



UMA PROPOSTA DE SOLUÇÃO PARA O PROBLEMA DA CONSTRUÇÃO DE ESCALAS DE MOTORISTAS E COBRADORES DE ÔNIBUS POR MEIO DO ALGORITMO DO *MATCHING* DE PESO MÁXIMO

Paulo Henrique Siqueira
Celso Carnieri

Maria Teresinha Arns Steiner

Departamento de Desenho, Universidade Federal do Paraná, C.P. 19081,
CEP 81531-990, Curitiba, PR,

e-mail: paulohs@ufpr.br

carnieri@mat.ufpr.br

Departamento de Matemática,
e-mail: tere@mat.ufpr.br

Ângela Olandoski Barboza

Departamento de Matemática, Centro Federal de Educação Tecnológica do Paraná,
Av. 7 de setembro, 3165,
CEP 80230-901, Curitiba, PR,

e-mail: aobarboza@yahoo.com

Recebido em 29/9/2003

Aceito em 31/6/2004

Resumo

O objetivo deste trabalho é mostrar a aplicação do Algoritmo do Matching de peso máximo, na elaboração de jornadas de trabalho para motoristas e cobradores de ônibus. Este problema deve ser resolvido levando-se em consideração o maior aproveitamento possível das tabelas de horários, com o objetivo de minimizar o número de funcionários, de horas extras e de horas ociosas. Desta forma, os custos das companhias de transporte público são minimizados. Na primeira fase do trabalho, supondo-se que as tabelas de horários já estejam divididas em escalas de curta e de longa duração, as escalas de curta duração são combinadas para a formação da jornada diária de trabalho de um funcionário. Esta combinação é feita com o Algoritmo do Matching de peso máximo, no qual as escalas são representadas por vértices de um grafo, e o peso máximo é atribuído às combinações de escalas que não formam horas extras e horas ociosas. Na segunda fase, uma jornada de final de semana é designada para cada jornada semanal de dias úteis. Por meio destas duas fases, as jornadas semanais de trabalho para motoristas e cobradores de ônibus podem ser construídas com custo mínimo. A terceira e última fase deste trabalho consiste na designação das jornadas semanais de trabalho para cada motorista e cobrador de ônibus, considerando-se suas preferências. O Algoritmo do Matching de peso máximo é utilizado para esta fase também. Este trabalho foi aplicado em três empresas de transporte público da cidade de Curitiba - PR, nas quais os algoritmos utilizados anteriormente eram heurísticos, baseados apenas na experiência do encarregado por esta tarefa.

Palavras-chave: *otimização de jornadas de trabalho, Algoritmo do Matching de peso máximo.*

1. Introdução

O objetivo deste trabalho é mostrar a aplicação do Algoritmo do *Matching* de peso máximo, na elaboração de jornadas de trabalho para motoristas e cobradores de ônibus, além de mostrar os resultados da aplicação da metodologia proposta em empresas da cidade de Curitiba - PR. Este problema deve ser resolvido, levando-se em consideração o maior aproveitamento possível das tabelas de horários de ônibus, com o objetivo de minimizar o número de funcionários, de horas extras e de horas ociosas. Desta forma, os custos das empresas de transporte coletivo com motoristas e cobradores podem ser minimizados.

Em algumas empresas de transporte coletivo de Curitiba, a construção das jornadas de trabalho para todos os funcionários é feita manualmente, considerando a experiência da pessoa responsável por esta tarefa. Esta maneira de construir

as jornadas é demorada e está sujeita a falhas, e poderia ser melhorada se resolvida por meio de métodos de otimização. Outras empresas utilizam algoritmos heurísticos, que são baseados na construção manual destas, além de Programação Dinâmica para a construção das jornadas de trabalho, construídas a partir das tabelas de horários das linhas de ônibus (Carnieri e Steiner, 1997a e 1997b). Nestes dois casos citados, mostra-se a necessidade do desenvolvimento de um sistema eficiente para a programação dos horários dos funcionários de transporte coletivo.

Este trabalho está dividido em sete seções, incluindo-se esta introdução. Na Seção 2 deste trabalho, é feita uma revisão da literatura sobre problemas de construção de escalas de trabalho para funcionários de transporte coletivo, além de tripulações área e ferroviária. Na Seção 3 é apresentada

a descrição do problema de construção de escalas de motoristas e cobradores de ônibus. Na Seção 4, o problema do *Matching* é apresentado por meio de seu modelo matemático. Na Seção 5, a metodologia da aplicação do Algoritmo do *Matching* é apresentada, com as restrições impostas para a programação de horários para os funcionários de transporte coletivo, da cidade de Curitiba. Na Seção 6, são comentados os resultados desta aplicação, e, finalmente, na Seção 7 são apresentadas as conclusões deste trabalho.

2. Revisão da literatura

Nesta seção, são apresentados, resumidamente, alguns trabalhos que envolvem construção de escalas de trabalho, para funcionários de transporte coletivo e tripulações aérea e ferroviária.

Uma técnica de elaboração de escalas de trabalho de motoristas e cobradores de ônibus, bastante conhecida na literatura, denomina-se método Hastus (*Horaire et Assignment d'un Système de Transport Urbain ou Semi-urbain*). Blais et al. (1976), Lessard et al. (1975) e Rousseau et al. (1985a e 1985b) publicaram trabalhos sobre esse método. Por este método, as jornadas de trabalho são construídas em três fases. Na primeira fase, um modelo de Programação Linear Inteiro é desenvolvido para construir as escalas de trabalho dos funcionários, levando-se em conta restrições de horários destes e jornadas de trabalho com máxima duração permitida. As jornadas de trabalho devem ser divididas, pois estas são geradas a partir dos horários das linhas dos ônibus, que, na maioria dos casos, têm duração maior do que as jornadas de trabalho com duração máxima permitida. Com o objetivo de reduzir o tempo computacional, as restrições de integralidade e de factibilidade do problema sofrem relaxações, construindo-se as primeiras escalas, ou parte delas.

O conjunto formado pelas escalas geradas na fase anterior pode ser denominado por conjunto de escalas aproximadas, pois as restrições do Problema de Programação Linear Inteira foram relaxadas, determinando algumas soluções infactíveis. Em uma etapa seguinte, as jornadas são divididas em escalas semelhantes às escalas geradas na etapa anterior, eliminando-se infactibilidades, reduzindo-se o número de motoristas e a quantidade de horas extras, que são formadas com a geração do conjunto de escalas na etapa anterior.

Em alguns casos, as divisões das jornadas de trabalho geram escalas com duração muito pequena, o que evidencia a formação de uma jornada diária de trabalho com duas escalas diferentes, ambas com duração pequena, e, quando combinadas, formam uma jornada diária com duração total menor do que a duração máxima permitida para uma jornada diária de trabalho. A segunda fase do método Hastus consiste na combinação das escalas geradas na primeira fase, por meio da utilização do Algoritmo do *Matching*, atribuindo-se pesos às combinações factíveis de escalas de trabalho. Estas combinações de escalas determinam as jornadas de trabalho diárias dos funcionários. Na última fase, as divisões das tabelas são reconsideradas, e

algumas melhorias são feitas modificando-se as divisões das jornadas de trabalho, visando, novamente, a redução do número de funcionários e a quantidade de horas extras.

Em Manington e Wren (1975) e Wren (1981), as heurísticas de divisões sucessivas permitem a construção das escalas de trabalho em duas fases. Na primeira fase, as divisões das escalas são feitas e combinadas, formando-se uma jornada diária de trabalho completa para o funcionário. Estas escalas combinadas devem formar dias de trabalho factíveis, com a mesma idéia utilizada na primeira fase do método Hastus. Na segunda fase, as divisões das escalas são modificadas várias vezes, e o algoritmo heurístico procura formar jornadas de trabalho factíveis, melhorando as escalas de trabalho obtidas na primeira fase, com relação às quantidades de funcionários e de horas extras formadas.

Rousseau (2000) comenta que o sistema Hastus tem sido utilizado com muito sucesso, em problemas de construção de horários para veículos e de tripulação, em um grande número de contextos em todo o mundo. O autor comenta sobre sua experiência em utilizar o referido sistema em um contexto regional, no qual o transporte público é normalmente caracterizado pelo alto nível de inter-relação entre os horários dos veículos e condutores. Dado um conjunto de viagens e um número relativamente grande de pontos de parada dispersos pelo território, o problema é encontrar um horário para os veículos, que minimize o número de veículos necessários, em conjunto com o horário da tripulação, que minimiza o custo dos motoristas. Casos estudados na França e na Alemanha usando a abordagem são relatados, apresentando economias de 3 a 5%.

Desrochers e Soumis (1989) mostram a elaboração de escalas de trabalho de motoristas e cobradores de ônibus. A função objetivo minimiza os custos dos dias de trabalho, e o método de geração de colunas foi utilizado para resolver o Problema de Cobertura de Conjuntos. Na primeira fase, os dias de trabalho factíveis são gerados e formam os nós de um grafo, e os dias de trabalho factíveis devem satisfazer as mesmas restrições da primeira fase do método Hastus. Esta geração de nós utiliza um algoritmo semelhante ao método *Branch and Bound*, desenvolvido por Ryan e Foser (1975).

Na segunda fase do referido artigo, o problema de encontrar o caminho com custo mínimo foi resolvido utilizando-se o algoritmo da Programação Dinâmica *backward*. Estes caminhos formam as possíveis jornadas diárias de trabalho de motoristas e cobradores de ônibus, utilizando-se os nós gerados pelo método de geração de colunas. O trabalho foi implementado em uma cidade americana e outra britânica, e os resultados do método proposto pelos autores foram melhores do que os resultados dos algoritmos heurísticos utilizados nas cidades citadas neste artigo.

Kwan et al. (2000), comentam em seu trabalho sobre o Tracs II (*Translink Citybus*), que é um sistema desenvolvido para construção de horários para condutores de trens e de ônibus. O referido sistema possui uma abordagem de geração de novas equipes ou troca de equipes. A habilidade do Tracs II, em resolver problemas complexos para trens,

tem feito com que se torne uma ferramenta genérica para resolver uma grande variedade de problemas de transporte público, incluindo problemas de horários de ônibus e motoristas. Os autores descrevem os problemas de construção de horários para condutores de ônibus e condutores de trens.

Wren e Gualda (1997) apresentam a aplicação de sistemas desenvolvidos na Universidade de Leeds, para a construção de jornadas de trabalho de motoristas de ônibus. Estes sistemas foram aplicados nas cidades de Fortaleza, São Paulo e Sorocaba. As tabelas de horários dos ônibus são divididas em jornadas, utilizando-se o sistema Boost (*Basis for Object Oriented Scheduling of Transport*), e designadas aos motoristas usando o sistema IMPACS ou o sistema Tracs II. Cada jornada deve ter duração próxima do padrão ideal de cada cidade. Estas divisões nem sempre são bem aproveitadas, pois, nestas três cidades, existem intervalos para refeição previstos no final da manhã e no final da tarde. Estes intervalos têm duração de 30 minutos, e são colocados no final da jornada de trabalho do motorista.

Em uma primeira fase do referido trabalho, criam-se as tabelas para cobrir a demanda dos usuários de transporte coletivo, geralmente feitas por simulação. Em seguida, aplica-se um dos sistemas citados para cobrir as tabelas, com as jornadas de trabalho dos motoristas. Com as jornadas de trabalho construídas, criam-se os intervalos para refeição. Na última fase, as jornadas de trabalho e os intervalos de refeição são reconsiderados. Os resultados encontrados nas três cidades foram considerados satisfatórios, sendo que ocorreram melhorias nos resultados, quando foram feitas mudanças na duração padrão dos intervalos de refeição, e no padrão ideal de trabalho dos motoristas.

Fores et al. (2002) apresentam em seu artigo uma ferramenta de Programação Linear Inteira flexível “dirigida pelo usuário”, para a otimização do sistema Tracs II, que executa a construção de horários para condutores de ônibus e de trem. O novo sistema permite ao usuário selecionar, dentre muitas, um certo número de funções objetivo e “dirigir” a relaxação da Programação Linear, de acordo com a experiência adquirida em resolver diversos tipos de problemas relevantes.

O problema da construção de jornadas de trabalho para a tripulação ferroviária assemelha-se ao problema de jornadas para motoristas de ônibus (Caprara et al., 1997). Considera-se, inicialmente, uma tabela de serviços a ser cumprida diariamente, em certos períodos de tempo, definindo-se alguns segmentos de jornadas de trabalho, os tempos inicial e final de uma jornada, além das estações de entrada e de saída para um funcionário.

O problema, descrito em Caprara et al. (1997), consiste em encontrar um conjunto de escalas, que cobre toda a tripulação e satisfaz a demanda de passageiros, com um custo mínimo. Para resolver este problema, os autores consideram um grafo direcionado, no qual o arco (i, j) existe se, e somente se, a sequência de trabalho (i, j) existir, ou seja, o funcionário j pode substituir o funcionário i em uma determinada estação. Os autores utilizaram o modelo clássico de cobertura de vértices, com a relaxação Lagrangeana para modelos de Programação Linear Inteira. Este trabalho foi

implementado em uma companhia férrea italiana, e os resultados desta aproximação Lagrangeana foram melhores do que os resultados heurísticos, utilizados pela companhia.

Uma aplicação da Pesquisa Operacional na construção de escalas para tripulação aérea pode ser encontrada em Gama et al. (1999). Os autores propõem a resolução deste problema em duas fases. Na primeira fase, formam-se pares de segmentos de vôo em dois dias consecutivos, nos quais a tripulação viaja e retorna à cidade base. Alguns dias isolados, que não formam pares, são separados dos demais períodos.

O problema de tripulação em pares consiste em encontrar o conjunto de pares que cobre todos os segmentos de vôo, com custo mínimo. O custo de um par de vôo é dado pela duração total deste par. Na segunda fase do artigo, toma-se a listagem da tripulação, construindo-se a escala mensal personalizada para cada funcionário, de acordo com sua atividade. Neste problema, os autores optaram pela aplicação do método de geração de colunas, gerando vários subproblemas *NP-Hard*. Algumas relaxações lineares e aproximações na solução final foram feitas no método de geração de colunas, com o objetivo de reduzir o tempo computacional. Este procedimento de solução foi testado na companhia *Air France*, e os resultados encontrados são melhores do que os métodos heurísticos utilizados pela companhia.

Outro problema relacionado é ilustrado por Caprara et al. (2002). Os autores examinam o problema de um *call center* italiano, que pertence a uma companhia que atende o SOS de aparelhos eletrônicos. O problema é programar os horários dos empregados da companhia, que atendem chamadas de emergência 24 horas por dia. O objetivo para o problema é minimizar o número total de atendentes. O problema é resolvido em duas etapas. Na primeira, são determinados o número mínimo de empregados e o padrão de trabalho associado para cada empregado, ou seja, os dias nos quais o empregado trabalha e os dias nos quais ele descansa. Este passo é alcançado pela formulação do problema como um Problema de Programação Inteira, encontrando a solução por meio do *Método Branch and Bound*. Na segunda etapa é feito o balanceamento da carga horária de trabalho entre os empregados, sempre assegurando a viabilidade da solução. Esta fase é resolvida heurísticamente como uma sequência de Problemas de Transporte, um para cada dia do período.

Nota-se que os trabalhos mencionados nesta seção, que utilizam o Algoritmo do *Matching*, limitam sua implementação à fase de construção de jornadas duplas de trabalho. O presente trabalho pretende mostrar a utilização do Algoritmo clássico do *Matching*, em diversas fases da construção de jornadas de trabalho para motoristas e cobradores de ônibus, inclusive na fase final, de designação das jornadas aos funcionários.

3. Descrição do problema real

O problema proposto neste artigo tem como objetivo principal mostrar a construção de jornadas de trabalho para motoristas e cobradores de ônibus, minimizando o número de fun-

cionários, de horas extras e de horas ociosas. Este trabalho foi aplicado na cidade de Curitiba e a aplicação de metodologia semelhante pode ser implementada em outras cidades. Em Curitiba, as empresas de transporte coletivo utilizam as jornadas semanais de trabalho dos motoristas e cobradores com padrão ideal de duração igual a 36 horas, e cada funcionário trabalha 6 dias por semana. As jornadas semanais com duração maior do que o padrão ideal representam um custo maior para a empresa, pois o que excede as 36 horas é contabilizado como hora extra, e o valor pago pela hora extra é 50% maior. Jornadas semanais com duração menor do que o padrão ideal, representam ociosidade para a empresa e devem ser evitadas também, já que a empresa é obrigada a pagar pelas 36 horas nestes casos.

O serviço de transporte urbano difere dos outros serviços, pois tem características próprias. Existe uma grande demanda em alguns horários, como no início da manhã e no final da tarde, e nos demais horários a demanda é bem menor. Desta forma, existem linhas de ônibus que operam continuamente durante grande parte do dia, enquanto outras operam somente nos horários de maior demanda. As linhas de ônibus, que possuem tabelas de horário muito extensas, devem ter suas tabelas divididas em escalas de trabalho, com duração menor do que a duração máxima permitida para uma jornada diária de trabalho para um motorista de ônibus.

Assim sendo, as tabelas de horários dos ônibus são feitas de acordo com a demanda de usuários, que implica em horários de início e término desencontrados, e dificulta a elaboração das jornadas de trabalho. Outra dificuldade deste problema é a necessidade do cumprimento da legislação trabalhista e dos acordos com o Sindicato dos Trabalhadores, pois este conjunto de regras impede, muitas vezes, um melhor aproveitamento dos horários. Estas regras são apresentadas na Seção 5 deste trabalho.

O método de resolução proposto faz a construção de jornadas de trabalho em três etapas. Iniciando-se pelas tabelas dos ônibus, combinam-se as escalas de trabalho de curta duração entre si, formando-se jornadas diárias duplas de trabalho com duração próxima à duração ideal, ou seja, 6 horas. As escalas que têm duração próxima de 6 horas não necessitam de outra escala para combinar, formando-se a denominada jornada simples de trabalho. Na segunda fase, as jornadas de trabalho de dias úteis são combinadas com jornadas de trabalho de final de semana, formando-se a jornada semanal de trabalho completa de cada funcionário. Na terceira e última etapa, as jornadas de trabalho são designadas para os funcionários das empresas, finalizando-se o processo.

O Algoritmo do *Matching* com Pesos foi utilizado em todas as etapas deste trabalho, e sua formulação matemática é apresentada na seção a seguir.

4. Problema do *Matching* com pesos

Conforme Ahuja et al. (1993), um *Matching* em um grafo $G(X, A)$ é um conjunto de arcos que respeita a propriedade de todo nó ser incidente a, no máximo, um arco do conjun-

to; assim sendo, um *Matching* induz um emparelhamento de nós em um grafo, usando os arcos do conjunto A . O problema de *Matching* procura um emparelhamento que otimize um critério anteriormente estabelecido.

Existem duas maneiras de classificar problemas de *Matching*: “Problemas de *Matching* de Cardinalidade”, que maximizam o número de emparelhamentos, e “Problemas de *Matching* Ponderados”, que maximizam (ou minimizam) o peso dos arcos que formam emparelhamento.

Neste trabalho, utiliza-se o *Matching* ponderado, estabelecendo-se combinações de vértices, com o objetivo de maximizar a soma dos pesos dos arcos definidos entre os vértices combinados. Cada vértice pode ser combinado com, no máximo, um outro vértice, formando-se os emparelhamentos, como mostra o exemplo representado na Figura 1, no qual tem-se que, dos 7 vértices dados, há um *Matching* entre os vértices (1,3), (2,5), e (6,7). Observe-se ainda que, neste exemplo, o vértice 4 ficou isolado ou exposto. Vale lembrar que este é apenas um exemplo, e que neste grafo podem ser obtidos outros conjuntos de emparelhamentos.

Seja X um conjunto de vértices e A um conjunto de arcos associados aos elementos de X . Seja $G(X, A)$ um grafo não direcionado, com pesos (ou custos) p_{ij} atribuídos aos arcos $a_k \in A$, nos quais i e j são vértices do conjunto X . Denomina-se *Matching* um conjunto de pares de vértices de um grafo com pesos não nulos, tais que cada vértice deve formar, no máximo, um par. Deste modo, um *Matching* de um grafo $G(X, A)$ é um subconjunto M do conjunto A de arcos de G , escolhidos de tal maneira que cada vértice é a extremidade de, no máximo, um arco de M , ou seja, não existem arcos adjacentes em M (Christofides, 1975).

Sejam $V = \{V_1, V_2, \dots, V_z\}$ o conjunto de todos os subconjuntos de vértices com cardinalidade ímpar do grafo $G(X, A)$, T_m o conjunto de todos os arcos com as extremidades no conjunto de vértices V_m e $T = \{T_1, T_2, \dots, T_z\}$. O número de vértices de V_m é dado por $(2n_m + 1)$, em que $n_m \in \mathbb{N}$. Um conjunto *Matching* de um conjunto T_m não pode conter mais do que n_m membros de T_m . Por exemplo, um conjunto V_1 com 3 vértices tem o número de vértices dado por $2n_m + 1 = 3$, ou seja $n_m = 1$. O conjunto de arcos T_1 , que pertencem ao conjunto V_1 , não pode conter mais do que n_m arcos pertencentes ao conjunto *Matching*, ou seja, todos os subconjuntos de vértices de um grafo $G(X, A)$ com 3 vértices não podem conter mais do que um arco no conjunto *Matching*. Se esta

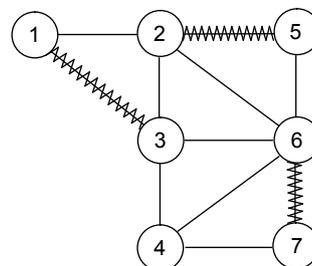


Figura 1. Representação de um conjunto *Matching*.

condição for violada, um vértice pode conter dois arcos no conjunto *Matching*.

O Problema do *Matching* de peso máximo pode ser formulado como um Problema de Programação Linear Inteiro, no qual as variáveis estruturais são x_{ij} . Assim, se o arco (i, j) não pertencer a M , $x_{ij} = 0$ e, caso contrário, $x_{ij} = 1$ (Minieka, 1978). Uma possível formulação para o Problema do *Matching* de peso máximo pode ser dada por:

$$\text{Maximizar } \sum_{(i,j)} p_{ij} x_{ij} \quad (1)$$

$$\text{Sujeito a } \sum_{j \in X} (x_{ij} + x_{ji}) \leq 1, \quad \forall i \in X \quad (2)$$

$$\sum_{(i,j) \in T_m} x_{ij} \leq n_m, \text{ onde } m = 1, 2, \dots, z \quad (3)$$

$$x_{ij} = \{0, 1\} \quad \forall (i, j). \quad (4)$$

A função objetivo (1) representa o peso total dos arcos pertencentes ao conjunto M . O conjunto de restrições (2) garante que cada nó i pode ter, no máximo, um arco (i, j) pertencente ao conjunto *Matching*, considerando-se todas as possíveis ligações entre o nó i . O conjunto de restrições (3) abrange todos os subconjuntos com cardinalidade ímpar do grafo G , impedindo-se a formação de arcos adjacentes pertencentes ao conjunto *Matching*. O conjunto de restrições (4) exige que todas as variáveis sejam binárias.

O Algoritmo do *Matching* de peso máximo foi construído baseando-se nas condições de folga complementar dos problemas primal e dual, e sua complexidade é da ordem $O(n^3)$ (Christofides, 1975), em que n é o número de vértices do grafo G . A sua complexidade algorítmica demonstra que trata-se de um algoritmo bastante eficiente.

A descrição detalhada do Algoritmo do *Matching*, devido a Edmonds e Johnson (1970), pode ser encontrada na literatura e, por ser bastante conhecida, não é apresentada neste trabalho.

O “Problema do *Matching* Ponderado em um grafo bipartido” é também conhecido como “Problema de Designação” (Christofides, 1975 e Ahuja et al., 1993).

A técnica mais conhecida na literatura para resolver o Problema da Designação é o Método Húngaro. Esta técnica é uma simplificação do Algoritmo do *Matching* de peso máximo, desconsiderando-se as formações de ciclos ímpares no grafo G , e, por este motivo, optou-se pela implementação do Algoritmo genérico do *Matching*, anteriormente especificado.

5. Metodologia adotada

Em Curitiba, as tabelas dos horários das linhas de ônibus são elaboradas pela URBS (Urbanização de Curitiba, S.A., órgão da Prefeitura Municipal) por meio de simulações da demanda de cada linha, e estas tabelas são entregues às empresas administradoras das linhas de ônibus. Nestas tabelas, existem linhas de ônibus que devem operar durante o dia todo, assim como existem linhas que operam somente em

horários de maior demanda, como no início da manhã, ou no final da tarde. Desta forma, algumas tabelas devem ser divididas em “pedaços” de trabalho, que são designados para os funcionários disponíveis de cada empresa.

Nesta fase preliminar do trabalho, são feitas divisões das escalas de longa duração das linhas de ônibus, obtendo-se escalas que podem ser combinadas com as demais, formando-se jornadas diárias de trabalho para os funcionários. Cada linha de ônibus possui os chamados pontos de “rendição”, que são locais, nos quais são possíveis as trocas de motoristas e/ou cobradores. Estes locais são geralmente terminais de ônibus ou o ponto final da linha. As divisões das escalas são feitas de acordo com estes pontos de “rendição” de cada linha. Algumas empresas utilizam um sistema que faz estas divisões, utilizando o procedimento desenvolvido por Carnieri e Steiner (1997a e 1997b), que faz uso, basicamente, da técnica da Programação Dinâmica, enquanto que outras empresas fazem todos os cálculos manualmente.

A técnica utilizada por Carnieri e Steiner (1997), para fazer as divisões das jornadas de longa duração, inicia com os cálculos da jornada de trabalho ideal para os dias úteis, e a jornada ideal para os finais de semana. O objetivo da jornada ideal está em se obter um equilíbrio, entre o número necessário de motoristas nos dias úteis e no fim de semana. A divisão de cada jornada de trabalho de longa duração é definida pela Programação Dinâmica, em que cada divisão é um estágio, e cada ponto possível de rendição é um estado da variável de estados. O valor de estado é a penalidade acumulada de quanto cada jornada se afasta da jornada ideal de trabalho, sendo que o afastamento por excesso é multiplicado por 1,5, pois a hora extra tem custo 50% maior.

Outra questão a se considerar nesta técnica é a existência das jornadas curtas, no início da manhã e no final de tarde, que devem ser combinadas com outras jornadas curtas, cujos horários devem ser próximos do horário de almoço. Estas jornadas curtas são geradas pela Programação Dinâmica, priorizando-as nas divisões próximas do horário de almoço, e utilizando a jornada ideal de trabalho, para este horário, como sendo a metade da jornada ideal utilizada nos demais horários. Desta forma, tem-se a formação de escalas de curta duração próximas do horário do almoço, que podem ser combinadas com as escalas de curta duração dos horários de *rush*, formando uma jornada de trabalho dupla. Nos casos em que a jornada dividida tem duração próxima da ideal, não existe a necessidade de combiná-la com outra jornada.

A metodologia proposta neste trabalho consta de 3 fases, descritas detalhadamente a seguir.

1ª fase: A utilização do Algoritmo do *Matching* de peso máximo na formação das jornadas diárias de trabalho

Existem três grupos de tabelas de escalas de horários das linhas de ônibus: dias úteis, sábados e domingos. Nesta fase do trabalho, as escalas longas, já divididas em uma fase preliminar, são consideradas juntamente com as escalas de curta duração, e devem ser combinadas entre si para a formação da jornada diária dupla de trabalho de um funcionário, com duração aproximada de 6 horas diárias. Em alguns casos, as escalas

de trabalho resultantes da fase preliminar já possuem duração próxima à duração ideal, de 6 horas, não havendo a necessidade de combiná-las com outras escalas. Por exemplo, na Figura 1, as escalas apresentadas podem ser combinadas para formar uma jornada diária de um funcionário, ou seja, as Escalas 1 e 3 formam uma jornada de trabalho factível, e as Escalas 2 e 4 formam outra jornada de trabalho factível, desde que a duração da jornada final não ultrapasse a duração final permitida.

Nesta fase do problema, as combinações das escalas são feitas por meio do Algoritmo do *Matching* de peso máximo, em que cada escala é representada por um vértice v de um conjunto de vértices V em um grafo $G(V, A)$, e os arcos pertencentes ao conjunto A representam combinações factíveis de escalas. Quando uma combinação de escalas não for possível, o arco entre essas duas escalas não existe, ou seja, o peso da combinação é nulo. No exemplo da Figura 1, as combinações entre as Escalas 1 e 2, e entre as Escalas 3 e 4 não podem ser feitas, pois existe coincidência de parte dos horários, o que implica na existência de pesos nulos entre estas escalas.

A construção dos pesos para as demais combinações foi proposta por Siqueira (1999) e está descrita a seguir. Nesta fase, o peso máximo (escolhido como igual a 100) é atribuído aos arcos que representam combinações que têm duração igual ao padrão ideal diário, de 6 horas. As demais combinações têm arcos com pesos proporcionais que variam de acordo com suas respectivas durações. Se uma combinação tem duração total menor do que 6 horas, o peso proporcional desta combinação é dado por:

$$p(i, j) = 100 \cdot \frac{d(i) + d(j)}{6} \quad (5)$$

em que $d(i)$ e $d(j)$ são as durações das escalas i e j , respectivamente. Quando uma combinação de escalas tem duração total maior do que 6 horas, o peso proporcional desta combinação é dado por:

$$p(i, j) = 100 - 150 \cdot \frac{(d(i) + d(j) - 6)}{6} \quad (6)$$

As igualdades (5) e (6) indicam que quanto maior o número de horas ociosas ou de horas extras, respectivamente, formadas pela combinação das escalas i e j , menor é o peso da combinação entre essas escalas.

De acordo com leis trabalhistas consideradas nesta implementação, um motorista deve ter, no máximo, duas escalas de curta duração em sua jornada diária, e estas devem ter um intervalo mínimo de 1 hora e máximo de 5 horas entre elas. As combinações que não estiverem neste intervalo, devem ter pesos nulos.

As empresas de transporte coletivo podem minimizar custos com a construção de jornadas duplas de trabalho, pois duas escalas são designadas para um único funcionário. Em Curitiba, são permitidas apenas as jornadas de trabalho simples e duplas, e não existe restrição quanto ao número de jornadas duplas. As jornadas duplas admitem um intervalo entre as duas escalas, mas, de uma forma geral, os funcionários preferem as jornadas simples, pois não possuem este intervalo, permitindo que os funcionários possam ter outras atividades no decorrer do dia. Além disso, as jornadas duplas podem ser constituídas por duas escalas de linhas de ônibus distintas, fato que também, em geral, não agrada aos funcionários.

As jornadas de trabalho diárias, construídas nesta fase do problema, não têm horário reservado para as refeições dos funcionários, sendo que eles administram estes horários da forma que mais lhes convém. Em outras cidades brasileiras, como Fortaleza por exemplo, existe um intervalo de refeição no final da manhã e outro no final da tarde, para cada funcionário (Wren e Gualda, 1997).

Conforme já mencionado, algumas empresas utilizam um algoritmo heurístico, baseado na forma como eram feitas manualmente as combinações de escalas, pelos funcionários encarregados para executar esta tarefa. Outras empresas fazem toda a construção das escalas de funcionários manualmente. Na Tabela 1, encontram-se os resultados nesta fase do problema, com a utilização do Algoritmo do *Matching* ponderado, comparando-se estes com os resultados das empresas que utilizam o algoritmo heurístico citado. Nesta tabela, o número de jornadas de trabalho obtidas representa o número de funcionários necessários, para atender a demanda das jornadas de dias úteis e de finais de semana, de cada empresa. Vale lembrar que estas quantidades consideram as jornadas simples e as jornadas duplas, cada uma delas contabilizando uma unidade na referida tabela, e que o mesmo algoritmo heurístico é utilizado por estas três empresas consideradas.

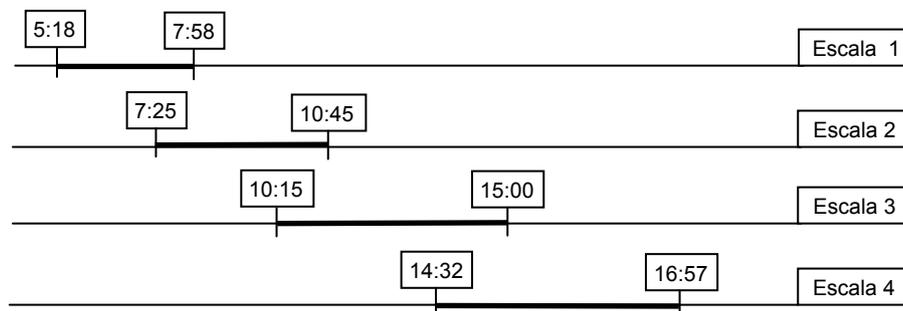


Figura 1. Exemplo de formação de escalas para a 1ª fase do problema.

2ª fase: A utilização do Algoritmo do *Matching* de peso máximo para a designação de jornadas de dias úteis para as jornadas de fim de semana.

Outra lei trabalhista determina que um horário de final de semana pode ser designado para cada motorista e cobrador. Esta é a segunda fase do problema, e o Modelo da Designação, apresentado na Seção 4.2, foi resolvido por meio do Algoritmo do *Matching* de peso máximo (Christofides, 1975 e Siqueira, 1999). Nesta fase foi atribuído o peso máximo, igual a 100, às combinações que têm duração igual ao padrão ideal semanal de 36 horas. Os demais pesos proporcionais são construídos de maneira análoga à construção de pesos da fase anterior.

O resultado desta fase é a atribuição de uma jornada de final de semana para cada jornada semanal de dia útil, formando-se a jornada semanal de trabalho de cada funcionário com custo mínimo.

Em todas as tabelas das empresas de transporte coletivo de Curitiba, o número de jornadas de final de semana (considerando-se juntamente sábados e domingos) é de 10% a 20% maior do que o número de jornadas de dias úteis, dependendo dos bairros atendidos pelas empresas. Deste modo, algumas jornadas de final de semana ficam sem designação para uma jornada de dia útil. Estas jornadas são atribuídas, então, aos funcionários denominados plantonistas, que nos dias úteis fazem plantão nas empresas, substituindo funcionários em casos de faltas ou acidentes. Para cada funcionário plantonista, devem ser designadas uma jornada de sábado e outra de domingo; desta forma, o número total necessário de motoristas é igual ao número de jornadas de dias úteis, somado com a metade do número de jornadas de final de semana sem designação. Estes resultados podem

ser constatados na Tabela 2, nas colunas referentes às jornadas de finais de semana (sábado e domingo) e na coluna de plantonistas. Nesta tabela, têm-se, então, as quantidades totais de motoristas que as empresas necessitam para cumprir a Tabela 1 construída na fase anterior.

Vale salientar que as empresas constroem as escalas separadamente, de acordo com as categorias de linhas de ônibus. As maiores categorias possuem, aproximadamente, 400 escalas diárias de trabalho.

3ª fase: A utilização do Algoritmo do *Matching* de peso máximo na designação das jornadas semanais de trabalho aos funcionários.

Após a construção das jornadas de trabalho dos motoristas e dos cobradores de ônibus, a terceira e última fase do problema consiste na designação de cada funcionário a uma jornada semanal de trabalho.

Vale lembrar que as tabelas de horários dos ônibus são bastante dinâmicas, e sofrem alterações constantemente por diversos motivos, tais como a alteração da demanda em algumas linhas de ônibus ou a implementação de novas linhas, e a eliminação de outras linhas. Nestes casos, as jornadas semanais de trabalho sofrem alterações, e uma nova designação de jornadas deve ser feita e, conseqüentemente, devem ser feitas novamente as designações de tais jornadas aos funcionários. Nos casos em que existem mudanças no quadro de funcionários, tais como funcionários que são demitidos ou que entram em férias, existe a necessidade da execução somente da designação dos funcionários às jornadas, pois estas não sofrem alterações nestes casos.

Nos dois casos citados, devem ser feitas mudanças nas designações de funcionários às jornadas de trabalho vigentes. O objetivo desta fase é, além de fazer a designação otimizada

Tabela 1. Quantidade de jornadas de trabalho construídas pelos métodos Heurístico (H) e Matching (M).

	Dias úteis		Sábado		Domingo	
	H	M	H	M	H	M
Cidade Sorriso	284	284	162	161	132	131
Redentor	291	291	165	162	134	133
Glória	305	297	174	174	151	151

Tabela 2. Quantidade de motoristas e plantonistas necessária em cada empresa.

Algoritmo heurístico						
	Dias úteis	Sábado	Domingo	Motoristas	Plantonistas	Total
Cidade Sorriso	284	162	132	284	5	289
Redentor	291	165	134	291	4	295
Glória	305	174	151	305	10	315
Algoritmo exato do <i>Matching</i>						
	Dias úteis	Sábado	Domingo	Motoristas	Plantonistas	Total
Cidade Sorriso	284	161	131	284	4	288
Redentor	291	162	133	291	2	293
Glória	297	174	151	297	14	311

dos funcionários às jornadas, minimizar as alterações nas jornadas de trabalho, evitando transtornos aos funcionários. Isto é feito, verificando-se a proximidade das jornadas anteriores com as atuais, por meio dos horários de início e de término das jornadas, e comparando-se as linhas que os funcionários faziam anteriormente, e as linhas das tabelas atuais.

Novamente, o Algoritmo do *Matching* com pesos foi utilizado, atribuindo-se o peso máximo igual a 100, às designações de jornadas que mais se aproximam das jornadas que os funcionários faziam anteriormente.

Inicialmente, todas as combinações de escalas para funcionários possuem peso máximo. A cada intervalo de tempo, que a nova jornada difere da jornada anterior de um funcionário, o peso da designação da nova jornada para o funcionário sofre um desconto. Utilizou-se um desconto de 20 pontos para cada 15 minutos de diferença entre as jornadas. Quando a linha da nova jornada é diferente da linha que o funcionário fazia, o peso da designação do funcionário, para a nova jornada, sofre um desconto de 20 pontos.

Na Tabela 3, encontram-se os resultados obtidos nesta fase, na qual são colocados os pesos máximo e mínimo encontrados em cada empresa, além do peso médio que mede o nível de satisfação dos funcionários, com as novas tabelas encontradas.

A otimização na designação dos funcionários às jornadas é extremamente importante, pois alguns funcionários não têm disponibilidade em todos os horários, já que podem possuir um outro emprego, ou então podem estudar em um determinado período. Desta forma, a nova jornada de um funcionário deve ter o horário compatível com sua disponibilidade, e, quando esta restrição for violada, o peso da designação do funcionário, para a nova jornada, deve ser nulo.

Quando um funcionário tem jornadas simples, e sua preferência é pela jornada simples, o peso das combinações deste funcionário com as jornadas duplas sofre um desconto de 50 pontos.

6. Análise dos resultados

A metodologia apresentada na Seção 5 deste trabalho foi aplicada em três empresas de transporte coletivo da cidade de Curitiba, conforme já mostrado na mesma seção, na qual os algoritmos utilizados anteriormente eram todos heurísticos. Os resultados obtidos pelo Algoritmo do *Matching* de peso máximo foram melhores do que os resultados dos algoritmos heurísticos, em todas as três fases.

Na Tabela 1, que apresenta a quantidade de jornadas diárias de trabalho (1ª fase), pode-se verificar que houve uma redução média de, aproximadamente, 1% para os dias úteis e 0,7% para os finais de semana, com a utilização do Algoritmo do *Matching* em relação ao algoritmo heurístico.

Já para a 2ª fase do problema, dando continuidade à resolução do problema, tem-se na Tabela 2, que apresenta a quantidade total de motoristas e plantonistas para cada empresa, uma melhoria média de, aproximadamente, 0,8%, que, a princípio, pode não parecer tão significativa, mas que na verdade retrata a economia com 7 funcionários [(289 + 295 + 315)-(288 + 293 + 311)],

Tabela 3. Pesos encontrados na 3ª fase da designação.

	menor peso	maior peso	média
Cidade Sorriso	10	100	87
Redentor	30	100	94
Glória	20	100	92

Tabela 4. Comparativo do resultado final pelo método Heurístico (H) e pelo método do *Matching* (M).

		Motoristas	Horas extras	Horas ociosas
Cidade Sorriso	H	289	150	227
	M	288	140	122
Redentor	H	295	142	197
	M	293	137	172
Glória	H	315	172	241
	M	311	170	190

melhoria que não pode ser desprezada pelas empresas.

Para a 3ª fase, considerando-se os pesos como nível de satisfação dos funcionários com as novas jornadas de trabalho, obteve-se, em média, níveis superiores a 90% com a aplicação do Algoritmo do *Matching*.

Estes resultados foram considerados bastante satisfatórios, mostrando que a implementação deste trabalho nas empresas citadas seria bastante útil. Deve-se observar que, apesar do Algoritmo do *Matching* ser um algoritmo exato, os resultados obtidos neste trabalho são aproximados, já que o referido algoritmo foi aplicado em três diferentes etapas. Há ainda o aspecto dos pesos, pois como estes podem ser tratados de diferentes formas, o desempenho do Algoritmo do *Matching* também poderá variar dentro de uma mesma fase.

Na Tabela 4 encontram-se os resultados para as empresas pesquisadas e, como pode-se verificar, houve uma redução significativa na quantidade de horas extras e de horas ociosas, nestas empresas. Os valores para as referidas horas (extras e ociosas) foram obtidos somando-se as diferenças entre as durações das jornadas semanais de trabalho, construídas neste trabalho, e a duração da jornada padrão ideal, que neste caso é de 36 horas semanais.

O tempo computacional total para a execução do Algoritmo do *Matching* em todas as três fases, em um microcomputador *Pentium II*, de 400 MHz, foi de, aproximadamente, 10 minutos em problemas em que 600 vértices foram considerados. Vale lembrar que tais vértices podem ser considerados somente escalas de trabalho (nas duas primeiras fases do problema), bem como jornadas de trabalho e funcionários (na última fase). Na última fase, encontram-se as maiores matrizes de vértices, pois têm-se jornadas de trabalho e funcionários, aumentando-se a dimensão do problema e, conseqüentemente, o tempo computacional para a execução desta fase.

7. Conclusões

O presente trabalho demonstra que o Algoritmo clássico do *Matching* com pesos pode ser aplicado em problemas de

construção de jornadas para funcionários de transporte coletivo, de forma bastante satisfatória.

A eficácia da metodologia proposta foi comprovada, e seu tempo computacional foi extremamente satisfatório, mostrando, desta forma, que algoritmos heurísticos clássicos também podem ser utilizados na resolução deste tipo de problema, apenas lembrando que as suas soluções nem sempre são ótimas globais.

Outro fator que influenciou positivamente a eficácia da metodologia proposta, foi a sua execução por partes, conforme as categorias das linhas de ônibus. Como o Algoritmo do *Matching* com pesos é da ordem $O(n^3)$, ao considerar-se um número muito grande de escalas, o seu tempo computacional, provavelmente, comprometeria a sua implantação. De qualquer forma, esta metodologia de resolução pode ser aplicada em cidades em que as linhas de ônibus não são separadas por categorias também, desde que em uma fase inicial efetue-se uma separação em grupos de linhas, com o objetivo de reduzir o tempo computacional do algoritmo.

Este trabalho foi implantado nas empresas citadas, e, alguns meses depois, em outras três empresas que operam na cidade de Curitiba, além de outras duas que operam na região metropolitana da cidade.

A metodologia aqui apresentada, desenvolvida e aplicada aos dados dos motoristas de ônibus, foi utilizada de forma análoga aos dados dos cobradores. A única diferença entre

motoristas e cobradores é que, ao final da jornada, os cobradores possuem trinta minutos a mais em suas jornadas, para que possam deslocar-se até a empresa, para prestar contas de seu dia de trabalho. Por este motivo, as três fases devem ser executadas em separado, não existindo a possibilidade de aproveitamento das escalas construídas para os motoristas nas duas primeiras fases mostradas neste trabalho. Os resultados para os cobradores foram omitidos, pois são muito parecidos com os resultados mostrados para os motoristas de ônibus.

Uma metodologia semelhante pode ser aplicada, também, a qualquer outra cidade, fazendo-se as adaptações necessárias, de acordo com os interesses e leis trabalhistas locais. Em cidades, nas quais são previstos intervalos para refeição, o método proposto neste artigo pode ser facilmente adaptado, considerando-se que o horário de refeição deve ser cumprido em um determinado período, com sua respectiva duração.

Os problemas de tripulação aérea e ferroviária, citados na Seção 2 deste trabalho, também podem ser modelados de forma semelhante a que foi apresentada neste trabalho. As seqüências de trabalho de tripulações ferroviárias ou aéreas que são factíveis, podem ser representadas por arcos de um grafo não direcionado, com um determinado peso atribuído a cada combinação. Desta forma, o Algoritmo do *Matching* com pesos pode ser aplicado para estes problemas, e as restrições de cada tipo de tripulação podem ser facilmente adaptadas.

Referências Bibliográficas

- AHUJA, R.K.; MAGNANTI, T.L.; ORLIN, J.B. **Network Flows – Theory, Algorithms and Applications**. Prentice Hall, New Jersey, 1993.
- BERGMANN, D.R.; BODIN, L. **Preprint workshop on automated technique for scheduling of vehicle operators for urban public transportation services** (eds.). Chicago, USA, 1975.
- BLAIS, J. Y.; LAPORTE, G.; LESSARD, R.; ROUSSEAU, J.M.; Soumis, F. **The problem of assigning drivers to bus routes in an urban transit system**. Report, n. 44. Centre De Recherche Sur Les Transports, Université de Montréal, 1976.
- CAPRARA, A.; FISCHETTI, M.; TOTH, P.; VIGO, D.; GUIDA, P.L. Algorithms for railway crew management. **Mathematical Programming**, n. 79, p. 125-141, 1997.
- CAPRARA, A.; MONACI, M.; TOTH, P. Models and Algorithms for Staff Scheduling Problems. **Abstracts Applied Combinatorial Optimization - IV ALIO/EURO Workshop**, Pucón, Chile, 2002.
- CARNIERI, C.; STEINER, M.T.A. Um sistema de programação de escala para motoristas de ônibus. **Anais do XX CNMAC, Congresso Nacional de Matemática Aplicada e Computacional**. Gramado, RS, 1997a.
- _____. A system for bus drivers scheduling, **Euro XV, Informatics XXXIV- Institute for Operations Research and the Management Sciences**. Barcelona, Espanha, 1997b.
- CHRISTOFIDES, N., **Graph Theory, an algorithmic approach**. Academic Press, New York, USA, 1975.
- DESROCHERS, M.; SOUMIS, F. A column generation approach to the urban transit crew scheduling problem. **Transportation Science**, v. 23, n. 1, p. 1-13, 1989.
- EDMONDS, J.; JOHNSON, E.L. **Matching: a well solved class of integer linear programs. Combinatorial structures and their applications**. Gordon And Breach, New York, USA, 1970.
- FORES, S.; PROLL, L.; WREN, A. A Hybrid IP/Heuristic Driver Scheduling System for Public Transport. **Journal of Operational Research Society**, v. 53, n. 10, p. 1093-1100, 2002.
- GAMACHE, M.; SOUMIS, F.; MARQUIS, G.; DESROSIERS, J. A column generation approach for large-scale aircrew rostering problems. **Operations Research**, v. 47, n. 2, p. 247-263, 1999.
- KWAN, A. S. K.; KWAN, R. S. K.; PARKER, M. E.; WREN, A. Proving the Versatility of Automatic Driver Scheduling on Difficult Train & Bus Problems. **Abstracts Computer-**

- Aided Scheduling of Public Transport (CASPT) - 8th. International Conference**, Berlin, Germany, 2000.
- LESSARD, R.; ROSSEAU, J.M.; DUPUIS, D.; **Hastus I: a mathematical programming approach to the bus driver scheduling problem**. Em Bergmann e Bodin, 1975.
- MANINGTON, B.; WREN, A. **A general method for bus crew scheduling**. Em Bergman e Bodin, 1975.
- MINIEKA, E., **Optimization algorithms for networks and graphs**. Marcel Dekker, New York, USA, 1978.
- ROUSSEAU, J.M., **Computer scheduling of public transport 2**, Elsevier, Amsterdam, 1985a.
- ROUSSEAU, J.M.; LESSARD, R.; BLAIS, J.Y. **Enhancement to the Hastus crew scheduling algorithm**, Em Rousseau, 1985b.
- ROUSSEAU, J.M., Scheduling Regional Transportation with Hastus. **Abstracts Computer-Aided Scheduling of Public Transport (CASPT) – 8th. International Conference**, Berlin, Germany, 2000.
- RYAN, D. M.; FOSER, B. A. **An integer programming approach to scheduling**. Em Bergman e Bodin, 1975.
- SIQUEIRA, P.H., **Aplicação do Algoritmo do Matching no Problema da Construção de Escalas de Motoristas e Cobradores de Ônibus**, 1999. Dissertação (Mestrado) - UFPR, Curitiba, 1999.
- WREN, A., **Computer scheduling of public transport urban passenger vehicle and crew scheduling**, Elsevier, Amsterdam, 1981.
- WREN, A.; GUALDA, N. D. F. **Integrated Scheduling of buses and drivers**, School of Computer Studies, Research Report Series, University of Leeds, 1997.

A PROPOSED SOLUTION FOR BUS DRIVER AND FARE COLLECTOR SCHEDULING PROBLEMS USING THE MAXIMUM WEIGHT MATCHING ALGORITHM

Abstract

The purpose of this paper is to discuss how the maximum weight Matching Algorithm can be applied to schedule the workdays of bus drivers and bus fare collectors. This scheduling should be based on the best possible use of timetables in order to minimize the number of employees, overtime and idle hours, thereby minimizing the operational costs of public transportation companies. In the first phase of this study, assuming that the timetables are already divided into long and short duration schedules, the short schedules can be combined to make up an employee's workday. This combination is done by the maximum weight Matching Algorithm, in which the scales are represented by vertices on a graph and the maximum weight is attributed to combinations of scales that do not lead to overtime or idle hours. In the second phase, a weekend schedule is assigned for every weekly work schedule. Based on these two phases, the weekly work schedules of bus drivers and bus fare collectors can be arranged at a minimal cost. The third and final phase of this study consisted of assigning a weekly work schedule to each bus driver and collector, considering his/her preferences. The maximum weight Matching Algorithm was also used in this phase. This method was applied in three public transportation companies in Curitiba, state of Paraná, which had until then used old heuristic algorithms based solely on managerial experience.

Keywords: *schedule optimization, maximum weight Matching Algorithm.*