

## Método heurístico para a otimização da programação de tripulações no transporte público urbano

Silvia Lopes de Sena Taglialenha, [s.taglialenha@ufsc.br](mailto:s.taglialenha@ufsc.br)  
Juan Carlos Guerra Trinidad, [jcguerratrinidad@gmail.com](mailto:jcguerratrinidad@gmail.com)

**Resumo:** O Problema de Programação de Tripulação (PPT) de ônibus urbano consiste em determinar o número mínimo de tripulantes e especificar suas viagens de tal forma a cobrir todas as viagens da frota em operação com o menor custo possível e atendendo as restrições relacionadas à operação e às normas trabalhistas da categoria. É classificado como um problema NP-Difícil, sendo, portanto, importante a aplicação de métodos heurísticos para resolver de forma satisfatória os problemas de grande porte. Neste trabalho apresenta-se um método heurístico que utiliza um método construtivo para obtenção de solução inicial e uma busca de diferentes tipos de jornadas para reotimização. O método foi implementado em C++ em problemas disponíveis na literatura e os resultados obtidos são comparados e mostraram-se bastante satisfatórios.

**Palavras chave:** Sequenciamento de tarefas, Problema de Programação de Tripulação, Heurísticas de descida.

## Heuristic method for optimizing crew scheduling in urban public transport

**Abstract:** The Crew Scheduling Problem is to determine the minimum number of crew members and specify their trips in such a way as to cover all trips of the operating fleet at the lowest possible cost and according the restrictions related to operation and labor standards. It is classified as a NP-Hard problem, so it is important to apply heuristic methods to solve large problems. This paper presents a heuristic method that uses a constructive method to obtain an initial solution and a search with different types of schedule works for reoptimization. The method was implemented in C++ in problems available in the literature and the results are compared and proved to be quite satisfactory.

**Key-words:** Task scheduling, Crew Scheduling problem, Descent heuristics.

### 1. Introdução

O transporte público é um direito que todo cidadão possui garantido por lei através do Inciso XX do Artigo 21º da Constituição Federal Brasileira de 1988 e da Lei Federal 12.578 de 2012 que trata da Política Nacional de Mobilidade Sustentável. O serviço de transporte coletivo promove a igualdade e inclusão social, possibilitando o acesso às infraestruturas urbanas de saúde, educação, trabalho e lazer ao cidadão.

Como na maioria das cidades brasileiras o mesmo é realizado de modo rodoviário por meio de ônibus, é indispensável, na tentativa de minimizar os efeitos negativos ambientais e

econômicos causados pelo uso excessivo de automóveis e motocicletas, que o planejamento e controle de aspectos operacionais do transporte coletivo sejam melhores considerados. Problemas de aspecto operacional implicam em uma variedade de problemas de tomada de decisão que podem variar, por exemplo, da decisão do melhor itinerário, alocação de frotas e condutores, grade de horários, até a formulação de novas linhas de ônibus. Tais fatores impactam diretamente no custo de prestação de serviço, o que por sua vez, podem encarecer a tarifa do transporte.

Neste sentido, neste trabalho é considerado o problema de alocação de frota entre terminais tendo como objetivo a redução do tempo de ociosidade dos veículos entre as viagens. É apresentada uma modelagem de programação matemática inteira baseada no tradicional modelo de fluxo de custo mínimo proposto por Dantzig et al. (1959), cujo objetivo do modelo matemático é determinar o menor custo (temporal ou financeiro) associado à realização de um grupo de viagens, dado que cada uma das viagens deve ser realizada por um único veículo de uma frota.

Os problemas reais que envolvem a alocação de frota em geral são de grande porte e em geral não é possível resolvê-los utilizando métodos exatos. Neste caso, as heurísticas aparecem como soluções eficientes para o tratamento desses problemas (normalmente NP-difíceis).

Neste trabalho apresenta-se um método heurístico construtivo para resolver um problema real de alocação de frotas para uma empresa de transporte público.

## 2. Definição do problema e referencial

Segundo Silva et al. (2014) as empresas do sistema de transporte público brasileiro fazem pouco uso de softwares e modelos de otimização para alocar o seus equipamentos e a mão de obra necessária para a sua operação, no entanto a programação de escala de motoristas, tornou-se um fator econômico altamente relevante para empresas de transporte público. Para ter um menor custo na folha de pagamento com a redução de número de motorista e assim atrair mais usuários para o sistema de transporte público com a redução de valor da tarifa.

No sistema de transporte público por ônibus, os custos de mão-de-obra para os motoristas e outros funcionários representam uma parcela significativa dos orçamentos das operadoras de ônibus, em alguns casos ultrapassando em 50%, conforme (XIE et al., 2017).

Segundo Silva et al. (2014), o planejamento de um sistema de transporte público é decomposto em quatro etapas por questões de simplificação, e assim o problema é tratável do ponto de vista computacional.

A etapa inicial é constituída pela elaboração das rotas e dos quadros de horários, de maneira que todas as regiões da cidade sejam atendidas e que o tempo de espera nos pontos e terminais atenda a um dado nível de qualidade estipulado no processo licitatório.

A segunda etapa trata do problema da alocação de veículos que realizarão as viagens das tabelas de horários. Nessa etapa, é determinada a quantidade necessária de veículos para atender as viagens de cada rota, assim como especificar as viagens a serem realizadas por cada veículo durante a operação.

A terceira etapa deve determinar a alocação das tripulações, tomando como base as viagens alocadas a cada veículo da frota em operação. Nessa etapa, deve ser determinada a quantidade necessária de tripulantes para conduzir os veículos e quais as viagens que cada tripulação realizará. Dessa maneira são definidas as jornadas diárias de trabalho que deverão ser executadas pelas tripulações da empresa.

A etapa final do planejamento da operação do sistema diz respeito ao problema do rodízio das tripulações. Nesta etapa são geradas as escalas mensais das tripulações com o menor custo possível de maneira a cobrir todas as jornadas diárias, respeitando as regras trabalhistas para o período. Essas etapas geralmente são realizadas na ordem que foram expostas e a solução do problema de uma etapa é usada como entrada da etapa subsequente, segundo Frieberg et al. (1999)

No seguinte trabalho é considerada a terceira etapa, conhecida na literatura especializada como, Problema da Programação de Tripulações (PPT) (*crew scheduling problem*). Nessa etapa são definidas as jornadas de cada tripulante, portanto, um tripulante está associado a uma jornada diária de trabalho e vice-versa. Uma jornada é uma sequência de tarefas dentro de um dia que é executada por um motorista de acordo com os regulamentos de trabalho.

Para atender as demandas que ocorrem especificamente nos horários de pico as jornadas podem ser inteiras, quando o tempo de espera entre as viagens é inferior a duas horas, ou de dupla pegada, quando o intervalo ultrapassa duas horas. Os tripulantes podem realizar uma troca de veículos durante uma jornada de trabalho, o que pode ser uma forma de reduzir o tempo ocioso das tripulações. Mas na prática, em geral operadora empresa restringe o número de trocas realizadas pelas tripulações, normalmente, para facilitar o controle sobre o desgaste dos veículos em operação.

Uma jornada deve ser definida de tal forma que o tempo de descanso entre o seu final e seu início no dia seguinte seja maior ou igual ao tempo mínimo de descanso estipulado pela legislação. Assim, uma mesma tripulação pode realizá-la em dias consecutivos sem infringir a regra do descanso em casa. A jornada também pode ser classificada em dois tipos, sendo nomeados como jornada inteira e jornada meia. A jornada inteira caracteriza-se por um período total de trabalho de 07h20min com um intervalo obrigatório para almoço de uma a duas horas e podem-se acrescentar duas horas a mais, classificadas como hora extra. A jornada meia adota uma jornada de quatro horas totais e nesta não há intervalo para almoço e também para a extensão de hora extra.

Em cada viagem há um custo associado aos salários fixos, horas extras ou ociosas durante a jornada de cada motorista e os custos artificiais como a troca de veículos ou uma dupla jornada. A melhor opção de escala deverá buscar balancear o total de horas extras e ociosas dentro do horizonte de planejamento, assim como minimizar o número de total de motoristas.

Fores et al. (199) afirma que o PPT consiste em determinar o número mínimo de tripulantes e especificar suas viagens de tal forma a cobrir todas as viagens da frota em operação com o menor custo possível. Nessa etapa são definidas as jornadas de cada tripulante, portanto, um tripulante está associado a uma jornada diária de trabalho e vice-versa. Uma jornada é uma sequência de tarefas dentro de um dia que é executada por um motorista de acordo com os regulamentos de trabalho.

As jornadas podem ser de dois tipos:

- Jornada inteira: quando o tempo entre as viagens é inferior a duas horas, e;
- Dupla pegada: que são jornadas que apresentam um intervalo maior do que duas horas entre duas viagens.

Essas jornadas existem para atender as demandas mais acentuadas que ocorrem especificamente nos horários de pico. Durante uma jornada de trabalho, os tripulantes podem realizar uma troca de veículos, que pode ser uma forma de reduzir o tempo ocioso das tripulações.

Os problemas de programação da tripulação visam gerar jornadas de trabalho, enquanto algumas regulamentações de trabalho devem ser garantidas, como por exemplo a duração máxima de uma jornada de trabalho, o mínimo de intervalos durante um serviço e cumprir o tempo de descanso entre o seu final e seu início no dia seguinte seja maior ou igual ao tempo mínimo de descanso estipulado pela legislação. Na seção seguinte apresenta-se uma modelagem matemática para o PPT.

### 3. Modelagem matemática do PPT

Considerando-se duas tarefas (viagens)  $i$  e  $j$ , definem-se as variáveis de decisão binárias  $y_i^k$ ,  $w_{ij}^k$  e  $x_{ij}^k$  como a seguir:

- $y_i^k = \begin{cases} 1, & \text{se agente } k \text{ realiza viagem } i; \\ 0, & \text{caso contrário} \end{cases}$ ;
- $w_{ij}^k = \begin{cases} 1, & \text{se agente } k \text{ realiza viagem } j \text{ imediatamente depois da viagem } i; \\ 0, & \text{caso contrário} \end{cases}$ ;
- $x_{ij}^k = \begin{cases} 1, & \text{se agente realiza uma ou mais viagens entre as viagens } i \text{ e } j; \\ 0, & \text{caso contrário} \end{cases}$ .

E então o modelo matemático do PPT, pode ser representado pelas (In)Equações (1)-(15), em que  $HO_k$  define o total de horas ociosas do agente  $k$  e  $HE_k$  o total de horas extras do agente  $k$ , para  $m$  tarefas (viagens) e  $n$  motoristas (agentes) adaptado de Smith et al. (1988).

$$\text{Min } Z = \sum_{k=1}^m HO_k + HE_k \quad (1)$$

Sujeito às restrições:

$$\sum_{k=1}^m y_i^k = 1 \quad \forall i = 1, \dots, n \quad (2)$$

$$y_i^k + y_j^k \leq 1 \quad \forall k = 1, \dots, m, \forall i, j = 1, \dots, n, j > i, hc_i > hs_j \quad (3)$$

$$\sum_{l=i+1}^{j-1} y_l^k \leq x_{ij}^k * (j - i - 1) \quad \forall k = 1, \dots, m, \forall i, j = 1, \dots, n \text{ com } j > i + 1 \quad (4)$$

$$\sum_{l=i+1}^{j-1} y_l^k \geq x_{ij}^k \quad \forall k = 1, \dots, m, j = 1, \dots, n, j > i + 1 \quad (5)$$

$$w_{ij}^k \leq y_i^k \quad \forall k = 1, \dots, m, i, j = 1, \dots, n, j > i \quad (6)$$

$$w_{ij}^k \geq y_i^k - x_{ij}^k - 1 \quad \forall k = 1, \dots, m, i, j = 1, \dots, n, j > i \quad (7)$$

$$w_{ij}^k \leq 1 - x_{ij}^k - 1 \quad \forall k = 1, \dots, m, i, j = 1, \dots, n, j > i \quad (8)$$

$$\sum_{i=1}^{n-1} \sum_{\substack{j=i+1 \\ 1 \leq hs_i - hc_i \leq 2}}^n w_{ij}^k = 1 \quad \forall k = 1, \dots, m \quad (9)$$

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^n w_{ij}^k = 0 \quad \forall k = 1, \dots, m \quad (10)$$

$$(hs_i - hc_i) * (y_i^k + y_j^k - 1) - In_k \leq 9.2 - Tm \quad \forall k = 1, \dots, m, \forall i, j = 1, \dots, n \quad (11)$$

$$HE_k \geq (hc_i - hs_i + y_i^k + y_j^k - 1 - Int_k) - 9.2 \quad \forall k = 1, \dots, m, i, j = 1, \dots, n, j > i \quad (12)$$

$$x_{ij}^k, w_{ij}^k \in \{0,1\} \quad \forall k = 1, \dots, m, \forall i, j = 1, \dots, n \quad (13)$$

$$y_i^k \in \{0,1\} \quad \forall k = 1, \dots, m, \forall i = 1, \dots, n \quad (14)$$

$$HO_k, HE_k \geq 0 \quad \forall k = 1, \dots, m \quad (15)$$

A Equação (1) representa a função objetivo que minimiza a soma total de horas extras e tempo ocioso das tripulações.

Considera-se hora extra qualquer excedente a 07h20min de trabalho diário. As restrições das Equações (2) garantem que todas as viagens sejam realizadas por apenas uma tripulação, ou seja, para cada viagem  $i$  deve existir, exatamente uma tripulação  $k$  que a realize. Nas Inequações (3) garante-se que cada tripulação  $k$  só realizará a próxima viagem  $j$  depois de ter realizado a viagem anterior  $i$ .

Ressalta-se que a tripulação  $k$  somente poderá realizar a tarefa  $j$  após a tarefa  $i$  se o horário de chegada ( $hc_i$ ) da viagem  $i$  é menor que o horário de saída ( $hs_j$ ) da viagem  $j$ .

As restrições das Inequações (4), (5), (6), (7) e (8) asseguram que a tripulação  $k$  só poderá realizar a tarefa  $l$  entre as tarefas  $i$  e  $j$  somente se também tiver executado a tarefa  $i$  e  $j$ . E nas Inequações (9) garante-se que cada tripulação  $k$  tenha somente um intervalo de

repouso e/ou alimentação, de no mínimo uma hora de no máximo duas horas na sua jornada de diária de trabalho.

As Equações (10) garantem que cada tripulação não tenha intervalos superiores a duas horas na jornada. Assim como a Inequações (11), em que  $In_k$  indica o intervalo do agente  $k$ , garantem que nenhuma jornada de cada tripulação  $k$  ultrapasse a jornada máxima ( $Tm$ ). Nas Inequações (12) limitam-se a quantidade máxima de horas extras permitidas. As restrições de (13) a (15) restringem o domínio das variáveis de decisão.

Com a modelagem (1)-(15) é possível resolver o PPT de forma exata, porém para um número pequeno de tarefas (viagens), uma vez que uma versão menos restrita do problema, o problema de alocação simples, já é um problema classificado como *NP-Hard*, ou seja, problemas para os quais não existem algoritmos cujo tempo de resposta são polinômiais em função da entrada de dados.

Quando não é possível determinar a solução exata de um dado problema de otimização é comum se recorrer ao uso de métodos alternativos, que em geral podem determinar uma solução aproximada para o problema em um tempo computacional razoável. Esses métodos alternativos são conhecidos na literatura como métodos heurísticos ou meta heurísticos, dependendo de sua complexidade de implementação.

Segundo Hillier et al. (2010) os métodos exatos são mais eficientes para problemas de pequena dimensão ou que não são classificados como NP-hard. Para problemas com grande dimensão, o tempo de implementação tende a aumentar exponencialmente a cada iteração, então evita-se aplicar modelos exatos. Mesmo assim, existem abordagens exatas para a resolução do problema, entre elas a programação dinâmica, programação linear e inteira, programação por restrições, geração de colunas, *branch and bound* e relaxação lagrangeana.

Assim, é necessário utilizar métodos heurísticos para resolver problemas que aparecem na vida real, que são amplos e demandam alto custo computacional, obtendo-se resultados muito interessantes, embora a otimalidade das soluções não seja garantida. Lourenço et al. (2001) afirma que o desenvolvimento de métodos heurísticos para a obtenção de resultados de forma satisfatória para os problemas de grande porte traz grandes contribuições acadêmicas.

Já os algoritmos meta heurístico são procedimentos destinados a encontrar uma boa solução, eventualmente a ótima, consistindo na aplicação de uma heurística subordinada, a qual tem que ser modelada para cada problema específico. A principal característica dos algoritmos meta heurísticos é a habilidade que estas possuem de escapar de ótimos locais em tempo computacional aceitável.

Nurmi et al. (2011) propõe uma solução com a combinação dos métodos de *Simulated Annealing* e *Tabu Search* em uma empresa finlandesa de transporte urbano. A solução inicial foi criada alocando as jornadas em dias aleatórios. As próximas jornadas são aceitas se causarem uma diminuição na função objetivo. O algoritmo mostrou-se eficiente para entradas randômicas de dados. Mais aplicações aparecem em (LI et al. , 2003; MESQUITA et al., 2011; FESTA et al., 2018).

#### 4. Método de solução proposto

O método heurístico proposto será aplicado e testado no problema considerado em (SILVA et al., 2014).

A empresa em estudo forneceu um banco de dados com as informações das escalas atuais praticadas pela mesma. Neste banco de dados são detalhadas todas as tarefas a serem cumpridas como mostrado na Tabela 1.

Linha	Saída	Chegada	Ponto inicial	Ponto final
$L_1$	$h_{s1}$	$h_{c1}$	$p_{i1}$	$p_{f1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$L_n$	$h_{sn}$	$h_{cn}$	$p_{in}$	$p_{fn}$

Fonte: Elaborado pelo autor (2019).

**Tabela 1** – Representação de uma escala.

Os parâmetros número de motoristas utilizados nas escalas utilizadas pela empresa em cada dia da semana são apresentados na Tabela 2.

Tipo de jornada	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
Interia	134	130	149	162	155	124	68
Pegada Dupla	6	3	5	4	1	0	0

Fonte: Elaborado pelo autor (2019).

**Tabela 2** – Análise das escalas utilizadas pela empresa.

Para a obtenção de uma proposta de solução neste trabalho considerou-se as tarefas indicadas no banco de dados.

Inicialmente aplica-se uma heurística construtiva para obtenção de solução inicial, cujo pseudocódigo é mostrado na Figura 1.

**Figura 1** – Pseudocódigo da heurística construtiva para obtenção da solução inicial.

```

Início Solução Inicial
1. Lt ← NewList <Tarefa>( );
2. Ordenar tarefas em uma lista em ordem crescente de hora de início.
3. Lm ← NewList <Motorista> (empty list);
4.
5. Para cada tarefa t em Lt, faça
6.
7.   Para cada Motorista em Lm faça
8.     Verifica se o Motorista está disponível para fazer a tarefa t
9.   Fim-para
10.
11.   Update: atualiza o status de todos os motoristas existentes
12.
13.   Se nenhum motorista for encontrado então
14.     Criar novo motorista m em Lm
15.     Adicionar t em motorista m
16.
17.   Se não

```

18. Adicionar t ao motorista com menor tempo entre tarefas
  19. Atualizar status do motorista selecionado
  20. **Fim-se**
  21. **Fim-para**
- Fim Solução Inicial**

Fonte: desenvolvido pelo autor (2019)

Inicialmente são ranqueadas as tarefas em ordem crescente de hora de início. Em seguida, para cada uma das tarefas, verifica-se a disponibilidade de algum motorista para receber a tarefa. A verificação acontece procurando em todos os motoristas existentes, selecionando somente os motoristas que estejam cumprindo os seguintes requisitos:

- Estar no ponto inicial onde a tarefa começa;
- Não estar realizando nenhuma outra tarefa;
- Verificar se esta cumprindo com as leis trabalhistas.

Posteriormente à atualização da lista de motoristas que estão disponíveis, é escolhido o motorista com menos minutos ociosos entre a hora que terminou sua última tarefa e o início da tarefa que será adicionada. Quando o motorista é escolhido se realiza a função do *update* cujo objetivo é fazer com que, toda vez que for verificado se existe algum motorista disponível com a tarefa escolhida, essa função atualize o status de todos os motoristas existentes, a fim de reduzir o tempo de descanso, otimizar as escolhas das tarefas que vão ser inseridas e estar o mais rápido possível para executar outra tarefa. Assim o resultado final é mais eficiente.

O status de cada motorista é classificado da seguinte maneira:

- Status 0: O motorista está no primeiro período da jornada;
- Status 1: O motorista está seu horário de descanso obrigatório;
- Status 2: O motorista está no seu segundo período da jornada;
- Status 3: O motorista está em seu período de horas extras.

Se depois da verificação não existir nenhum motorista disponível, cria-se um novo motorista no ponto inicial para realizar essa tarefa.

Isto garante que todas as tarefas serão alocadas a exatamente um único motorista e que nenhum motorista possuirá a sua jornada violada pelas leis trabalhistas. Assume-se que número de motoristas seja suficientemente grande a fim de possibilitar a alocação de todas as tarefas a cada um dos motoristas. Ou seja, o número de motoristas é suficiente para a realização de todas as tarefas do problema, e que as condições de restrição sejam sempre satisfeitas.

#### 4. Resultados

A heurística proposta neste trabalho foi aplicada considerando-se os dados de várias empresas de transporte público da cidade de Belo Horizonte disponíveis na biblioteca ([http://www.decom.ufop.br/gustavo/Bacia\\_do\\_Barreiro/Bacia\\_do\\_Barreiro.htm](http://www.decom.ufop.br/gustavo/Bacia_do_Barreiro/Bacia_do_Barreiro.htm)), consideradas por Silva et al. (2014).



Avaliou-se o número de motoristas, número de tarefas realizadas por cada motorista, tempo total ocioso e tempo de hora extra e a porcentagem que o motorista tem para realizar a máxima produtividade.

Os valores apresentados na Tabela 3 são de alguns motoristas da primeira empresa no dia de sexta-feira.

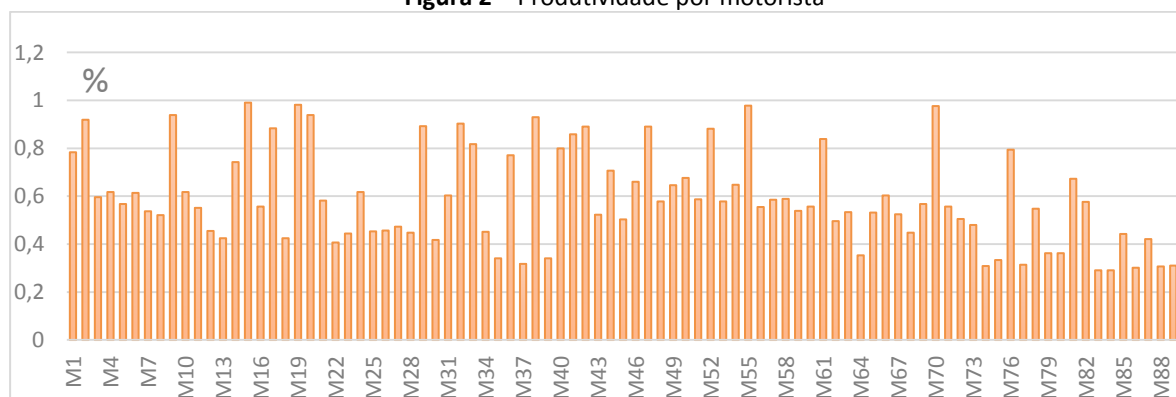
Motoristas	Numero de Tarefas	Tempo Ocioso (min)	Tempo Trabalhado (min)	%	Horas Extras (min)	Status
M1	3	7	439	0.78	0	2
M2	3	11	515	0.91	75	3
M3	3	13	334	0.59	0	1
M4	2	5	346	0.61	0	2
M5	1	0	318	0.56	0	2
M6	2	6	344	0.61	0	2
M7	1	0	301	0.53	0	2
M8	2	21	292	0.52	0	1
M9	3	3	526	0.93	86	3
M10	2	8	346	0.61	0	2
M11	2	6	309	0.55	0	2
M12	1	0	255	0.45	0	2
M13	1	0	238	0.42	0	2
M14	3	16	416	0.74	0	2
M15	3	40	555	0.99	115	3
M16	1	0	312	0.55	0	2
M17	3	76	495	0.88	55	3
M18	2	8	238	0.42	0	2
M19	3	21	550	0.98	110	3
M20	4	32	526	0.93	86	3

Fonte: Elaborado pelo autor (2019).

**Tabela 3** – Representação de uma escala.

Observa-se na Figura 3 que todos os motoristas abaixo de a 0.42 estão na jornada meia e os demais são jornadas interias. Além disso, os motoristas que estão acima de 0.784 já começam realizar horas extras.

**Figura 2** – Produtividade por motorista



Fonte: desenvolvido pelo autor (2019).

Neste exemplo temos 89 motoristas em total, 19 de jornadas meias e 70 de jornadas inteiras. Mas das 70 de jornadas inteiras, 19 motoristas estão fazendo horas extras. Ressalta-se que todas as tarefas foram alocadas.

O propósito de nosso projeto é diminuir as jornadas de trabalhos, significa que se jornadas inteiras atendem mais horas extras vamos ter menos jornadas de trabalho. Analisando os dados do gráfico da Figura 2, observa-se que ainda existe margem de melhora.

Tipo de jornada	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
Inteira	119	115	138	151	141	108	57
Pegada Dupla	18	19	20	17	16	9	7

Fonte: Elaborado pelo autor (2019).

**Tabela 4** – Análise das escalas utilizadas por Silva et al. (2014).

Para fazer uma comparação dos resultados obtidos com os resultados obtidos por Silva et al. (2014), calculou-se uma média de todas as empresas de cada dia da semana que nos foram fornecidos, já que nos trabalhos de Silva et al. (2014) não está especificado qual empresa em particular foi considerada para os resultados apresentados.

Tipo de jornada	Segunda	Sexta	Sábado	Domingo
Inteira	76	81	54	35
Meia Jornada	20	16	8	5

Fonte: Elaborado pelo autor (2019).

**Tabela 5** – Análise das escalas utilizadas do autor

A se analisar as Tabelas 1, 2 e 3, observa-se que o número médio de motoristas necessários para atender todas as tarefas ficou menor em relação ao utilizado pela empresa e às soluções por Silva et al. (2014).

## 5. Considerações Finais

O modelo proposto contribui no sentido de oferecer ferramentas que podem ser utilizadas na elaboração e utilização de técnicas organizacionais de planejamento.

A comparação dos resultados obtidos com a aplicação do modelo com jornadas inteiras e jornadas meias indica que pode ter uma maior redução nos números de motoristas, cujo impacto financeiro implicaria significativamente nos custos totais de uma empresa de transporte público, o PPT ganha uma importância muito maior no planejamento operacional do Sistema de Transporte Público. A otimização das escalas de trabalho afeta tanto as empresas operadoras quanto os usuários, pois com determinada economia gera-se a possibilidade de um maior investimento na qualidade do transporte público e de redução do valor das tarifas. Com isso mais usuários serão atraídos ao serviço, diminuindo o uso de veículo particular e ajudando o meio ambiente.

O método proposto determina soluções iniciais para aplicação de um processo de melhoria e consegue realizar a troca do sequenciamento das tarefas a serem cumpridas com a intenção de minimizar o tempo ocioso entre as mesmas. Acredita-se que aplicando o método

proposto junto a outro método heurístico de melhoria, os resultados podem determinar escalas otimizadas com valores menores em relação ao custo. Sugere-se que o método de melhoria seja aplicado visando minimizar o número de motoristas escalados, uma vez que diminuir os custos salariais gerariam maiores economias para a empresa.

Para dar continuidade ao presente trabalho pode-se sugerir a integração da escala de tripulantes com a escala de veículos, pois nem sempre os veículos disponíveis estão aptos a realizar as viagens. Então, torna-se necessário que a oficina de manutenção saiba com antecedência quais veículos têm prioridade na manutenção em função das viagens programadas.

## Referências

- DANTZIG, G. B.; RAMSER, J. H. The truck dispatching problem. **Management Science**. v. 6, p. 80-91, 1959.
- FESTA, E. L.; NOGUEIRA, C. W. ; TAGLIALENHA, S. L. S. Otimização de escalas de tripulação de transporte público urbano com a meta heurística variable neighborhood search. In: Darly Fernando Andrade. (Org.). **Métodos Quantitativos - Pesquisa Operacional**. 1ed. Belo Horizonte: Poisson, v. 3, p. 210-222, 2018.
- FORES, S.; PROLL, L.; WREN, A. An improved ILP system for driver scheduling. In: Wilson, N. H. M. (Ed.) **Computer-Aided Transit Scheduling**. Montreal, p. 43-61, 1999.
- FRIBERG, C.; HAASE, K. An exact branch and cut algorithm for the vehicle and crew scheduling problem. In: Wilson, N. H. M. (Ed.) **Computer-Aided Transit Scheduling**. p. 63-80. Montreal: Springer, 1999.
- HILLIER, F.S.; LIEBERMAN, G. J. **Introdução à pesquisa operacional**. McGraw Hill Brasil, 2013.
- LESSARD, R.; ROSSEAU, J. M.; DUPUIS, D. **Hastus I: A mathematical programming approach to the bus driver scheduling problem**. Amsterdam: Elsevier, 1981.
- LI, J.; KWAN, R. S. A fuzzy genetic algorithm for driver scheduling. *European Journal of Operation Research*, 147(2), 2003.
- LORENA, L. A. N. **A new hybrid heuristic for driver scheduling**. *International Journal of Hybrid Intelligent Systems*. Amsterdam, v. 4, p. 39-47, 2007.
- LAU, H.C. On the complexity of manpower shift scheduling. **Computers and Operations Research**, 23 (1), pp. 93-102, 1996.
- LOURENÇO, H. R.; PAIXÃO, J. M. P.; PORTUGAL, R. Multiobjective metaheuristics for the bus driver scheduling problem. **Transportation Science**. Barcelona, v. 35, n. 3, p. 331-343. 2001.
- MESQUITA, M.; et al. A new model for the integrated vehicle-crew-rostering problem and a computational study on rosters. **Journal of Scheduling**, 14(4), 2011.
- NURMI, K.; KYNGAS, J.; POST, G. Driver rostering for bus transit companies. **Engineering Letters**, 19 (2), 2011.
- SANTOS, A. G.; MATEUS, G. R. Crew scheduling urban problem: an exact column generation approach improved by a genetic algorithm. **IEEE Congress on Evolutionary Computation**. Cingapura, p. 1725-1731. 2007.
- SMITH, B. M.; WREN, A. A bus crew scheduling system using a set covering formulation. **Transportation Research**. Amsterdam, v. 22A, p. 97-108. 1988.

SILVA, G. P.; REIS, A. F. da S. A study of different metaheuristics to solve the urban transit crew scheduling problem. **Journal of Transport Literature**. Ouro Preto, v. 8, n. 4, p. 227–251, 2014.

SMITH, B. M.; WREN, A. A bus crew scheduling system using a set covering formulation. **Transportation Research**. Amsterdam, v. 22A, p. 97-108, 1988.

SZABO, Jaroslav. Comparison of Methods for Generating Initial Solution for Simulated Annealing. **Central European Researchers Journal**, [s.i.], v. 2, n. 1, p.37-41, 2016.

XIE, LIN et al. Metaheuristics approach for solving personalized crew rostering problem in public bus transit. **Journal of Heuristics**. [s.i.], p. 321-334. 2017.

SHEN, Y.; KWAN, R. S. K. Tabu search for driver scheduling. In: S. Voss, J.R. Daduna (Ed.), **Computer-Aided Scheduling of Public Transport**. Montreal: Springer p.121–135, 2001.