Simulação de Transportes Sobre Trilhos Usando Autômatos Celulares para os Trens e Agentes Inteligentes para os Usuários

Daniel M. G. Morais, Luciano A. Digiampietri

[1]Escola de Artes, Ciências e Humanidades – Universidade de São Paulo (USP)
Av. Arlindo Bettio, 1000 – CEP 03828-000 – São Paulo – SP – Brazil

Abstract. *This paper discusses the development of a model of an on rails transport simulation system that combines the usage of cellular automata for the rails and intelligent agents for the user behavior. This paper also presents a framework for the development of this kind of application allowing interconnections between then in the same simulation, allowing the simulation of different transport modals.*

Resumo. *Este artigo apresenta o desenvolvimento de um modelo para a simulação de sistemas de transporte sobre trilhos combinando autômatos celulares para representar os trens e agentes inteligentes para o comportamento do usuário. É apresentado também um framework para o desenvolvimento deste tipo de aplicação tendo em vista a conexão com outros modelos de simulação a fim de possibilitar a simulação de transporte multi-modal.*

1. Introdução

Com o aumento na população das grandes cidades e, por conseguinte, o aumento da quantidade de deslocamentos de pessoas, é necessário o adequado planejamento do sistema de transporte urbano, visando a minimizar o tempo de deslocamento, melhorando a qualidade de vida da população e o próprio funcionamento do ambiente urbano.

A ineficiência do sistema de transporte público, provocada pelo aumento da demanda, má distribuição dos fluxos e também o aumento da procura pelo transporte individual, causado inclusive pela ineficiência do transporte público, acaba por ampliar a saturação do sistema, aumentando o tempo de viagem do usuário, levando deslocamentos que, em condições ideais, demorariam poucos minutos, a demorar horas. Deslocamentos estes feitos em condições estressantes, seja pelo transporte público superlotado, seja pelos imensos congestionamentos encontrados nas vias de tráfego.

Sendo assim, é importante que as autoridades responsáveis pelo planejamento de sistemas de transportes possuam ferramentas adequadas para o planejamento de transporte urbano, permitindo assim a maior fluidez possível para o sistema em todas as condições, aumentando a qualidade de vida da população e eficiência econômica da cidade.

Este artigo apresenta um modelo e sua implementação para a simulação de sistemas de transporte sobre trilhos, prevendo a utilização de autômatos celulares para o funcionamento das linhas e a utilização de agentes inteligentes para a representação do comportamento dos usuários.

O restante deste artigo está organizado da seguinte forma. Na seção 2, são expostos os conceitos que fundamentam este trabalho. Na seção 3 é apresentado o cenário

utilizado como base para o modelo. A seguir, na seção 4, é apresentado o framework desenvolvido como base para a construção do modelo. Na seção 5, o modelo em si é apresentado. Em seguida, na seção 6, é feita uma discussão sobre o modelo proposto e, enfim, na seção 7, são feitas as considerações finais acerca deste trabalho.

2. Fundamentação Teórica

Antes de apresentar o modelo proposto por este trabalho, é importante contextualizar alguns conceitos acerca do tópico estudado, de forma a fundamentar algumas decisões tomadas durante o desenvolvimento. Serão apresentados conceitos acerca de simulação de sistemas de transporte e, também a fundamentação sobre os motivos da utilização de autômatos celulares [Nagel 2002] para abordar o assunto.

A simulação de sistemas de transporte pode ser classificada de acordo com o seu nível de observação dos componentes em três categorias: microscópica [Klügl and Rindsfüser 2007, Usher et al. 2010, Kretz et al. 2011, Xiaobei et al. 2009, Pan et al. 2010, Ma et al. 2008, Ma and Yang 2009, Gao et al. 2009], macroscópica [Xue et al. 2009] e mesoscópica [Jin and White 2012], descritos em [Barceló 2010].

A simulação microscópica é baseada na descrição de movimento de cada componente do fluxo de tráfego, onde cada qual possui um comportamento próprio definido em função do ambiente e de outros componentes do sistema. Para a simulação de veículos, um dos modelos mais populares é o *Car-Following* [Huixin and Wenhong 2010, Wang et al. 2009], para a simulação de diversas faixas de rolagem em rodovias, por exemplo, onde o comportamento do carro é determinado pelo do veículo imediatamente a sua frente, e cada qual possui informações como aceleração, frenagem e perfil de tomada de decisão próprios.

Por sua vez, a simulação macroscópica se preocupa com o comportamento de massa. Ao invés do papel do indivíduo, o que se tem na simulação é a modelagem do ambiente e a identificação do comportamento da massa, observando parâmetros como densidade, volume e velocidade de deslocamento (desta massa, e não do indivíduo). Uma implementação tradicional deste tipo de simulação é o estudo do deslocamento de usuários em um determinado ambiente utilizando mecânica dos fluídos para a descrição do comportamento.

Já a simulação mesoscópica consiste em uma generalização da simulação microscópica sem, contudo, apresentar totalmente o comportamento de massa típico das simulações macroscópicas. Trata-se de modelagens híbridas, unindo aspectos de ambos os tipos anteriormente expostos, de forma a reduzir o custo da abordagem microscópica, sem perder as características de indivíduo. Um exemplo deste tipo de abordagem é a utilização de grupos representando um conjunto idêntico de características (por exemplo, os usuários que vão de um ponto A até um ponto B, no mesmo instante, são tratados como um usuário só).

Ainda, de acordo com uma revisão sistemática acerca do assunto ([Morais and Digiampietri 2012]), foi observado que um grande número de trabalhos focados na simulação de sistemas de transporte opta pela utilização de modelos microscópicos para a simulação, de forma a obter maior fidelidade sobre o indivíduo e

seus efeitos sobre todo o cenário. Ainda, a abordagem mais popular acerca do assunto é a utilização de agentes inteligentes para a modelagem de comportamentos complexos (por exemplo, o deslocamento de usuários na multidão, em [Chen et al. 2010]), enquanto, quando há a possibilidade de simplificação, utiliza-se autômatos celulares, como por exemplo, no trabalho ([Ma et al. 2011]), onde é modelado o comportamento de veículos em uma faixa apenas, atravessando uma via, com velocidade constante, onde tal comportamento é modelado através de autômatos celulares, porém com a simulação de usuários atravessando a via sendo feita através de agentes inteligentes.

Fundamentando teoricamente o estudo de sistemas de transporte, o trabalho de [Rodrigue et al. 2007] categoriza as principais estruturas utilizadas neste estudo. Em especial, são apresentados os conceitos de "Nó", como representação de polos de atividade urbana ou acessibilidade a sistemas de transporte, na forma de estações de trens e terminais de ônibus, por exemplo e "Ligações", que são a representação da infraestrutura utilizada para o deslocamento entre os nós, como por exemplo, rodovias, vias arteriais ou mesmo os trechos de deslocamento entre estações de trens.

Tendo em vista o exposto e em especial a revisão cuidadosa da literatura realizada em [Morais and Digiampietri 2012], neste trabalho adotamos o modelo de autômatos celulares para a simulação do sistema de trens, e agentes inteligentes para a simulação de usuários. A opção por autômatos celulares para trens foi tomada devido às características do ambiente, onde temos um ambiente controlado e com baixa complexidade nas decisões tomadas, onde as mudanças de parâmetros em geral são globais e afetam todos os componentes de forma idêntica (por exemplo, redução de velocidade ou alteração no tempo de parada em estação). Por sua vez, a utilização de agentes inteligentes para usuários visa a refletir não só a diferenciação entre eles (há uma variedade muito grande entre origens e destinos, por exemplo), como também a possibilidade de termos perfis de comportamentos distintos com uma infinidade de efeitos sobre o sistema [Wooldridge 2006].

3. Descrição do Cenário

Estre trabalho se propõe a apresentar um modelo para simulação de transportes sobre trilhos utilizando autômatos celulares para o funcionamento e operação da linha e agentes inteligentes para a simulação de comportamento e processo de decisão do usuário.

Para a construção do modelo proposto, consideramos como cenário um sistema de múltiplas linhas de trens interligadas a partir de transferências feitas pelo usuário, a partir de interconexões. Para ilustrar o modelo, considere o sistema de metrô gerido pela Companhia do Metropolitano de São Paulo (Metrô).

Atualmente, tal sistema é composto por cinco linhas, Vermelha, Azul (Linha 1), Verde (Linha 2), Vermelha (Linha 3), Amarela (Linha 4) e Lilás (Linha 5), sendo que esta última não possui ligação direta com outras linhas do sistema. Temos então a linha Vermelha, que é interligada com a linha Azul a partir da estação Sé e com a Amarela a partir da estação República, a linha Azul, interligada com a linha Amarela a partir da estação Luz e com a linha Verde pelas estações Paraíso e Ana Rosa, e por fim, temos a linha Verde, que conecta-se com a linha Amarela pela estação Consolação/Paulista (Figura 1). Desta forma, para diversos destinos é possível a adoção de mais de uma rota, fazendo diversas baldeações. No modelo foram então utilizadas as linhas Vermelha, Azul, Verde e Amarela, pelas interconexões entre as mesmas.

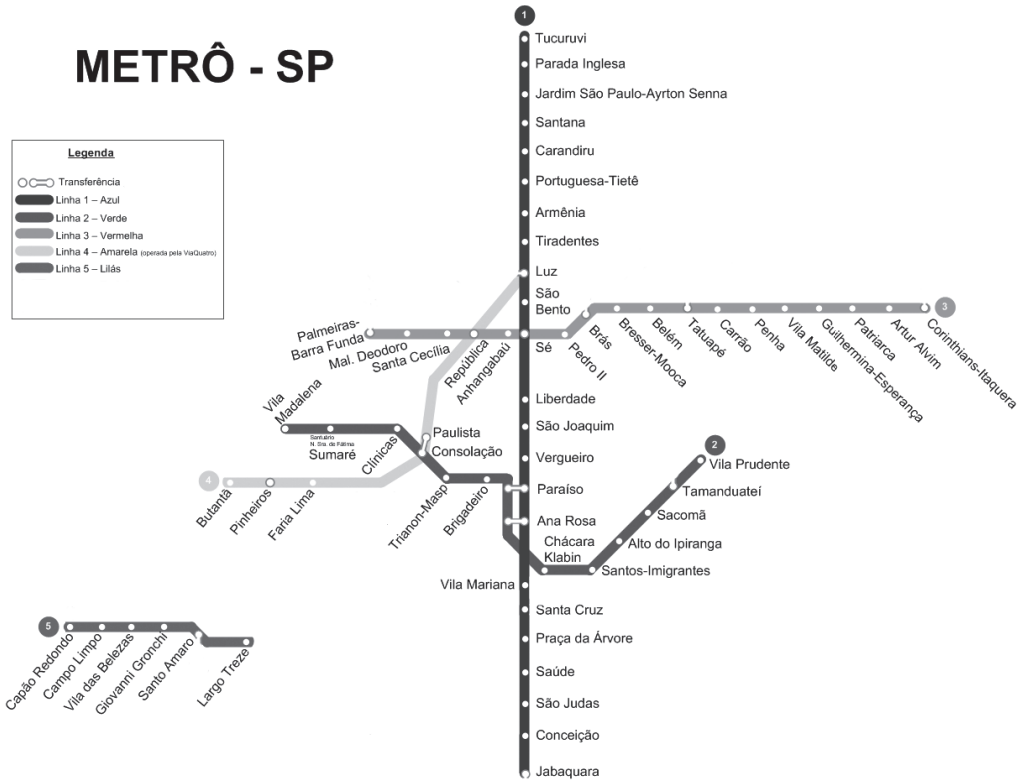

Figura 1. Representação esquemática de linhas e conexões do Metrô de São Paulo - Adaptação de [Metro 2014]

Como estruturas de dados, podemos entender o modelo como um grafo, onde cada nó representa uma estação, e as arestas representam os túneis e conexões, cada qual com seu respectivo peso (indicando o custo do deslocamento entre os nós). Nos túneis, trafegam apenas trens, enquanto nas conexões, o trajeto é feito pelo usuário a pé.

O que o modelo apresentado neste trabalho propõe é a representação computacional da operação de tal sistema de trens através de autômatos celulares, combinado à representação dos usuários e seu comportamento, como agentes inteligentes, e também o seu relacionamento. Ou seja, temos o sistema de trens operando de forma isolada e cabe ao usuário reagir aos sinais passados pelas diferentes situações ocorridas no sistema de trens. Ainda, para a construção de tal modelo, é também proposto um framework, o qual permita a reutilização e modelagem de outros sistemas de simulação de transporte.

4. O Framework

Para este trabalho, um framework foi implementado de maneira incremental, à medida que o modelo foi especificado e desenvolvido, observando como premissas a reutilização de código e possibilidade de utilização de modelos de simulação como componentes interconectados da perspectiva do usuário. Para tal, foram definidas uma série de interfaces que devem ser implementadas pelas classes utilizadas no modelo em questão. São elas:

- *ISystem*: Interface para a representação de um determinado sistema de transporte. No nosso cenário, o Metrô é um sistema. Outras possibilidades são, por exemplo,

um conjunto de linhas de ônibus logicamente agrupadas, por exemplo, as linhas de ônibus que atendem a região sul da cidade;
- *iStructure*: Representa uma estrutura de distribuição de usuários, no caso do nosso cenário, uma estação de trem. O equivalente a um nó do grafo. Em outros cenários, pode representar, por exemplo, um terminal de ônibus;
- *ITransition*: Utilizada para a representação de ligações entre estruturas, por exemplo, os túneis que ligam estações ou as transferências entre as mesmas. É o equivalente a arestas de um grafo;
- *IPassengerQueue*: Representa uma fila de passageiros/usuários do sistema. Utilizada para representar, por exemplo, uma plataforma, onde os passageiros aguardam a chegada do trem;
- *IOrientation*: Interface utilizada para a sinalização de direção no sistema. No caso de linhas de trem, podemos ter uma direção $\mathbf{A} \rightarrow \mathbf{B}$ e uma direção $\mathbf{B} \rightarrow \mathbf{A}$, onde $\mathbf{A}$ e $\mathbf{B}$ são estações terminais de uma determinada linha. Não necessariamente precisa possuir uma representação $\mathbf{A} \rightarrow \mathbf{B}$, $\mathbf{B} \rightarrow \mathbf{A}$, podendo representar qualquer situação de orientação pertinente ao modelo representado;
- *IVehicle*: Utilizada para a definição de um veículo sobre o sistema, no caso, um trem. É capaz de coletar e liberar usuários.

A intenção deste framework é permitir a construção de modelos computacionais distintos interligados, por exemplo, a construção da modelagem de um sistema de transporte sobre trilhos, por sua vez interligado a outro modelo, de ônibus urbanos. Desta forma, o framework é responsável, por exemplo, pelo desenho de rotas para o usuário, assim como por gerenciar a execução destes modelos de forma sincronizada.

A sincronização é feita através de ciclos. Ciclos são, para este framework, janelas de tempo definidas pelo desenvolvedor nas quais ocorre um conjunto de ações dentro de cada sistema, como por exemplo, o deslocamento de trens por um trecho curto, ou um intervalo de parada em uma estação. Desta forma, tem-se como ideia que um ciclo seja uma unidade atômica para o modelo em questão.

5. Modelagem de Transporte Sobre Trilhos Utilizando Autômatos Celulares

Utilizando o framework desenvolvido, foi criada uma estrutura de autômatos celulares para a representação do sistema de metrô. Isto ocorre em duas etapas. Na primeira, é definida uma estrutura de grafos que traduz o sistema de transporte em uma estrutura computacional. Temos então uma classe chamada *Station*, implementando *IStructure*, que representa uma estação assim como um nó do grafo; uma classe chamada *Tunnel*, que implementa *ITransition*, e que representa a ligação entre duas estações coberta por trens; e uma classe chamada *Connection*, também implementando a interface *ITransition*, a qual representa ligações entre duas estações de linhas diferentes, percorrida a pé, ambas as classes representando arestas do grafo. Tanto a classe *Station* quanto a classe *Tunnel* apresentam como parâmetro seu comprimento em células – $\mathbf{C}(\mathbf{station})$ e $\mathbf{C}(\mathbf{tunnel})$, respectivamente –, levando em conta a representação do sistema como automato celular, enquanto a classe *Connection* tem, além da distância em metros, o tempo em ciclos que o usuário leva para percorrê-la.

Assim, a representação de uma linha é feita apenas por objetos da classe *Station* e *Tunnel* enquanto as ligações com outras linhas são feitas a partir da classe *Connection*.

Esta representação é utilizada para dois objetivos: representar o sistema da perspectiva do usuário, para que o mesmo decida qual a rota será usada, com arestas com peso, e também para a transformação do modelo em estruturas de autômatos celulares, para a execução do modelo.

Ainda, cada linha é subdividida em dois segmentos, $\mathbf{Si}$, representando os segmento de ida e $\mathbf{Sv}$, representando o segmento de volta, representando a orientação das arestas do grafo, onde cada um deles possui duas filas associadas, as quais representam os pátios de início ($\mathbf{Qi}$) e término ($\mathbf{Qf}$) destes segmentos, onde um trem inicia seu percurso em $\mathbf{Qi}$, atravessa todas as células e termina seu percurso sendo enfileirado em $\mathbf{Qf}$. Ainda, temos que $\mathbf{Qi}(\mathbf{Si}) = \mathbf{Qf}(\mathbf{Sv})$ e $\mathbf{Qf}(\mathbf{Si}) = \mathbf{Qi}(\mathbf{Sv})$. Cada um destes segmentos tem um comprimento em células $\mathbf{C}$, definido por:

$$C(S) = \sum_{j=1}^{m} C(station_m) + \sum_{k=1}^{n} C(tunnel_n)$$

onde $\mathbf{m}$ é o número de estações e $\mathbf{n}$ o número de túneis registrados na simulação.

Cada trem é representado por uma instância da classe *Train* ($\mathbf{T}$) que implementa *IVehicle* e definido pelas seguintes características:

- Comprimento – $\mathbf{C}(\mathbf{T})$: O número de células ocupadas pelo veículo;
- Velocidade – $\mathbf{V}(\mathbf{T})$: Número de células percorridas pelo trem a cada ciclo;
- Contêiner de Usuários – $\mathbf{Uc}(\mathbf{T})$: Estrutura de dados para armazenar os usuários que estão viajando neste trem;
- Capacidade de Absorção – $\mathbf{A}(\mathbf{T})$: Número máximo de usuários coletados a cada parada em estação.
- Tempo de Parada – $\mathbf{Tp}(\mathbf{T})$: O tempo que o trem está parado.

Ainda, como parâmetros da linha ($\mathbf{L}$), temos:

- Número de trens operando simultaneamente – $\mathbf{N}(\mathbf{L})$: O número de trens que estarão ativos na linha simultaneamente;
- Tempo de parada – $\mathbf{Tp}(\mathbf{L})$: O número de ciclos que um determinado trem fica estacionado em uma determinada estação para embarque e desembarque de passageiros;
- Distância padrão entre trens – $\mathbf{\Delta L}$: Número de células de distância que o trem deve manter do trem seguinte, de forma a gerar um intervalo regular entre a passagem de trens na estação. Esta distância é dada por:

$$\Delta L = \frac{C(S) - C(T) * N(L)}{N(L)}$$

- Distância mínima entre trens – $\mathbf{\Delta_{min} L}$: A distância mínima que um trem deve manter de seu sucessor de forma a evitar acidentes. O mínimo para este valor é $\mathbf{1}$.

A segunda etapa é a transformação do modelo da linha de grafo em uma estrutura para aplicação de autômatos celulares, representado de maneira simplificada na figura 2, a qual mostra os segmentos próximos às conexões e as respectivas conexões entre as linhas. Para tal, cada um dos segmentos é convertido em um array multidimensional com

$\mathbf{C(S)}$ posições, cada qual representando uma célula. A transição de estados é feita a partir da ocupação ou não da célula por um trem e sua respectiva ação ao atingir determinada célula. Então os estados possíveis para uma célula são:

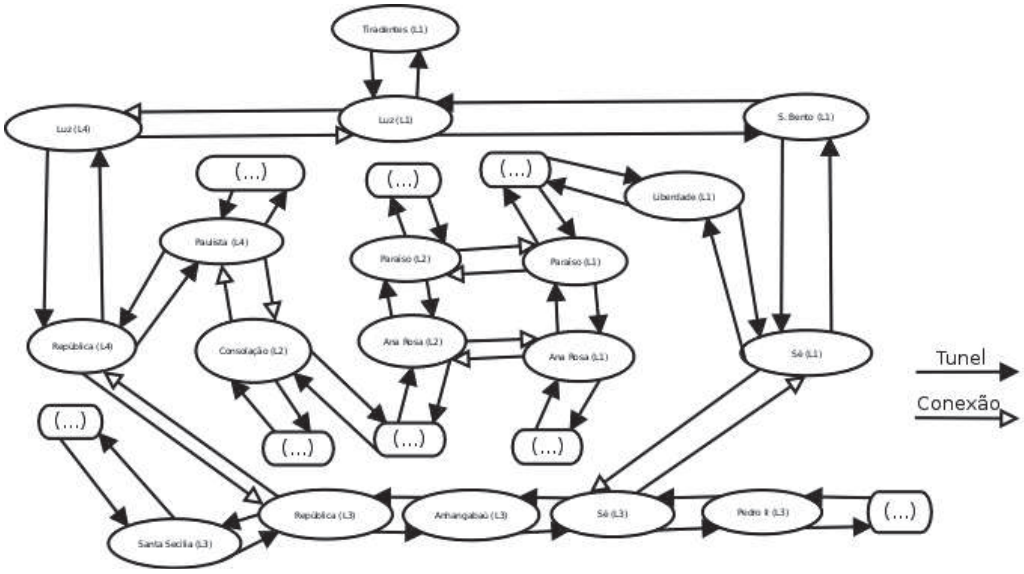

Figura 2. Representação simplificada do grafo referente ao Metrô de São Paulo

- Célula vazia – $\mathbf{P0}$;
- Célula ocupada por um pedaço de um trem em movimento – $\mathbf{P1}$;
- Célula ocupada por um pedaço de um trem parado em estação – $\mathbf{P2}$ (P2);
- Célula ocupada por um pedaço de um trem parado devido a ter atingido o limite da distância entre trens – $\mathbf{P3}$;

A transição entre estados de uma célula é feita a partir do algoritmo especificado no algoritmo 1. O algoritmo apresenta a execução do autômato por um ciclo apenas. Note que, no intervalo de um ciclo, há $\mathbf{K}$ transições de estado para cada célula, onde $\mathbf{K}$ é o número de células percorridas por um trem por ciclo, e a movimentação de um trem pode ser interrompida nas situações a) a sua posição máxima no momento coincide com a posição máxima de uma estação; e b) a distância entre a posição máxima do trem e seu trem subsequente é menor ou igual a $\mathbf{\Delta L}$.

5.1. Gerenciamento e Deslocamento dos Usuários

Para o gerenciamento do usuário, temos, nas estações, as filas de embarque de cada segmento (representando as plataformas) e as filas de saída e transferência, sendo a primeira para aqueles usuários que atingiram seu destino e as de transferência aquelas que gerenciam as transferências entre linhas.

Por sua vez, o usuário foi definido como um agente inteligente. Ele observa o ambiente e reage a estímulos de acordo com o seu caminho planejado. O caminho a ser seguido é construído então como uma fila com tuplas compostas pela estação e a ação a ser tomada naquela estação (sendo elas DESEMBARCAR, EMBARCAR, IGNORAR). Um usuário responde aos seguintes eventos:

```
   Entrada: Um array S representando o segmento onde ocorrerá o
            deslocamento
1  c ← Comprimento(S);
2  enquanto c > 0 faça
3      s ← S[c];
4      se s é a primeira célula ocupada pelo trem então
5          c ← (c − 1);
6          se (s está no estado P1) ou (s está no estado P2 e Tp(L) = Tp(T))
           ou (s está no estado P3) então
7              se S[c + 1] até S[c + ΔL + V(T)] estão no estado P0 então
8                  Move o trem em s para S[c + V(T)]; c ← c − C(T);
9              senão
10                 Muda o estado de s até S[c − C(T)] para P3;
```

Algoritmo 1: Algoritmo de atualização do array que representa um segmento.

- ArriveOn(Station S): Evento disparado quando o usuário chega a uma estação a partir de um veículo, por exemplo, quando o trem estaciona. A partir da ação desejada para a estação, dentre as três mencionadas, o usuário executa uma ação, que pode ser desembarcar do trem, ir para uma transferência, ou seguir adiante;
- EnterOn(Station S): Evento disparado quando o usuário entra em uma estação. Este evento serve para decidir o que o usuário irá fazer a seguir, de acordo com o sentido para onde deseja se deslocar;
- ExitFrom(Station S): Evento disparado quando o usuário deixa uma estação, por exemplo, saindo da simulação ou entrando em uma transferência;
- EnterOn(Train T): Disparado quando o usuário entra em um trem;
- ExitFrom(Train T): Disparado quando o usuário sai de um trem.

Observe que o tratamento de um evento pode, inclusive disparar outros eventos. Por exemplo, a chegada em uma estação - ArriveOn(R) - dependendo da ação planejada pelo usuário, pode disparar o evento ExitFrom(T), no caso, a saída de um veículo.

A ligação dos usuários e filas com o modelo autômato celular ocorre quando o trem (T) estaciona na estação (R), ou seja, o autômato celular entra no estado P2. Isto provoca duas ações: a) O evento arriveOn(R) é disparado para todos os usuários embarcados, ou seja, todos os usuários que estão no container Uc(T) e b) O evento trainArrived(T) é disparado para o segmento, que, por sua vez dispara o evento enterOn(T) para os A primeiros passageiros da fila de embarque daquele segmento, para que os mesmos embarquem no trem (sendo alocados então em Uc).

A transferência entre linhas (Connection) é tratada como duas filas, uma representando o sentido Estação A− >Estação B e outra representando o sentido contrário. Cada qual recebe tuplas com o usuário e um inteiro representando o número de ciclos que o usuário se encontra na fila. A cada ciclo, a fila é varrida e o número de ciclos é acrescido em um, até que, quando o contador atinge o tempo configurado para a instância de Connection que representa esta transferência, é acionado o evento EnterOn(S), onde S é a estação de destino da conexão, onde o usuário toma a decisão de sair do sistema ou

embarcar em algum dos sentidos, na linha conectada.

5.2. Alimentação e Operação do Modelo

A alimentação do modelo com usuários é feita a cada ciclo, em cada uma das estações, de acordo com a distribuição prevista entre Estação Origem X Estação Destino no tempo, definida na construção do modelo. Então a cada ciclo, X(s) usuários são instanciados, seu caminho é calculado (a partir de uma adaptação do algoritmo de *Dijkstra* para o grafo utilizado no modelo), e então o evento EnterOn(s) é invocado, representando a entrada do usuário na estação a partir do ambiente externo.

6. Discussão sobre o Modelo Apresentado

O modelo apresentado visa a prover dois tipos de simulações possíveis, dependendo da configuração feita: Microssimulações, considerando individualmente os usuários dentro do sistema, e mesossimulações, tratando os usuários por grupos com objetivos (origem, destino e tempo de entrada no sistema) iguais. A opção pela segunda configuração deve ser levada em conta, uma vez que considerando o volume de usuários atendidos (cerca de 3,7 milhões de usuários por dia de operação, segundo [Metro 2007]), o custo computacional do modelo é grande e, portanto, o tratamento por grupos torna-se pertinente, dependendo da análise a ser feita sobre os resultados.

Sobre o aspecto multimodal, apesar de a implementação apresentada se centrar apenas em um meio, transporte sobre trilhos, é importante ressaltar que cada uma das linhas é tratada como um ambiente independente, com suas particularidades de funcionamento, como por exemplo número de trens, tempo de parada, velocidade, estações e, cuja ligação ocorre apenas pelas transferências entre linhas, geridas por um mecanismo independente. Da mesma forma, em implementações de outros modais, usando outros modelos, como sistemas de BRT (Bus Rapid Transit), ou veículos automotores individuais, contanto que tais modelos utilizem o framework para representação de suas estruturas e conexões, permitindo, ainda, que modelos distintos de simulação sejam utilizados, desde que a sincronização de tempo e respectivas conexões seja utilizada.

Este aspecto multimodal é atingido, levando em consideração a estrutura proposta em [Rodrigue et al. 2007], onde, no framework, os nós são representados pelas entidades que implementam *IStructure*, e, por conseguinte, podem, não apenas representar uma estação de trem apenas, mas também uma determinada região da cidade em questão.

A representação dos usuários (individualmente ou como grupos) utilizando-se agentes inteligentes é especialmente útil para simular quais os potenciais benefícios de se ter usuários melhor informados, isto é, dados os estados de lotação de algumas linhas em horários específicos, se o usuário tiver a ciência de que caminhos alternativos serão mais eficientes e utilizar estes caminhos, será possível simular quais serão os benefícios para todo o sistema.

A análise dos resultados da simulação é feita a partir dos logs gerados pelo framework e pela implementação. Sendo eles:

- Número de usuários na fila, por estação, por ciclo;
- Tempo de espera, em fila, pelos usuários;
- Número de usuários, por trem, por ciclo;

- Tempos de deslocamento previsto e tempo observado, por usuário;
- Número de usuários em transferência entre linhas.

A qualidade da simulação depende diretamente da qualidade dos dados de entrada informados, sendo uma boa fonte, por exemplo, coletas feitas nos moldes da pesquisa origem-destino ([Metro 2007]), feita esporadicamente para identificação de demanda da linha.

As análises possíveis vão desde efeitos de alterações operacionais como aumento do número de trens ou tempo de parada, melhorias em túneis de transferência, ou, até mesmo os efeitos do acréscimo de novas linhas sobre todo o sistema, desde de seja observada na parametrização o aumento da demanda sobre o sistema causada pela maior cobertura, tanto de novos usuários em linhas já existentes (que utilizam outra forma de deslocamento), quanto das novas estações adicionadas.

7. Considerações Finais

A construção deste modelo é um trabalho em andamento. A estrutura apresentada representa uma simulação básica sobre sistemas de transporte sobre trilhos. Há uma série de complementos e melhorias a serem feitos no modelo proposto e, consequentemente, no framework. Discorreremos sobre elas a seguir.

Situações como saturação das filas ou trens, que diminuem a capacidade de absorção de usuários por essas estruturas, porém são previstas no framework e no modelo. Também um acréscimo será a possibilidade de causar intervenções no modelo (por exemplo, interrupção de deslocamento, ou redução de velocidade) durante a execução, de forma a representar situações comuns, por exemplo, na linha Vermelha paulista que, por possuir trechos de superfície, quando há chuva nestes trechos, a velocidade é reduzida.

Atrelado a esta funcionalidade, outro implemento a ser feito é a criação de perfis de comportamento para os usuários, onde a definição de rota a ser feita é calculada de acordo com o perfil (levando em consideração conforto, por exemplo) e calculada quando houver mudanças de situação na linha, mesmo que o usuário já esteja percorrendo o trajeto.

Ainda, outra evolução a ser feita é a utilização do framework para a construção de modelos para outros modais, interligando-os ao modelo de transporte sobre trilhos apresentado, de forma a permitir que se trabalhe com múltiplos modais de transporte como uma única grande simulação, permitindo estudos como por exemplo a relação entre o sistema de ônibus e o sistema de trens de uma cidade.

Referências

Barceló, J., editor (2010). *Fundamentals of Traffic Simulation*. Springer.

Chen, M., Bärwolff, G., and Schwandt, H. (2010). A study of step calculations in traffic cellular automaton models. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 747–752.

Gao, L., Liu, Z., Xu, Q., and Feng, X. (2009). A delay model of pedestrian-vehicle system on two crossings. In *Advanced Forum on Transportation of China (AFTC 2009)*, pages 192–198.

Huixin, W. and Wenhong, W. (2010). Microscopic dynamic simulation model for pedestrian-vehicle mixed traffic. In *International Conference on E-Health Networking, Digital Ecosystems and Technologies (EDT)*.

Jin, X. and White, R. (2012). An agent-based model of the influence of neighbourhood design on daily trip patterns. *Computers, Environment and Urban Systems*, 36(5):398–411.

Klügl, F. and Rindsfüser, G. (2007). Large-scale agent-based pedestrian simulation. In *Proceedings of the 5th German conference on Multiagent System Technologies*, MATES '07, pages 145–156, Berlin, Heidelberg. Springer-Verlag.

Kretz, T., GroBe, A., Hengst, S., Kautzsch, L., Pohlmann, A., and Vortisch, P. (2011). Quickest paths in simulations of pedestrians. *Advances in Complex Systems*, 14(5):733–759.

Ma, J., Lo, S., Xu, X., and Song, W. (2011). Dynamic features of pedestrian-vehicle counter flow conflicts. *ICTE 2011 - Proceedings of the 3rd International Conference on Transportation Engineering*, pages 697–702.

Ma, W., Ma, W., and Yang, X. (2008). Development and evaluation of a fuzzy logic control approach for pedestrian crossing. In *Proceedings of the 2008 International Conference on Intelligent Computation Technology and Automation - Volume 01*, pages 882–886, Washington, DC, USA. IEEE Computer Society.

Ma, W. and Yang, X. (2009). Signal coordination models for midblock pedestrian crossing and adjacent intersections. In *Proceedings of the 2009 Second International Conference on Intelligent Computation Technology and Automation - Volume 02*, ICICTA '09, pages 193–196, Washington, DC, USA. IEEE Computer Society.

Metro (2007). Companhia do Metropolitano de São Paulo Pesquisa Origem e Destino.

Metro (2014). Site da Companhia do Metropolitano de São Paulo - Metrô.

Morais, D. and Digiampietri, L. (2012). Uma revisão sistemática acerca de métodos de simulação envolvendo múltiplos meios de transporte. *Revista de Sistemas de Informação da FSMA*, 10.

Nagel, K. (2002). Cellular automata models for transportation applications. In *Proceedings of the 5th International Conference on Cellular Automata for Research and Industry*, ACRI '01, pages 20–31, London, UK, UK. Springer-Verlag.

Pan, M., Dong, S., Sun, J., and Li, K. (2010). Microscopic simulation research on signal cycle length of mixed traffic considering violation. In *Proceedings of the 2010 International Conference on Intelligent Computation Technology and Automation - Volume 02*, ICICTA '10, pages 674–678, Washington, DC, USA. IEEE Computer Society.

Rodrigue, J.-P., Comtois, C., and Slack, B. (2007). *The Geography of Transport System*. Routledge.

Usher, J., Liu, X., and Kolstad, E. (2010). Simulation of pedestrian behavior in intermodal facilities. In *Proceedings of the 2010 Spring Simulation Multiconference*.

Wang, L., Mao, B., Chen, S., and Zhang, K. (2009). Mixed flow simulation at urban intersections: Computational comparisons between conflict-point detection and cellular automata models. In *Proceedings of the 2009 International Joint Conference*

on Computational Sciences and Optimization - Volume 02, CSO '09, pages 100–104, Washington, DC, USA. IEEE Computer Society.

Wooldridge, M. (2006). *An introduction to Multiagent Systems*. John Wiley & Sons ltd.

Xiaobei, J., Hui, X., and Hongwei, G. (2009). Analysis of crowd behavior in route choice based on dynamic programming. In *Proceedings of the 9th International Conference of Chinese Transportation Professionals, ICCTP 2009: Critical Issues in Transportation System Planning, Development, and Management*, volume 358, pages 742–750.

Xue, Y., Tian, H.-H., He, H.-D., Lu, W.-Z., and Wei, Y.-F. (2009). Exploring jamming transitions and density waves in bidirectional pedestrian traffic. *The European Physical Journal B - Condensed Matter and Complex Systems*, 69:289–295.