

UMA ABORDAGEM HEURÍSTICA PARA O PROBLEMA DE PROGRAMAÇÃO DE TRENS EM LINHAS SINGELAS

Jose Eugenio Leal
Adriana Costa Soares

Luciana Silveira Netto Nunes

Pontifícia Universidade Católica do Rio de Janeiro
Departamento de Engenharia Industrial

RESUMO

O objetivo deste trabalho é desenvolver um procedimento heurístico para resolução do problema de programação de trens em vias singelas. O procedimento heurístico proposto servirá como uma ferramenta de apoio à decisão ao centro de controle operacional no gerenciamento do tráfego ferroviário na medida que fornecerá ao controlador uma ferramenta que o auxilie no seqüenciamento dos trens em tempo real. Este modelo pode ajudar o controlador a resolver os conflitos entre os trens (ultrapassagem ou cruzamentos) em estações de maneira a cumprir algum objetivo pré-definido. Atualmente, os conflitos são resolvidos em um gráfico espaço - tempo utilizando a experiência dos controladores e o conhecimento de algumas condições da via. A utilização do modelo, em uma aplicação real, mostrou que utilizando tanto a técnica de *Branch and Bound* quanto a técnica de descida em vários níveis, o modelo é capaz de produzir, a princípio, resultados satisfatórios para o problema.

ABSTRACT

This paper describes the development of a train scheduling model that helps the operational control center in the railroad traffic management. This model was designed to solve train schedules on single railway tracks. The model has been developed with one major application in mind, namely: as a decision support tool for train dispatchers to schedule trains in real time. It can help the dispatcher to resolve the train conflicts (overtakings as well as crossings) at the sidings in a way to achieve a pre-defined objective. The use of the model in an application to a "real life" problem has shown that the heuristic model, with the Branch and Bound technique as well as partial breadth-first tree search in many levels, is able to produce useful results for the problem.

1. INTRODUÇÃO

Um dos maiores desafios no controle de tráfego ferroviário em linhas singelas é a resolução dos conflitos entre trens. Na ferrovia, os movimentos dos trens são monitorados através de painéis com o movimento em tempo real dos trens, baseado em uma programação inicial pré-estabelecida. Nesta programação, a partir do tempo de partida de cada trem, são projetadas as chegadas ao destino final, sem considerar os cruzamentos e ultrapassagens. Desta forma, qualquer perturbação no tráfego ferroviário, seja por questões técnicas ou erros humanos, podem causar uma instabilidade no sistema e resultar em atrasos na programação, diminuindo o nível de serviço ao cliente. O nível de serviço na operação ferroviária pode ser medido através do tempo total de viagem e a pontualidade na chegada dos trens.

Segundo Sahin (1998), a resolução destes conflitos é um problema combinatório NP-completo com complexidade computacional $O(2^n)$, onde n é o número de conflito. Isto significa que o tempo computacional e a memória requerida para resolver este problema crescem exponencialmente com o tamanho do problema. O tamanho do problema depende do número de cruzamentos e ultrapassagens (número de conflitos), que dependem do tamanho da linha, intensidade do tráfego, velocidade dos trens e suas diferenças e o horizonte de tempo considerado. Assim, apenas problemas com instancias pequenas podem ser resolvidas otimamente.

Atualmente, o processo de resolução de conflitos é determinado pela experiência profissional dos controladores de tráfego, o que não necessariamente fornece uma solução ótima para o problema, sob o ponto de vista da eficiência operacional. Estas decisões são tomadas na medida que os eventos ocorrem durante o dia, o que não permite intervenções na programação final e o fornecimento de informações precisas para o cliente sobre o horário de chegada da carga no seu destino final. A solução gerada é apresentada em um gráfico espaço – tempo, construído conforme os eventos acontecem.

As resoluções dos conflitos entre trens podem resultar em ultrapassagens, atrasos ou cruzamentos. As soluções se tornam complexas, por envolverem um grande número de variáveis que dificilmente podem ser analisadas pelo raciocínio humano normal. Outra questão importante é a prioridade atribuída aos trens. Isto motiva o emprego de um método computacional, que não seja subjetivo, ou seja, busque uma solução próxima da ótima em um espaço curto de tempo e, em tempo real.

O modelo proposto neste artigo desenvolve uma heurística para a resolução destes conflitos de trens em tempo polinomial, fornecendo uma programação viável que minimize o tempo total de percurso dos trens. A solução destes conflitos é apresentada em um gráfico espaço- tempo, que servirá como uma ferramenta de auxílio aos controladores, na medida que antecipa para o usuário toda a programação ao longo do dia. A análise do gráfico, em conjunto com as habilidades e conhecimentos do controlador, podem resultar na diminuição do tempo de chegada no destino final, já que se podem tomar decisões “antecipadas” baseadas na saída gráfica resultante.

2. IDENTIFICAÇÃO DOS CONFLITOS

Os conflitos ocorrem nas operações ferroviárias em linhas singelas. Esta é uma linha única que liga dois pátios adjacentes. É utilizada para circulação de trens nos dois sentidos. A circulação dos trens é representada através de um gráfico espaço-tempo, onde no eixo das ordenadas estão localizadas as estações e no eixo das abscissas, o tempo (de 0 a 24 horas). Este gráfico é também utilizado para visualizar a necessidade de cruzamentos e ultrapassagens.

O conflito acontece quando dois trens estão prestes a ocupar o mesmo trecho ao mesmo tempo. Esta situação pode ocorrer de duas maneiras distintas: cruzamentos (para trens em sentidos opostos) e ultrapassagens (para trens no mesmo sentido). As figuras 1 e 2 ilustram estas duas situações, respectivamente.

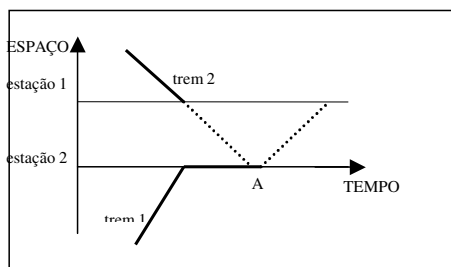


Figura 1: Trens viajando em sentido contrário (cruzamento)

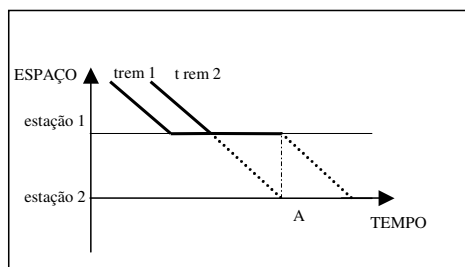


Figura 2: Trens viajando no mesmo sentido (ultrapassagem)

A partir da consideração dos conflitos entre trens, serão escolhidos os pontos de cruzamentos e ultrapassagem e qual dos dois trens envolvidos no conflito deverá parar. Estas escolhas fazem parte do conjunto de decisões rotineiras da operação ferroviária.

A figura 1 caracteriza um conflito entre os trens 1 e 2 viajando em sentidos opostos. Observa-se da figura que não é possível que o trem 1 e o trem 2 sigam viagem ao mesmo tempo, pois só há uma via (via singela). Assim, somente um trem poderá seguir viagem. No caso do trem 2 seguir viagem, o trem 1 ficará parado até o tempo A, momento em que o trem 2 chega ao final do trecho. Somente a partir desse momento o trem 1 poderá seguir viagem. Este é um caso de cruzamento entre trens.

A figura 2 trata de um caso de ultrapassagem entre trens. O trem 1 deverá esperar até o tempo A para que possa seguir viagem. Este será ultrapassado pelo trem 2.

3. PRINCIPAIS MODELOS PARA A RESOLUÇÃO DE CONFLITOS ENTRE TRENS

O problema de controle de tráfego ferroviário é um problema bastante abordado na literatura. Diferentes autores buscaram formas de resolução deste problema, desenvolvendo modelos matemáticos e procedimentos de solução. De acordo com Sahin (1998), as técnicas mais utilizadas para resolução deste problema podem ser divididas em três grupos: simulação, programação matemática e sistemas inteligentes.

Petersen e Taylor (1982) apresentam um modelo de simulação para a ferrovia de uso geral, ajudando na programação de um conjunto de trens, em tempo real, com velocidades e prioridades distintas. O modelo é aplicado em linha singela e linhas múltiplas e descreve a movimentação de trens.

Ferreira (1992), apresenta alguns procedimentos operacionais que apóiam os despachadores no gerenciamento e controle do tráfego ferroviário. O autor descreve uma técnica para o cruzamento, em movimento, de dois trens em via singela. Também realiza um estudo sobre os atrasos dos trens. Supõe-se um quadro de horários rígidos, contendo as partidas de um número de trens nas sucessivas estações. Se um trem atrasa em uma estação ou entre duas estações, este atraso poderá causar efeitos nas chegadas e partidas em outras estações.

Carey (1994) apresenta um modelo para determinar o movimento de trens, onde consideram linhas múltiplas e trens de diferentes velocidades. Todos os trens possuem paradas padrões. O objetivo é minimizar atrasos ou custos e encontrar as demandas de viagem. O modelo resolve conflitos entre trens, sujeito a diversas restrições e é aplicado a uma rede mais geral, sem muita complexidade. É aplicado para o movimento de trens de passageiros e, foi utilizada programação matemática inteira.

Szpigel (1972) apresenta um modelo de programação linear fornecendo uma solução otimizada relativa aos atrasos de trens. Este modelo busca a solução dos conflitos, levando em conta as prioridades dos trens e tempos de viagem. A solução ótima será aquela que apresentar menor valor para a soma ponderada dos tempos de viagem de todos os trens. Para o autor, o problema de seqüenciamento de trens é visto como um problema de *job-shop scheduling*, onde tarefas constituídas de operações devem ser executadas por um conjunto de máquinas. Neste modelo foram utilizados os termos “viagem de trem” e “trechos” no lugar de “tarefas” e “máquinas”. Embora antigo, o trabalho de Szpigel é citado em quase todos os trabalhos que tratam deste assunto. (Sedrez (1983), Higgins (1996), Sahin (1998), etc)

No trabalho de Sedrez (1983) o autor avaliou os possíveis impactos provocados pela circulação de trens especiais. Estes trens são trens não regulares os quais são inseridos na malha, sem que se modifique os horários de partida e tempos de percurso dos trens normais. O autor utilizou a mesma abordagem de Szpigel (1972), onde foram inseridas as seguintes modificações: a criação de um horário por estações e a criação de um gráfico de trens.

Higgins et al (1996) analisa diferentes formas de abordagem do problema, fornecendo resultados para a resolução exata do problema, através da técnica de *Branch and Bound* e aplica procedimentos heurísticos de solução, como Algoritmos Genéticos e Busca Tabu. O objetivo final é a minimização do tempo total de viagem ponderado pelas prioridades de cada trem. No modelo proposto, os autores consideram as velocidades diferentes de cada trem, a velocidade máxima de cada trecho, distância mínima entre trens e a prioridade dos trens.

O modelo de Rosseto (1997), é um modelo computacional que busca a otimização de solução de conflitos de trens, sendo baseado em uma heurística e implementado como parte de um modelo de simulação para a estrada de ferro de Carajás. A heurística proposta busca alocar as ocupações nos trechos disponíveis da via, levando em consideração uma ordem de prioridades, visando otimizar o tempo total de paradas. Os conflitos são solucionados em ordem cronológica. Desta forma, de uma só vez é solucionado um conjunto de conflitos. A simulação parte dos tempos de percurso e horários de partida previstos.

4. O MODELO PROPOSTO

O modelo propõe uma heurística para resolver conflitos entre trens, utilizando uma nova abordagem para as formulações de Szpigel (1972). Parte-se inicialmente de uma grade de horário de trens, onde constam todos os horários de partida previstos, origem e destino de cada trem. Os tempos de percurso também são conhecidos.

4.1 Notação e convenções

Um trecho completo é composto de vários sub-trechos, delimitados por estações. O trecho completo tem n_{est} estações, incluindo as estações inicial e final do trecho. Como o trecho pode ser percorrido em dois sentidos, convencionou-se, neste trabalho, que um sentido de viagem com quilometragem crescente será definido como positivo e o sentido contrário, como negativo. Assim os trens, segundo o sentido do percurso, terão um índice positivo ou negativo. A variável $indtrem_i$ guarda o índice do trem i assumindo o valor +1, para viagens no sentido positivo e -1, no sentido contrário.

A programação inicial originada da grade de horário de trens possui $ntrens$ circulando na linha. Estes trens apresentam diversos conflitos. O objetivo da heurística é realizar uma nova programação livre de conflitos. Cada trem i possui uma estação de início e uma estação de fim de percurso, expressas pelas variáveis ini_i e fim_i .

A viagem do trem estará definida pelo momento de entrada em cada trecho entre duas estações, e pelo momento de chegada à estação final do trecho. Para cada trem i , o momento de entrada no trecho à frente da estação j , estará guardado na variável $tvia_{i,j}$. Cada trem i , pode ter uma parada programada na estação j , para realizar uma tarefa operacional. O tempo desta parada estará guardado na variável $tpar_{i,j}$. Observa-se que esta definição de viagem, trata da viagem programada do trem, na situação livre de conflitos. A programação definitiva de cada trem será o resultado do procedimento.

Cada trem i pode ter associado a ele uma prioridade $prior_i$, que fornece um peso relativo para o trem indicando a sua importância e grau de prioridade, no tratamento da solução de conflitos com outros trens. Inicialmente, para efeito de entendimento do procedimento, todos os trens terão igual prioridade.

Todas estas variáveis definidas acima são dados de entrada para o procedimento. Uma variável necessária para resolver os conflitos é o tempo de ocupação do trem i no trecho j , calculado a partir dos dados de entrada e guardado na variável $ocupa_{i,j}$, que guarda o tempo de ocupação do trem i , no trecho à frente da estação j .

Durante o procedimento, os conflitos ao serem resolvidos, podem ocasionar atrasos no momento da entrada do trem i no trecho à frente da estação j . Este valor será guardado na variável $atraso_{i,j}$.

Com relação a matriz de tempos $tvia$, é importante esclarecer o tipo de dado fornecido na matriz para evitar interpretações equivocadas, que conduzem eventualmente a erros. A viagem dos trens na matriz $tvia$ fornece os horários de passagem pelas estações de entrada em cada trecho, inclusive na estação final.

4.2 Identificação do conflito - Ordem de tratamento dos conflitos.

Um conflito se dá quando um trem tenta ocupar um trecho, durante a ocupação deste mesmo trecho, por outro trem.

Para trens de mesmo sentido, uma vez que um trem de referência se encontra no trecho, um outro trem, anterior a ele que saísse do trecho, ou um trem, posterior a ele, que entrasse no trecho

durante a sua ocupação estaria em conflito. Para trens de sentidos opostos, a seguinte situação define o conflito. Uma vez que um trem de referência se encontra no trecho, um trem no sentido oposto que entre, ou saia do trecho durante a sua ocupação, estaria em conflito. Estas situações foram ilustradas anteriormente nas figuras 1 e 2.

No caso de conflitos de trens de mesmo sentido, a contagem de trechos é direta e cada estação de entrada *icol* do trem *i* de referência corresponde à mesma estação *icol* do trem *j*, potencial conflitante.

No caso de trens de sentido contrário, as estações de entrada de trecho não têm a mesma correspondência. Para um trem de referência *i* de sentido positivo, entrando em *icol*, a estação do trem de sentido negativo e potencial conflitante entrando no mesmo trecho é *icol* + 1. No caso de um trem *i* de sentido negativo, entrando em *icol*, o trem de sentido positivo, conflitando com ele, vai entrar no trecho na estação *icol*-1.

Generalizando, para trens de sentidos contrários, o trecho de entrada do trem *j*, é em geral:

$$icol - indtrem_j \quad (1)$$

O tipo de conflito, se os trens são de mesmo sentido, ou de sentido oposto pode ser detectado por:

$$\clubsuit \text{ Se } indtrem_i \times indtrem_j = 1, \text{ se os trens são de mesmo sentido} \quad (2)$$

$$\clubsuit \text{ Se } indtrem_i \times indtrem_j = -1, \text{ se são de sentido contrário.} \quad (3)$$

A estação de entrada do trem *j*, pode ser definida como *jcol*. Assim, temos as seguintes relações entre o trem de referência *i* e o potencial conflitante *j*:

$$jcol = icol, \text{ para trens de mesmo sentido e} \quad (4)$$

$$jcol = icol - indtrem_j, \text{ para trens de sentido contrário} \quad (5)$$

O conflito existe se:

$$(tvia_{ilin,icol} < tvia_{j,jcol} < tvia_{ilin,icol} + ocupa_{ilin,icol}) \quad (6)$$

Ou se:

$$(tvia_{ilin,icol} < tvia_{j,jcol} + ocupa_{j,jcol} < tvia_{ilin,icol} + ocupa_{ilin,icol}) \quad (7)$$

4.3 A solução do conflito.

O conflito se resolve atrasando um dos trens. O atraso pode ser decidido tanto para o trem *ilin* (trem de referência), como para o trem *j*. Quando os trens têm a mesma prioridade, têm-se dois casos:

a) Fazer o trem j entrar no trecho depois da saída de $ilin$, e portanto:

$$atraso_{j,jcol} = tvia_{ilin, icol} + ocupa_{ilin,icol} - tvia_{j,jcol} \quad (7)$$

b) Fazer o trem i entrar no trecho depois do trem j sair:

$$atraso_{ilin,icol} = tvia_{j,jcol} + ocupa_{j,jcol} - tvia_{ilin,icol} . \quad (8)$$

4.4 A seleção do próximo conflito

O próximo conflito a ser tratado pode ser encontrado fazendo uma varredura no espaço, desde uma estação inicial da linha, ou fazendo uma varredura no tempo, a partir do momento mais cedo de operação na linha. O primeiro conflito encontrado segundo um dos critérios, é o selecionado para ser tratado em primeiro lugar.

Diversos autores (Higgins (1996), Szpigel (1972), entre outros) indicam a conveniência de selecionar o conflito pelo tempo. Isto porque, tratar primeiro o conflito mais cedo, implica em propagar para frente, no tempo os efeitos da solução deste conflito. No caso do tratamento pelo espaço, esta solução pode afetar e criar conflitos em outros trens, em momento mais cedo que o atual, criando um *loop* complexo de soluções.

No modelo proposto, utiliza-se o critério de varredura no tempo para a seleção do próximo conflito a ser tratado.

4.5. Um procedimento de solução.

4.5.1. A função objetiva

O problema consiste em definir uma programação para os trens, resolvendo os conflitos, considerando as prioridades dos trens de forma a minimizar uma função objetiva. A função mais usada (Spiegel, 1972) tem sido a de minimizar os atrasos ponderados pela prioridade dos trens. Tanto se pode expressar esta função tomando os atrasos diretamente, como tomando os tempos de chegada dos trens, ponderados pelas prioridades, já que esta última função apenas agrega uma constante à primeira.

Então, a função objetiva pode ser:

$$Min F = \sum_{i=1}^{ntren} tvia_{i, fim_i} * prior_i \quad (8)$$

O procedimento pode ser então resumido nos seguintes passos:

Passo 0: No início a solução atual é a solução, considerando a programação inicial, ignorando os conflitos.

Enquanto não chegar a uma boa solução faça (1):

Passo 1. Selecione uma solução atual.

Passo 2. Enquanto houver conflitos faça (2):

1. Identifique um conflito.
2. Identifique o trem de referência, seu sentido e o trecho de referência.

3. Identifique o trem de conflito e seu sentido de viagem.
4. Calcule o atraso para o trem de conflito.
5. Calcule o atraso para o trem de referência.
6. Escolha o menor atraso.
7. Refaça os horários do trem atrasado da estação atual até a estação final da viagem.

Fim faça 2.

Passo 3. Calcule o valor da função objetiva.

Passo 4. Compare com outras soluções.

Fim Faça 1.

4.6. A propagação ou absorção de atrasos.

Os atrasos nas estações não vão, obrigatoriamente, se somando e se propagando para as estações seguintes. Na verdade, atrasos sofridos por um trem em uma estação, com valores menores que atrasos sofridos em uma estação a frente vão ser absorvidos pelo atraso à frente. Desta forma, ao haver um atraso devido a um conflito, a sua propagação para frente vai depender da comparação do valor do atraso com o atraso atual nas estações à frente, no percurso do trem.

A propagação de atrasos vai ser calculada sucessivamente, para cada estação à frente da tratada recentemente. Assim, o atraso de um trem i em uma estação j , à frente da estação recentemente tratada, vai ser dado por:

$$\text{Atraso}_{i,j} = \max(\text{atraso}_{i,j}; \text{atraso}_{i,j-1}) \quad (9)$$

5. APLICAÇÃO DO MODELO PROPOSTO

O modelo proposto neste artigo foi implementado em uma empresa de transporte de cargas da região Sudeste, na malha São Paulo. Os dados iniciais do problema foram retirados do gráfico de trens preenchido pelo controlador de tráfego (horário de partida e chegada) e estão descritos a seguir:

- ♣ Número de trens = 25 (13 trens descendo, +1, e 12 trens subindo, -1)
- ♣ Número de estações = 25
- ♣ Prioridades dos trens = 1

A grade inicial de programação dos trens está representada em um gráfico espaço tempo, onde é possível identificar os conflitos presentes na linha. As paradas programadas para troca de equipe e manutenção dos trens também estão representadas no gráfico (representadas como linhas horizontais). A figura 3 ilustra a situação inicial.

O procedimento descrito no item 4 foi implementado em um programa de computador. Os resultados são utilizados como dados de entrada para o sistema de apoio à decisão, apresentado por Nunes (2004). A saída do sistema é um gráfico automático (espaço – tempo) que representa a programação sem conflitos, conforme a figura 4.

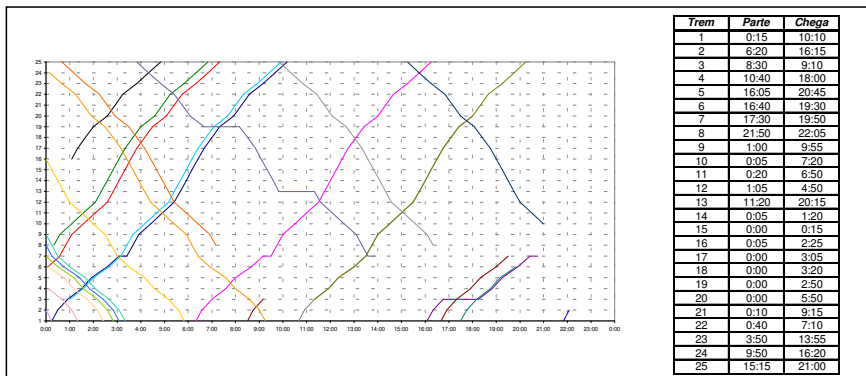


Figura 3: Situação inicial, com conflitos

Para resolver os conflitos presentes na grade foi adotado o procedimento descrito neste artigo, utilizando duas técnicas computacionais distintas. Em um primeiro teste foi utilizado o procedimento heurístico descrito na seção 4.5, com uma redução do espaço de busca de soluções através da técnica de *Branch & Bound* sugerida por Higgins (1996). Em uma segunda tentativa, foi adotado um procedimento de descida de vários níveis na árvore binária de solução, onde foram guardadas em um vetor as melhores soluções dos níveis. A cada descida é escolhida a melhor solução e o procedimento continua. Os gráficos resultantes estão ilustrados nas figuras 4 e 5 respectivamente.

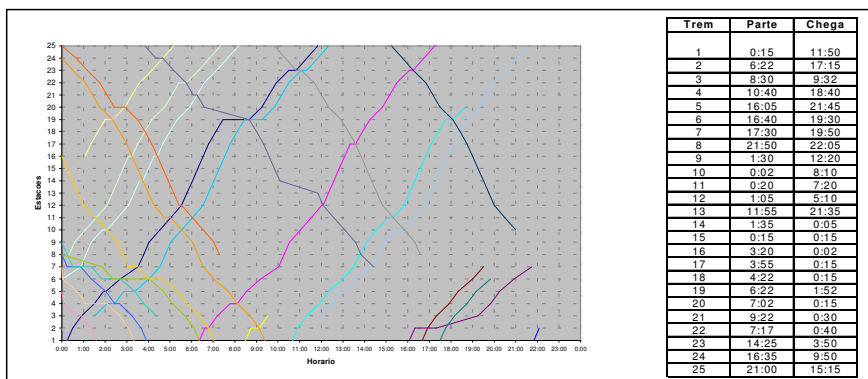


Figura 4: Solução dos conflitos entre trens, com B&B

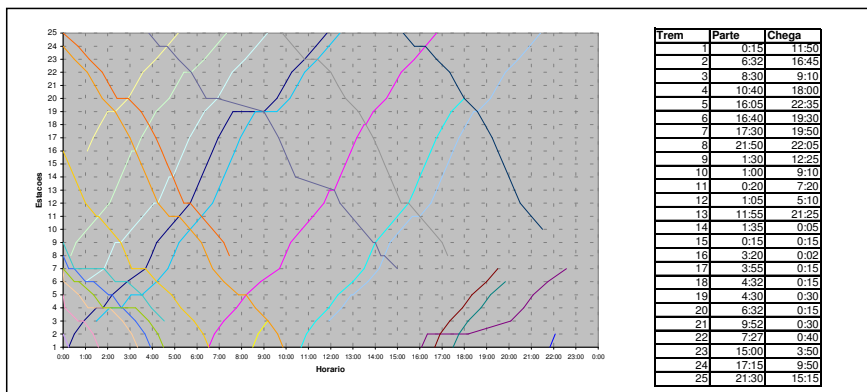


Figura 5: Solução dos conflitos entre trens, com 4 níveis

A soma dos tempos totais de percurso de todos os trens encontrados utilizando o procedimento descrito neste artigo e as técnicas apresentadas anteriormente foram, respectivamente, 17.430 minutos (*branch and bound*) e 17.458 minutos (descida em níveis). Estes resultados apontam método de *Branch and Bound* como uma estratégia de resolução que produz melhores resultados.

Para comparar a qualidade dos resultados encontrados, foram feitas análises do gráfico efetivamente realizado pela ferrovia e a solução gerada pelo procedimento proposto. A figura 6 ilustra a solução real adotada pela ferrovia para os mesmos trens nos mesmos trechos. Este gráfico foi construído em tempo real pelos controladores, na medida que os eventos foram acontecendo ao longo do dia.

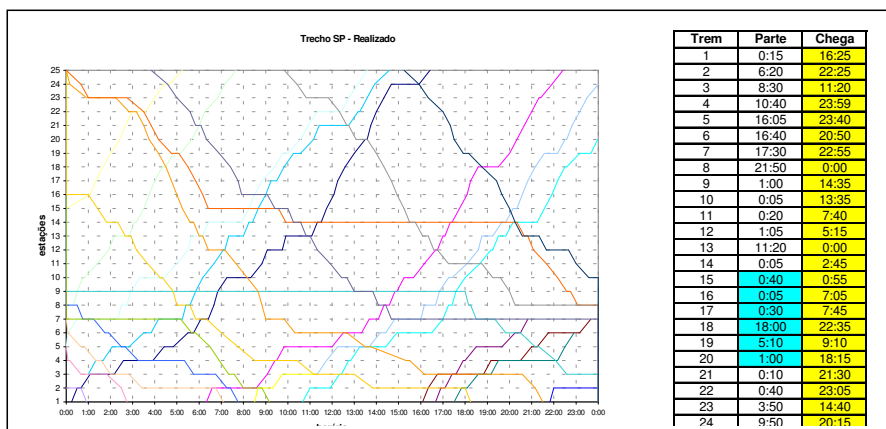


Figura 6: Solução real implementada pela ferrovia

O tempo total de percurso dos trens para a situação ilustrada na figura 7 foi de 23.984 minutos, 6.507 minutos a mais do que a melhor solução encontrada pelo procedimento proposto.

Comparando os resultados apresentados na figura 6 com a solução inicial apresentada na figura 3, pode-se observar que houve um atraso na partida de 7 trens (9,10,15,16,19,20 e 21) e atrasos na chegada de 19 trens. Ou seja, somente 6 trens cumpriram a programação inicial (3,4,6,7,8 e 15). Ao se comparar este resultado com a solução implementada pela ferrovia, pode-se observar que nenhum trem cumpriu a programação inicial e praticamente todos os trens chegaram depois do tempo sugerido pelo procedimento proposto neste artigo. Com relação à partida dos trens, a solução realizada pela ferrovia de atrasar os trens 15,16,17,18 e 19 provocou um atraso global superior ao atraso dos trens 9,10,15,16,19,20 e 21 sugerido pelo procedimento. Isto reitera a hipótese de que a solução baseada somente na experiência do controlador não conduz a resultados muito eficientes.

6. CONSIDERAÇÕES FINAIS

O modelo proposto busca preencher uma lacuna existente no controle e despacho de trens, oferecendo ao controlador de tráfego uma ferramenta de auxílio na resolução de conflitos e, conseqüentemente, na programação de trens. Os controladores de tráfego coordenam a circulação de trens e veículos de manutenção da via, controlam e programam horários de circulação de trens, além de administrar a estação e controlar atividades de pátios e terminais. O problema de programação de horários é, portanto, bastante complexo e a atual forma empírica de resolver o problema, não permite, em geral, soluções de boa qualidade. Todas estas atividades rotineiras envolvem, também, uma grande carga de atenção e stress, o que acaba provocando erros ou falhas na programação dos trens. A ferramenta desenvolvida visa minimizar estas falhas e, em conjunto com a experiência dos controladores, fornecer melhores resultados para os tempos de percurso.

A maior contribuição deste trabalho, portanto, está em apresentar um procedimento heurístico simples para a resolução dos conflitos entre os trens na programação de horários. Este processo nas ferrovias está intimamente relacionado com o nível de serviço prestado aos clientes. São as paradas não programadas na grade de horário de trens que provocam atrasos na chegada do trem ao seu destino final. Atualmente, a programação de trens é um processo manual e o gráfico de trens é construído em tempo real não sendo possível prever nenhuma parada não programada até que ela aconteça de fato. Os controladores utilizam a sua experiência para decidir quais serão as paradas não programadas na grade de trens, não existindo nenhuma ferramenta computacional disponível no centro de controle operacional das empresas para auxiliar neste despacho dos trens.

Este modelo deverá ser integrado aos sistemas informatizados do Centro de Controle Operacional das ferrovias, o que irá fornecer ao controlador uma visão geral de toda a movimentação da linha e um gráfico de trens livre de conflitos. Assim, no início do dia, já será possível prever todas as paradas provocadas por cruzamentos e/ou ultrapassagens.

A aplicação do modelo para um problema real, apresentado na seção 5, mostrou que o procedimento caminha em direção a boas soluções, fornecendo um tempo total de percurso menor do que aquele atualmente obtido pela ferrovia. A solução aplicada pela ferrovia, baseada

apenas na experiência dos controladores, teve um tempo de percurso bem superior ao sugerido pelo modelo.

Com relação às técnicas computacionais empregadas, tanto a técnica de *Branch & Bound* quanto o método de descida em vários níveis na árvore binária de soluções, produziram bons resultados apontando, inicialmente, o método de *Branch and Bound* como a estratégia de solução que produz melhores resultados.

Agradecimentos.

Os autores agradecem ao CNPq pela concessão de bolsas de estudo e pesquisa e apoio financeiro ao projeto com recursos do Fundo de Pesquisa em Transportes.

REFERÊNCIAS BIBLIOGRÁFICAS

- Carey, M. (1994) A model and strategy for train pathing with choice of lines, platforms and routes. *Transportation Research*. Vol 28 b, número 5, pp. 333-353.
- Ferreira, J. G. (1992) *Procedimentos operacionais para o gerenciamento e controle de tráfego ferroviário*. Dissertação de mestrado do Departamento de Engenharia Industrial. PUC/RJ. Brasil.
- Higgins, A; E, Kozan e L, Ferreira. (1996) Optimal scheduling of trains on a single line track. *Transportation research B*. Vol 30, N° 2, pp. 147-161
- Medeiros, S. R. (1989) *Modelo de simulação de tráfego ferroviário*. Dissertação de mestrado do Departamento de Engenharia Industrial. PUC/RJ. Brasil.
- Nunes, L. S. N. (2004) **Sistema de apoio à decisão ao centro de controle operacional no gerenciamento do tráfego ferroviário**. Dissertação de mestrado do Departamento de Engenharia Industrial. PUC/RJ. Brasil.
- Petersen, E. R. e A. J Taylor,. (1982) A Structured model for rail line simulation and optimization. *Transportation science*. Vol 16, N° 2, pp. 192-206.
- Rosseto, C. F. (1997) *Modelo para programação de despachos de trens em vias singelas*. Dissertação de mestrado da Escola Politécnica da Universidade de São Paulo. Brasil.
- Sedrez, C. A. B. (1983) **Impactos decorrentes da circulação de trens especiais**. Dissertação de mestrado do Departamento de Engenharia de Transportes. Instituto Militar de Engenharia. Rio de Janeiro. Brasil
- Sahin, I. (1998) Railway traffic control and train scheduling based on inter-train conflict management. *Transportation Research B* vol 33 pp 511-534
- Szpigel, B. (1972) *Seqüenciamento de trens*. Dissertação de mestrado do Departamento de Engenharia Elétrica PUC/RJ. Brasil

Endereço dos autores

Jose Eugenio Leal (jel@ind.puc-rio.br)
Adriana Costa Soares (soares.adriana@globo.com)
Luciana Silveira Netto Nunes (lucianasnunes@yahoo.com.br)
Departamento de Engenharia Industrial
Pontificia Universidade Catolica do Rio de Janeiro
Rua Marques de Sao Vicente 225, 9 andar L, Gávea, Rio de Janeiro.