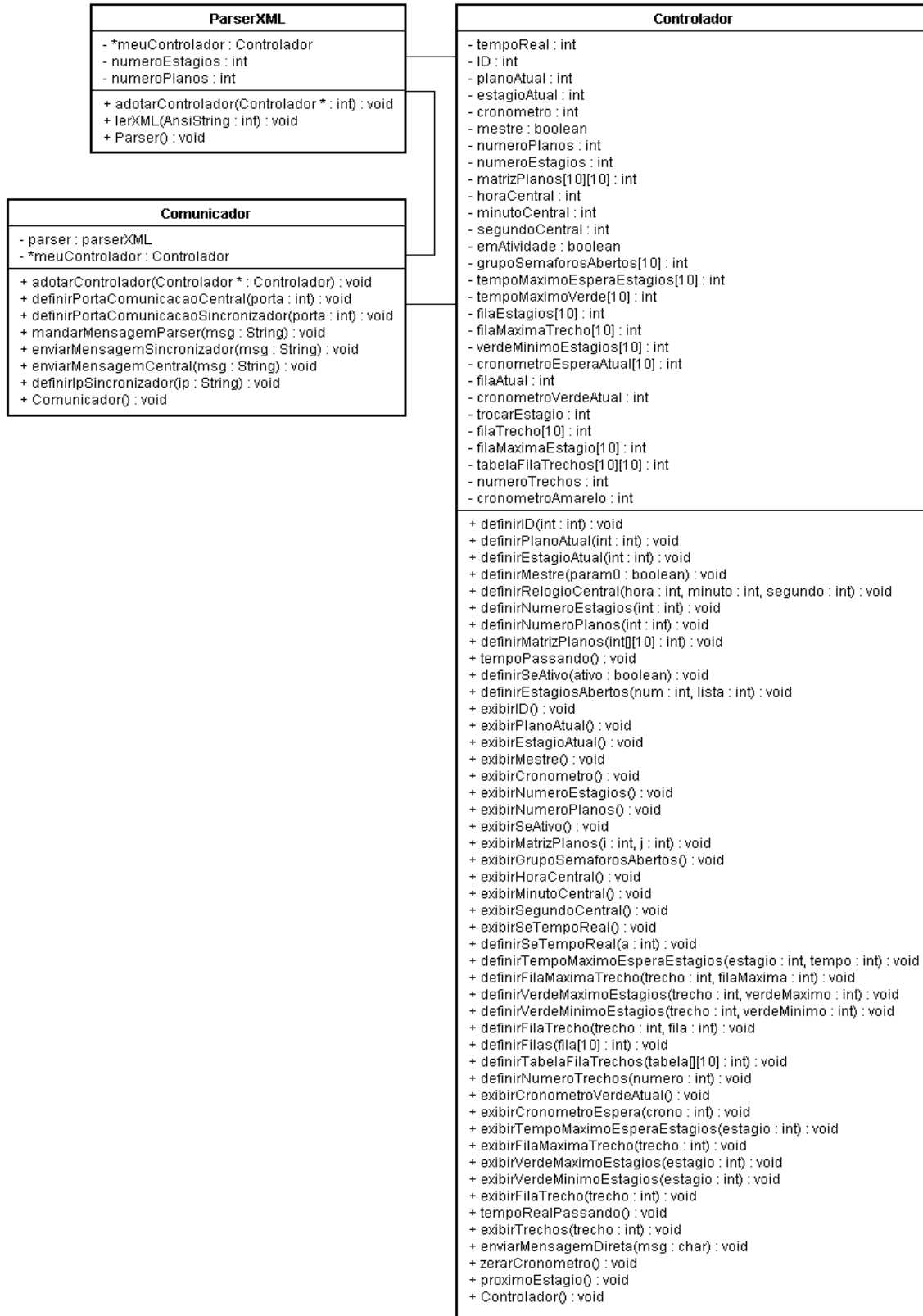


Figura 3-5 Diagrama de classes do controlador

3.4.2.1 - Classe Controlador

Esta é a classe que contém os métodos de controle de tráfego. Através do auxílio das classes Comunicador e ParserXML, esta classe realiza a tomada de decisão sobre a situação dos grupos semaforico com base nos dados que ela recebe. A classe Controlador não possui nenhuma referência para outras classes, mas as outras classes possuem ponteiros para ela.

O controlador virtual tem a capacidade de operar tanto em modo de tempo fixo quanto em tempo real. Para o controle em tempo fixo, a principal configuração guardada pelo controlador para que ele possa realizar seu trabalho é a tabela de tempos. Nesta tabela são guardados os tempos de verde de cada estágio em cada plano.

Para o controle em tempo real, um número maior de atributos de configuração é necessário, visto que existe uma maior complexidade de operação. Para o modo de execução em tempo real, a classe Controlador possui os seguintes atributos de configuração:

- Tempo máximo de espera dos estágios – Tabela que armazena, em segundos, o tempo máximo que cada estágio pode ficar em vermelho com veículos na fila;

- Tempo máximo de verde – Tabela que guarda os tempos máximos que cada estágio pode ficar em verde independentemente da situação dos demais trechos;
- Tempo de verde mínimo – Tabela com o tempo mínimo que cada conjunto de grupos semaforicos devem permanecer em verde quando for dado o direito de passagem;
- Tamanho máximo de filas – Tabela com número máximo de veículos em determinado trecho para forçar troca de estágio.

A apresentação destes atributos fornece uma idéia da maior complexidade do controle em tempo real em relação ao controle de tempo fixo. Mais adiante serão apresentados os algoritmos dos dois modos de operação.

3.4.2.2 - Classe Comunicador

A classe Comunicador é responsável por estabelecer a comunicação com as outras entidades computacionais que compõem o sistema de controle de tráfego. Os módulos com os quais esta comunicação é estabelecida são: a Central de Controle de Tráfego e o simulação de tráfego Sitra B+. Para se comunicar com a simulação foi utilizado um Processo Sincronizador que é um agente intermediário na comunicação entre eles.

O Processo Sincronizador realiza leituras e escritas nos arquivos de operação utilizados pelo programa Sitra B+ e mantém a sincronização entre a simulação e o controlador. Este processo foi desenvolvido por outro pesquisador, não fazendo este parte do presente trabalho.

A comunicação com a Central e com o Processo Sincronizador é realizada através de conexões *socket*. Para realizar a comunicação com a Central, foi criado um elemento *ServerSocket*, de modo que o Controlador aguarda conexões da Central. A Central tenta realizar a conexão na porta de comunicação do Controlador e este aceita a conexão. A partir daí, fica estabelecido um canal onde os dados podem trafegar em ambos os sentidos. Qualquer uma das entidades pode encerrar a comunicação;

A comunicação com a simulação de tráfego, que utiliza o Processo Sincronizador como intermediário, ocorre de maneira inversa à comunicação com a Central de Controle de Tráfego. Neste caso, o servidor da conexão é o programa Sincronizador e é o controlador o responsável por solicitar a conexão.

A função principal da classe Comunicador é enviar e receber dados das entidades externas ao controlador. Toda informação recebida por esta classe está em formato XML. Estas mensagens são repassadas para a classe ParserXML onde serão tratadas e extraídas as informações.

3.4.2.3 - Classe ParserXML

A classe ParserXML tem como objetivo decodificar as mensagens em formato XML que são recebidas da classe Comunicador e extrair delas informações. Um objeto da classe Comunicador adota uma instância desta classe. Ao ser criado um objeto da classe ParserXML, sua primeira etapa é receber um ponteiro para um objeto Controlador. É através deste ponteiro que será estabelecida a comunicação entre o parser e controlador.

O método principal da classe ParserXML é o lerXML(String). Este método faz a leitura das *tags* na mensagem e, dependendo delas, chama métodos da classe Controlador. Os dados contidos nas *tags* XML são passados como parâmetros para as funções da classe Controlador.

3.4.3 Controlador Virtual - Tempo Fixo

O modo de execução em tempo fixo foi implementado na primeira iteração do ciclo de desenvolvimento em espiral. Este modo de trabalho é o mais simples e, por este motivo, foi construído juntamente com os componentes básicos do controlador.

Para construção da funcionalidade do controle em tempo fixo foi necessário, primeiramente, construir uma tabela de planos. Esta foi representada por uma matriz de números inteiros onde as colunas representam os planos existentes e as linhas, seus respectivos estágios. Cada elemento da matriz representava, em segundos, o tempo determinado para cada um dos estágios de cada plano.

Foi construído também um cronômetro regressivo que capta o tempo do estágio corrente e o decrementa até zero. Quando é atingido o tempo máximo de um estágio, o controlador faz a troca de estágio, informando a simulação para que os devidos semáforos sejam alterados. Após a troca de estágio, o tempo do novo estágio é inserido no cronômetro e a ação se repete.

Para operar em tempo fixo, o controlador deve ser configurado por uma entidade externa com as seguintes configurações: número de estágios, número de planos e tabela de planos. Com base nestes dados, o controlador percorre a

tabela de planos e extrai a divisão do tempo entre os diferentes estágios. O fluxograma da figura 3-6 apresenta de forma mais detalhada seu funcionamento.

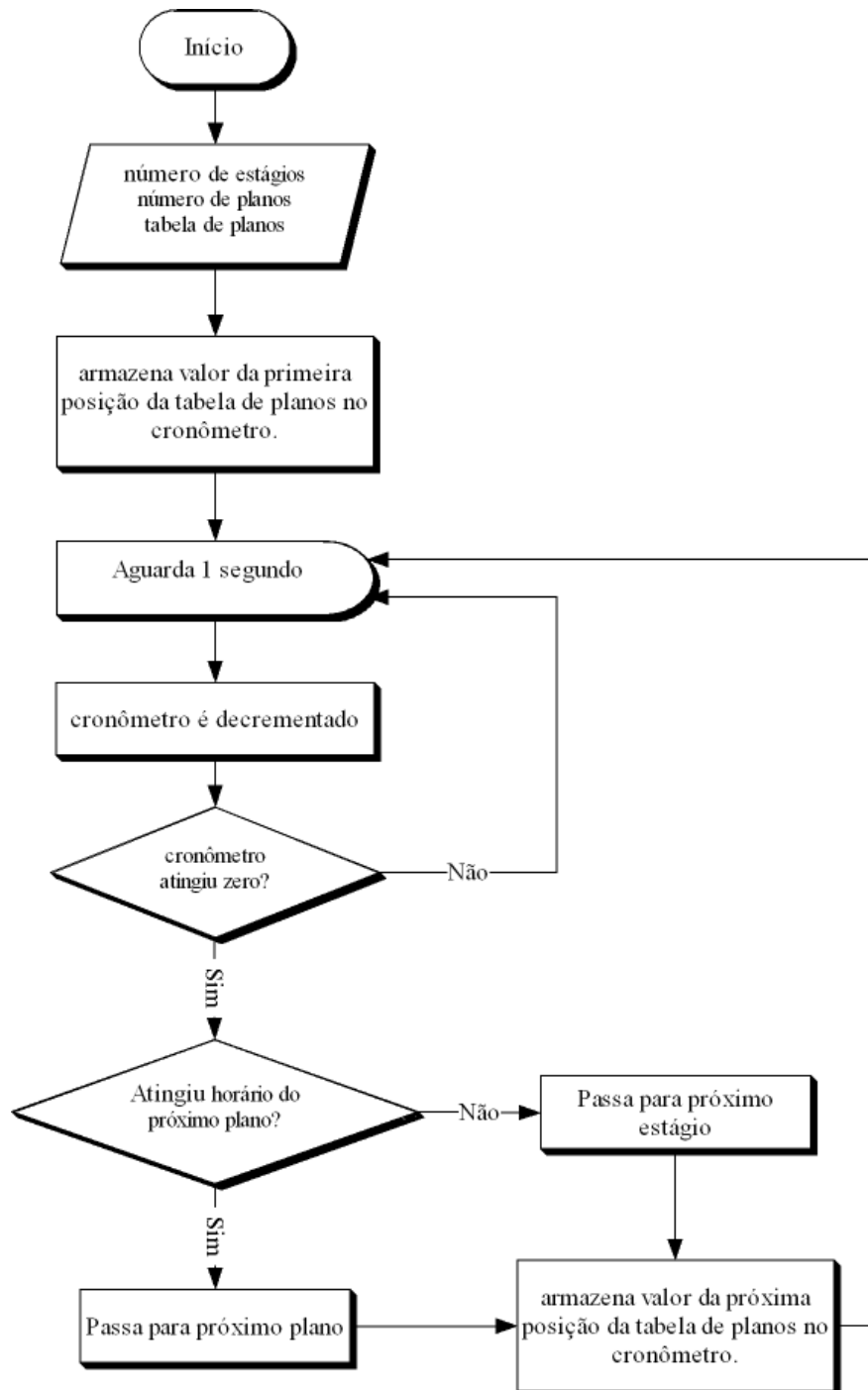


Figura 3-6 Fluxograma do modo em tempo fixo

Este controle apenas coloca em execução a configuração que lhe foi passada.

Neste modo de operação, as mensagens trocadas com a simulação são unidirecionais. Executando em tempo fixo não há necessidade de obter informações do estado da simulação, pois esta informação não é utilizada para tomada de decisão. Nesta primeira iteração do ciclo de desenvolvimento, não havia sido implementado o envio de informações para a Central de Controle, o controlador tinha a função de apenas colocar em execução a configuração que lhe foi passada.

A figura 3-7, onde as setas representam fluxo de informação, demonstra o funcionamento do sistema na primeira versão,

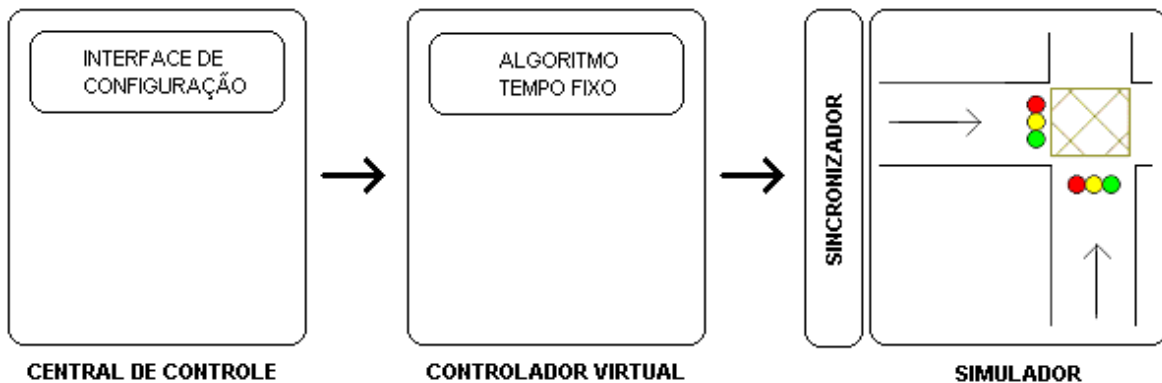


Figura 3-7 Funcionamento da primeira versão do Controlador

3.4.4 Controlador Virtual - Tempo Real

Dando continuidade ao desenvolvimento em espiral, funcionalidades do controle em tempo real foram adicionadas posteriormente. Nesta etapa foi necessária também a implementação do envio de informações para a Central de Controle, bem como a comunicação bi-direcional com a simulação.

A transferência de dados da simulação para o controlador se fez necessária para que a informação sobre a condição sobre o fluxo de veículos fosse passada para o controlador. Com base nestas informações, o controlador tem a capacidade de utilizar o algoritmo de controle em tempo real para atuar sobre os grupos semafóricos.

Para operar em tempo real, o controlador necessita de um número maior de atributos de configuração. Ao invés de utilizar a tabela de planos, presente no controle em tempo fixo, o controlador utiliza quatro outras tabelas de configuração: tabela de verde mínimo, tabela de verde máximo, tabela de fila máxima e tabela de tempo de vermelho máximo. Estas informações são atualizadas constantemente, captando dados da simulação em execução.

A figura 3-8 apresenta o funcionamento do controle em tempo real utilizado no controlador.

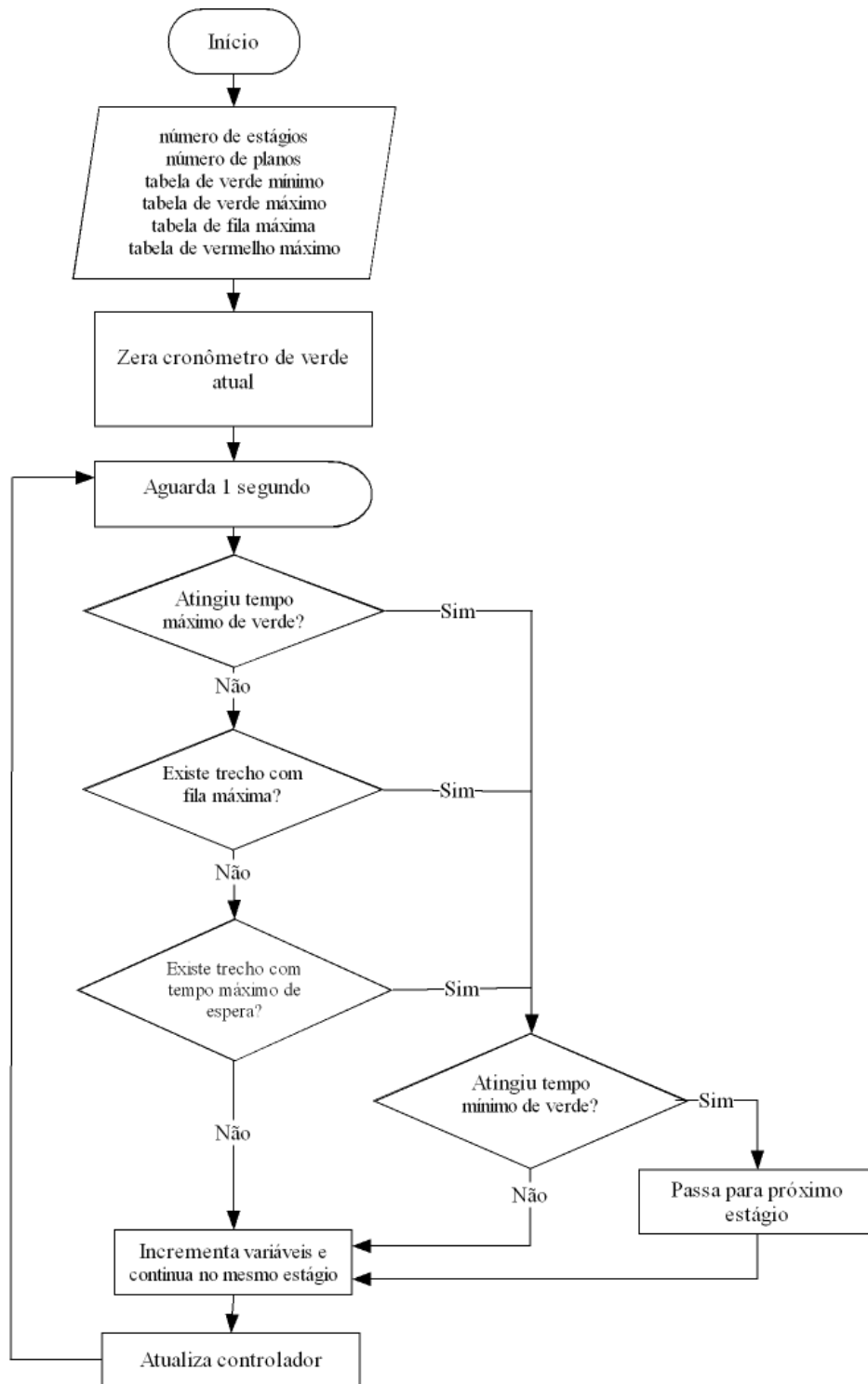


Figura 3-8 Funcionamento do controle em tempo real

Como foi possível perceber, ao final do fluxo de dados do controle em tempo real, é realizada uma atualização do controlador. O controlador deve, portanto, verificar o estado atual da simulação para que no próximo ciclo estes dados sejam utilizados na tomada de decisão.

Nesta etapa do projeto, foi necessária a implementação do envio de informações para a Central de Controle. A Central utiliza estas informações para realizar a monitoração dos fluxos veiculares. A figura 3-9 apresenta a organização do sistema na versão 2.0 do Controlador Virtual.

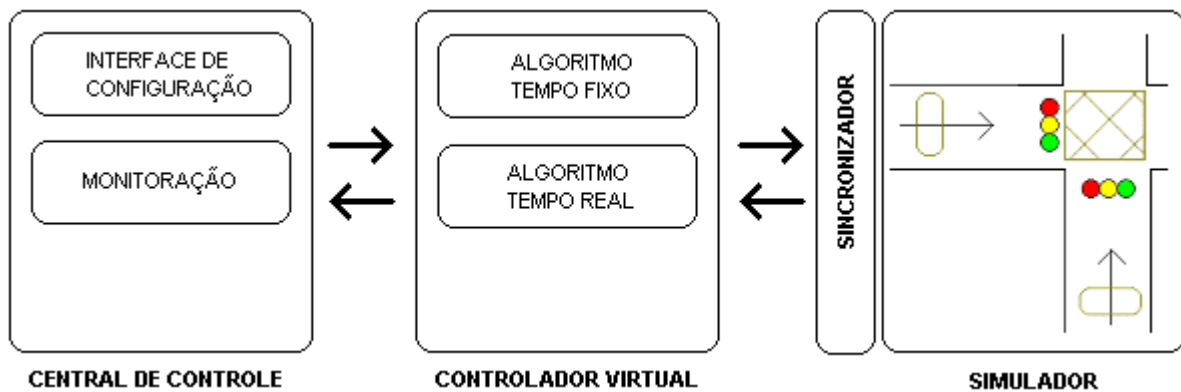


Figura 3-9 Funcionamento da segunda versão do Controlador.

3.4.5 Controlador Virtual - Modo Passivo

O desenvolvimento do módulo de controle passivo fez parte da terceira iteração do ciclo de desenvolvimento em espiral. Nesta etapa do projeto, foi implementado um módulo que permite que o Controlador Virtual seja controlado remotamente. Neste modo de operação, o Controlador envia uma quantidade maior de dados sobre o estado da simulação à Central de Controle. Isto permite que a Central execute um algoritmo próprio de controle em tempo real.

Quando o controlador está trabalhando no modo de controle passivo, nenhuma decisão é tomada por ele. O controlador torna-se responsável apenas em manter a Central informada sobre as condições das vias da simulação e atuar sobre a simulação com base nas decisões tomadas pela central de controle que opera em tempo real.

O controle em modo passivo permite à Central, que tem conhecimento sobre o estado de outros controladores, realizar seu trabalho com a finalidade de proporcionar melhorias do tráfego da simulação de forma global. No modo de controle em que é o Controlador virtual quem executa o algoritmo de tempo real, cada controlador tinha a capacidade de realizar melhorias apenas nas vias sobre seu domínio.

Um exemplo de melhoria que pode ocorrer utilizando o algoritmo de controle na central, é a coordenação dos movimentos. Isto aumenta a chance dos veículos que entram no domínio da Central de, ao passar por uma intersecção, encontrar os semáforos abertos nas próximas intersecções, evitando paradas freqüentes.

A figura 3-10 apresenta como ficou a versão final do sistema de controle com o Controlador Virtual na sua versão 3.0.

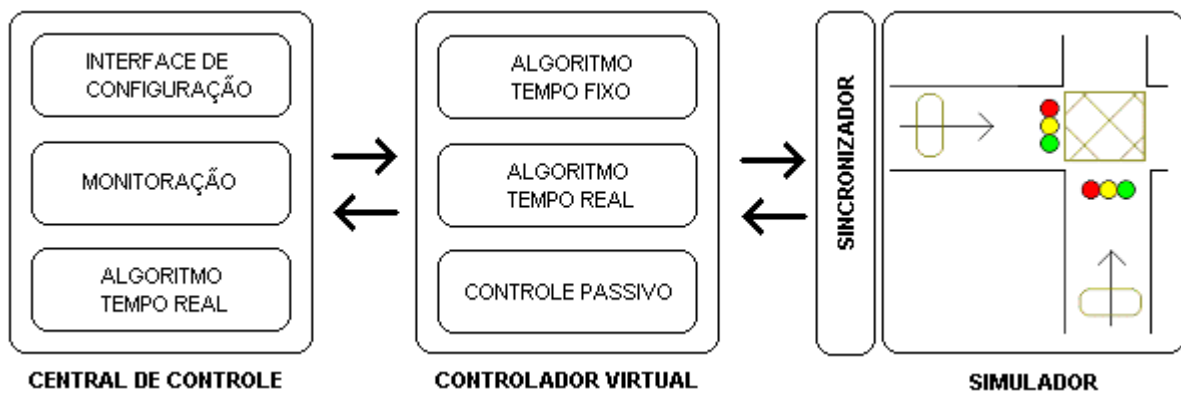


Figura 3-10 Funcionamento da terceira versão do Controlador.

4. Resultados

O presente trabalho resultou na construção de uma ferramenta que emula um controlador semafórico. O Controlador Virtual desenvolvido permite simular o funcionamento de um controlador que implemente o controle em tempo real. Além da utilização de um algoritmo de controle em tempo real implementado internamente ao controlador, este permite ser gerenciado por um algoritmo externo, fornecendo maior flexibilidade em sua utilização.

Além de operar em tempo real, foi desenvolvido um algoritmo de controle em tempo fixo para fins de comparações. Desta forma, o operador pode implementar um algoritmo de controle em tempo real e comparar seu desempenho com um controlador em tempo fixo padrão sobre as mesmas condições.

O acesso às configurações do controlador pode ser realizado remotamente, através de conexões sockets e troca de mensagens em formato XML. Para situações onde se deseja visualizar localmente o estado do controlador, foi desenvolvida uma interface gráfica que apresenta a configuração estabelecida, bem como o estado atual do controlador. É possível, também, visualizar a troca de mensagens efetuadas com as outras entidades do sistema.

O Controlador Virtual é dominado por uma Central de Controle que fornece as configurações necessárias para seu funcionamento e atua sobre a simulação através de um processo denominado Sincronizador. A forma como foi construído permite que ele se adapte para se comunicar com diferentes softwares de diferentes plataformas, desde que obedeçam aos requisitos do protocolo de comunicação proposto.

Além de versátil, o sistema desenvolvido tem a robustez de lidar com eventos inesperados. Ainda que uma outra aplicação tente estabelecer com ele uma troca de mensagens fora do padrão para o qual foi proposto, ele ignora essas mensagens inválidas evitando resultados inesperados.

O instrumento virtual construído alcançou seu objetivo. O Controlador Virtual demonstrou ser capaz de fornecer uma plataforma para avaliação de controle semafórico sobre um ambiente de micro-simulação de tráfego urbano.

5. Conclusões e Perspectivas

Durante o desenvolvimento do presente trabalho foi desenvolvido um controlador semaforico virtual que atua sobre uma micro-simulação de tráfego urbano. Este instrumento virtual atendeu as especificações a que se propunha.

Ainda que melhorias possam ser implementadas, o Controlador Virtual mostrou-se operacional. Através de testes realizados em laboratório, foi possível provar seu funcionamento dentro do esperado.

Uma proposta para trabalhos futuros seria a implementação de um protocolo de comunicação especificado por padrões internacionais na área.

Outra melhoria futura é tornar este controlador compatível com o TUC (*Traffic-responsive Urban Control*), uma nova estratégia de controle em tempo real, que utiliza um algoritmo baseado em dados históricos. Para isto, seria necessário mudar a estrutura do presente controlador, introduzindo um banco de dados para armazenar dados históricos.

Futuramente, após a revisão do sistema, espera-se que seja possível substituir a simulação pela realidade. Utilizar um controlador semaforico com a estrutura do Controlador Virtual proposto para que este atue sobre semáforos de intersecções reais.

6. Referências Bibliográficas

DENATRAN. **Manual de Semáforos**. Departamento Nacional de Trânsito e Companhia de Engenharia de Tráfego. Brasília, 1979.

BRITO, Ronei Mascarenhas. **Desenvolvimento de um Simulador para Análise e Projeto de Sistemas de Controle de Tráfego em Malha Fechada**. Florianópolis, 1998.

ALMEIDA, Mauricio B. **Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares**. Brasília, 2002.

WINSTON, Patrick Henry. **Inteligência Artificial**. Tradução: Carlos Octávio Pavel. Rio de Janeiro: Livros Técnicos e Científicos, 1988. 498p. ISBN 85-216-0609-5.

OLIVEIRA, Rômulo Silva. **Mecanismos de Adaptação para Aplicações Tempo Real na Internet**. Artigo SEMISH'98. Belo Horizonte-MG, agosto de 1998.

GAZIS, Denos C., **Traffic Theory**. 1930

JOHNSON, G.W. **LabVIEW graphical programming: practical applications in instrumentation and control**. New York: McGraw-Hill. 1997. 665p.

LARA, Natalina. **Sistemas de Tempo Real**. 2002,
<http://www.inf.uri.com.br/formandos/natalina.htm>. Acesso em 23 out. 2004.

FARINES, Jean-Marie and FRAGA, Joni da S. and OLIVEIRA, Rômulo S.
Sistemas de Tempo Real, Escola de Computação, São Paulo, 2000.

SOUZA, S. G. ; KRAUS JR., W. ; FARGES, J. L. . **Um algoritmo de busca em profundidade para controle ótimo em tempo real de tráfego urbano**. XIII Congresso Panamericano de Transportes, 2004, Albany, NY. Proceedings of the XIII Panamerican Congress of Transportation, 2004. v. 1. p. 12-23.

SELIC, Bran. **The Real-Time Standard: Definition and Application**, Proceedings of the 2002 Design, Automation and test in Europe conference and Exhibition, 2002

MARQUES, José Alves, GUEDES, Paulo, **Fundamentos de Sistemas Operativos**, Editorial Presença, 1ª Edição, Lisboa 1990.

MICHAELIS. **Michaelis**: Moderno Dicionário da Língua Portuguesa. Editora Melhoramentos. 1998.

ALMEIDA, Mauricio B. **Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares**. Brasília, 2002.

BORATI, Isaias Camilo. **Programação Orientada a Objetos usando Delphi**. Visual Books. Florianópolis, 2001.

LARMAN, Craig (2002). **Applying UML and Patterns An Introduction to Object-Oriented Analysis and Design and the Unified Process**. 2nd Edition. Prentice-Hall, Inc., 1999.

PRESSMAN, R. S. **Engenharia de Software**. São Paulo, Makron Books, 1995.

NIA, **National ITS Architecture**: <http://www.its.dot.gov>. Acesso em 10 dez. 2004.

Claramunt C., Jiang B., 2001, **A Qualitative Model for the Simulation of Traffic Behaviours in a Multi-lane Environment**, *Journal of Geographical Sciences*, v. 11, Special issue on geo-visualization, pp. 29-42.

VILLAMIL, Marta B. **Simulação de Grupos de Humanos Virtuais utilizando Abordagens Micro e Macroscópicas**. MSc. dissertation. 2003.

7. Anexos

7.1 Interfaces do Controlador Virtual

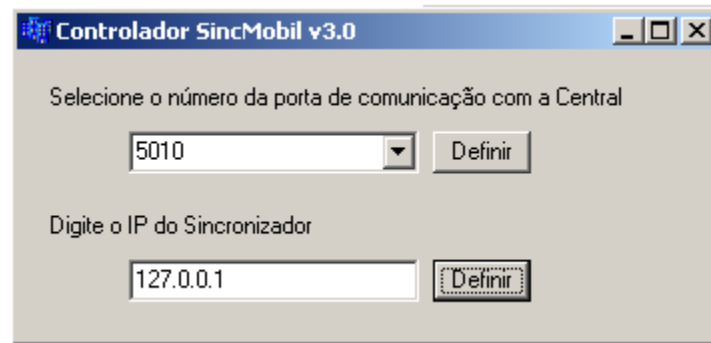


Figura 7-1 Tela inicial de configuração

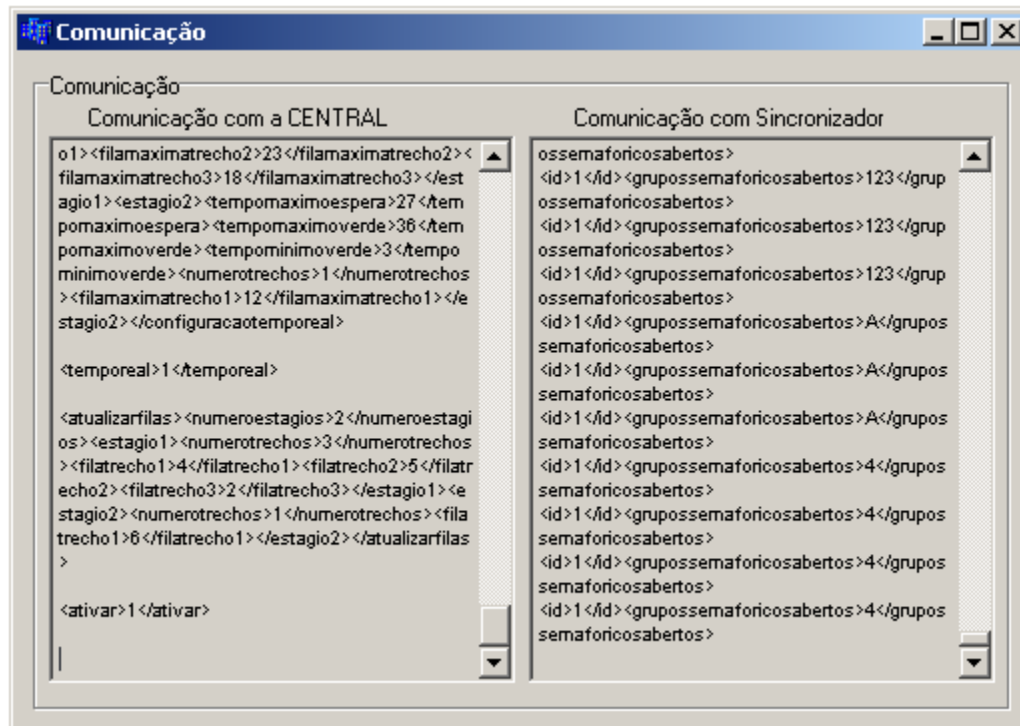


Figura 7-2 Troca de mensagens entre entidades

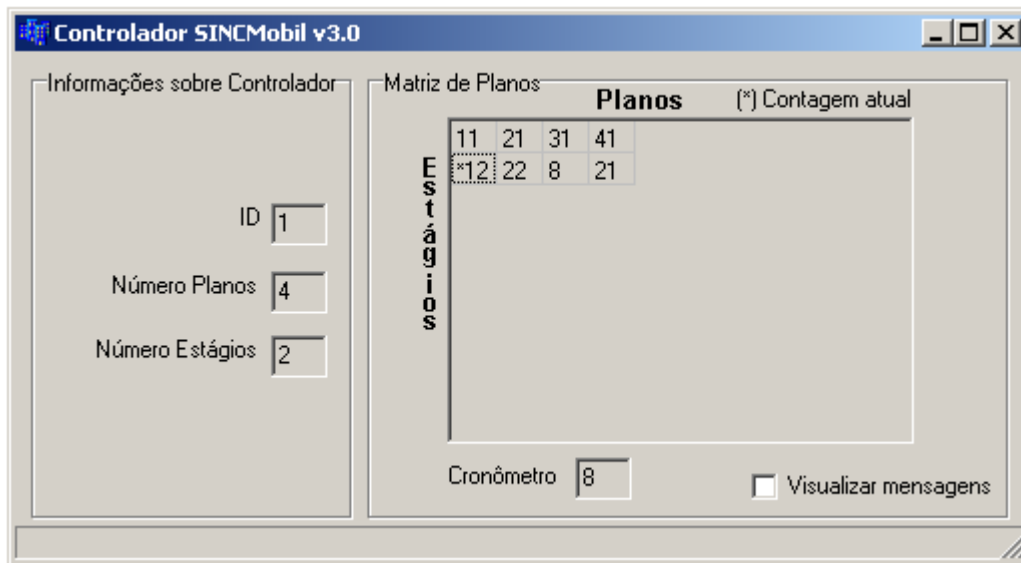


Figura 7-3 Controle em tempo fixo

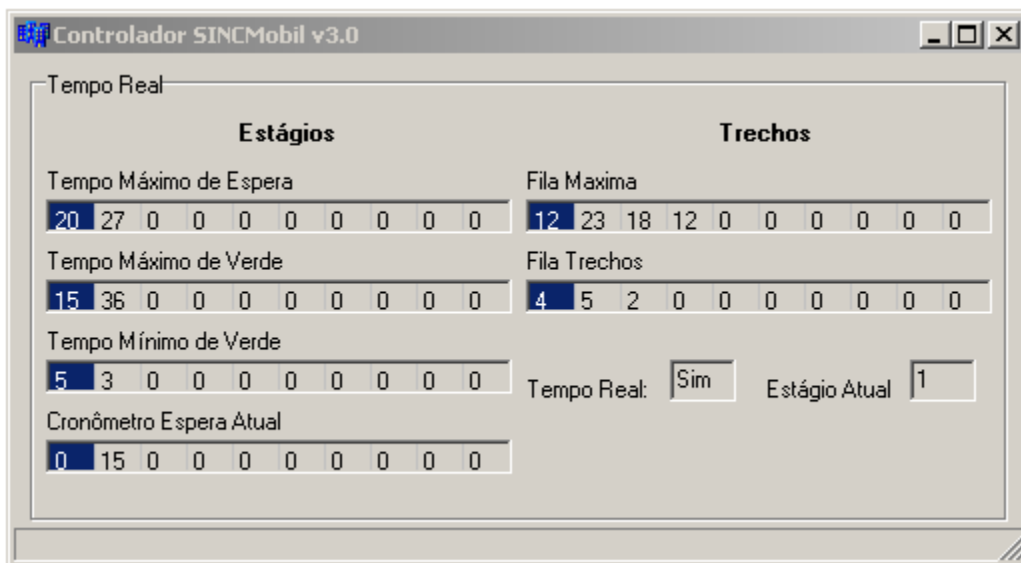


Figura 7-4 Controle em tempo real

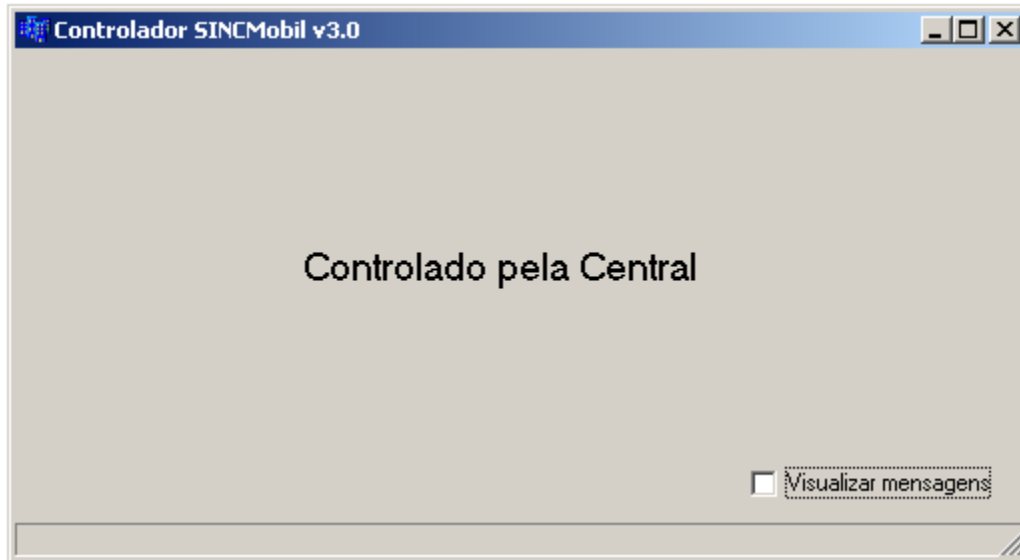


Figura 7-5 Controle em modo passivo

7.2 Mensagens XML

```
<?xml version="1.0" encoding="iso-8859-1"?>
<central>
  <configurar>
    <id>2</id>
  </configurar>
</central>
```

Figura 7-6 Mensagem que configura as identidades dos controladores.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<controlador>
  <sincronizar>
    <hora>14</hora>
    <minuto>50</minuto>
    <segundo>35</segundo>
  </sincronizar>
</controlador>
```

Figura 7-7 Mensagem que ajusta o relógio do Controlador.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<central>
  <setarestagios>
    <numeroestagios>2</numeroestagios>
    <estagio1>123</estagio1>
    <estagio2>4</estagio2>
  </setarestagios>
</central>
```

Figura 7-8 Mensagem que seta os estágios do Controlador, onde estágio é um conjunto de semáforos que fica verde ao mesmo tempo.


```

<?xml version="1.0" encoding="iso-8859-1"?>
<central>
  <setarplanos>
    <numeroplanos>3</numeroplanos>
    <plano1>
      <estagio1>35</estagio1>
      <estagio2>45</estagio2>
    </plano1>
    <plano2>
      <estagio1>70</estagio1>
      <estagio2>50</estagio2>
    </plano2>
    <plano3>
      <estagio1>10</estagio1>
      <estagio2>12</estagio2>
    </plano3>
  </setarplanos>
</central>

```

Figura 7-9 Mensagem que define os planos de tempo fixo do Controlador, onde plano é o conjunto de tempos de verde de uma seqüência de estágios.

```

<?xml version="1.0" encoding="iso-8859-1"?>
<ativar>1</ativar>

```

Figura 7-10 Mensagem que ativa ou pausa o Controlador. Se o parâmetro for '1' o Controlador inicia sua atividade, se '0', o Controlador é parado.

```

<?xml version="1.0" encoding="iso-8859-1"?>
<temporeal>1</temporeal>

```

Figura 7-11 Mensagem que define o tipo de operação do sistema. Se o parâmetro for '0', o controlador opera com os planos de tempo fixo. Se o parâmetro for '1' o Controlador opera em tempo real local. Se o parâmetro for '2' o sistema opera em modo passivo. O padrão do Controlador, assim que ele é iniciado, é a operação em tempo fixo.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<configuracaotemporeal>
  <numeroestagios>2</numeroestagios>
  <estagio1>
    <tempomaximoespera>95</tempomaximoespera>
    <tempomaximoverde>60</tempomaximoverde>
    <tempominimoverde>15</tempominimoverde>
    <numerotrechos>3</numerotrechos>
    <filamaximatrecho1>12</filamaximatrecho1>
    <filamaximatrecho2>23</filamaximatrecho2>
    <filamaximatrecho3>18</filamaximatrecho3>
  </estagio1>
  <estagio2>
    <tempomaximoespera>77</tempomaximoespera>
    <tempomaximoverde>90</tempomaximoverde>
    <tempominimoverde>25</tempominimoverde>
    <numerotrechos>1</numerotrechos>
    <filamaximatrecho1>22</filamaximatrecho1>
  </estagio2>
</configuracaotemporeal>
```

Figura 7-12 Mensagem que configura o Controlador para operação em tempo real.