

08 a 11 de Outubro de 2018
Instituto Federal Fluminense
Búzios - RJ

OTIMIZAÇÃO DE TRIPULAÇÕES EM ÔNIBUS URBANO: UM MÉTODO HÍBRIDO UTILIZANDO PROGRAMAÇÃO COM RESTRIÇÕES E ALGORITMOS GENÉTICOS

Me. João Ricardo Diniz Amaral¹ – joao@qualitetecnologia.com.br

Prof. Dr. Nilton Alves Maia² – nilton.maia@unimontes.br

Prof. Dr. Narciso Ferreira dos Santos Neto³ – narciso_santos1@yahoo.com.br

^{1,2,3} Universidade Estadual de Montes Claros – Montes Claros, MG, Brazil

Resumo. O Problema da Programação de Tripulações objetiva gerar escalas de trabalho a partir de um conjunto de viagens pré-estabelecidas e escalonar uma determinada tripulação para cumprir estas viagens. Solucionar o problema consiste na geração de escalas com o menor custo operacional possível, atendendo as diversas restrições do problema. Encontrar a melhor escala, diminuindo a quantidade de horas extras, tempo ocioso e tripulação necessária para cumprir um horizonte de planejamento em um tempo computacional aceitável é uma tarefa de difícil solução. O presente trabalho teve por objetivo propor um Método Híbrido para Otimização da Escala de Tripulações em Ônibus Urbano, utilizando Programação com Restrições e Algoritmos Genéticos. Os testes foram realizados com dados operacionais reais de uma empresa de transporte público da cidade de Montes Claros (MG). O método proposto obteve resultados positivos, gerando escalas de jornada menos custosas do que as utilizadas pela empresa.

Palavras-chave: Problema da Programação de Tripulações. Otimização Computacional no Transporte Público. Métodos Híbridos.

1. INTRODUÇÃO

O transporte público por ônibus urbano é de fundamental importância nos centros urbanos. Leite (2012) salienta que as pessoas estão se concentrando cada vez mais nas grandes cidades, e estima que em duas décadas 80% da população urbana do planeta estará concentrada em grandes centros populacionais, nos países desenvolvidos. Diante deste cenário, otimizar o transporte público urbano é essencial, garantindo a qualidade da mobilidade urbana dessa massa populacional.

Para analisar um sistema de transporte de passageiros é importante levar em consideração os requisitos de atendimento aos horários programados, realização das paradas

programadas nos pontos e atendimento da demanda de passageiros da melhor forma, respeitando as restrições de disponibilidade da tripulação (motorista e cobrador) e frota.

Uma das fases do processo de planejamento do transporte público trata do Problema da Programação de Tripulações (PPT). Prates e Silva (2014) afirmam que nesta etapa busca-se obter sequências de jornadas diárias, compreendendo os dias úteis, sábados e domingos, respeitando às restrições legais e trabalhistas dos colaboradores.

Programar tripulações consiste no processo de construir escalas e jornadas de trabalho, satisfazendo necessidades e restrições impostas por empresas e usuários do sistema. CUNHA (2014) afirma que em várias organizações as pessoas envolvidas no desenvolvimento de rodízios precisam de ferramentas de suporte para prover os funcionários corretos nos horários certos. Normalmente utiliza-se planilhas, bancos de dados, possíveis ferramentas de rodízios, modelos matemáticos e algoritmos apropriados.

Para o planejamento no transporte público urbano por ônibus, as escalas são criadas com base nas tabelas de horários. Diferentemente do setor aéreo, as escalas são menores e também normalmente não se aplica o conceito de início e fim da jornada ser obrigatório no mesmo lugar. O objetivo é encontrar um conjunto com uma sequência de jornadas diárias com menor custo e atenda as necessidades da tabela de horários em um horizonte de planejamento (CUNHA, 2014). Dentro do fluxo diário de jornadas existem as jornadas simples e as jornadas dupla pegada. Esta última se diferencia por ser realizada quando ocorre com um intervalo entre duas horas ou mais entre uma jornada e outra.

2. OBJETIVOS E PARÂMETROS DE ENTRADA

Os objetivos deste trabalho foram minimizar o número de tripulações necessárias para cumprir todas as jornadas, minimizar a quantidade total de horas extras e ociosas da tripulação, equilibrar a carga de trabalho entre a tripulação, maximizar o número de duplas pegadas possível e minimizar o tempo de processamento do horizonte de planejamento. Para o desenvolvimento do método proposto foram definidos parâmetros de entrada gerais, comuns ao cotidiano de empresas de transporte público urbano, sendo:

- 1) O horizonte de planejamento é iniciado em uma segunda feira.
- 2) Todas as tripulações são alocadas com um intervalo de descanso de 11 horas entre dias seguidos de jornadas.
- 3) Cada tripulação folga pelo menos 1 domingo durante o horizonte de planejamento.
- 4) 6 é o número máximo de dias consecutivos de trabalho sem folga (sistema de folga variável).
- 5) A tripulação que realiza dupla pegada corresponde a no máximo 12% da tripulação total disponível.
- 6) As jornadas são apenas do tipo pegada simples (principal) e dupla pegada (quando necessário).
- 7) A jornada semanal é de 44 horas.
- 8) A jornada diária máxima é de 7 horas e 20 minutos.
- 9) 2 horas é o máximo de horas extras por dia por tripulação.
- 10) O intervalo mínimo entre jornadas diárias é de 11 horas.

3. SIMULADOR DE ALOCAÇÃO DE TRIPULAÇÕES (SAT)

A primeira etapa do trabalho consistiu em desenvolver o Simulador de Alocação de Tripulações (SAT). Os objetivos do simulador consistem em gerar escalas diárias válidas utilizando o mínimo de tripulações, minimizar o tempo de processamento para gerar uma

escala válida para o dia, minimizar a quantidade de horas extras geradas pela tripulação e Maximizar o número de duplas pegadas, não ultrapassando o limite permitido para o dia. Os parâmetros do SAT basearam-se nas principais características das jornadas diárias utilizadas nos modelos de rodízio das tripulações (Mayrink e Silva, 2013), conforme a Tabela 1:

Tabela 1: Parâmetros de entrada para o SAT.

| Parâmetro | Descrição | Valor Fixado |
|-------------------------|--|--------------|
| <i>execTime</i> | Tempo de execução do algoritmo para geração de um dia de alocações válidas do horizonte de planejamento. | - |
| <i>jDiaria</i> | Tempo máximo de trabalho diário de uma tripulação (em minutos). | 440 |
| <i>limDuplaPegada</i> | Porcentagem máxima da tripulação que pode fazer dupla pegada por dia. | 12% |
| <i>minEntreJornadas</i> | Mínimo de tempo do intervalo entre jornadas em dias diferentes (em minutos). | 660 |
| <i>limHoraExtra</i> | Limite que falta na jornada para completar com hora extra (em minutos). | 120 |
| <i>maxHoraExtra</i> | Máximo de horas extras que uma tripulação pode fazer por dia (em minutos). | 120 |
| <i>qTrip</i> | Tripulações disponíveis no horizonte de planejamento (incrementado pelo simulador até atingir um valor válido do planejamento diário). | 1 |
| <i>tEntrePegadas</i> | Tempo do intervalo entre uma pegada e outra na dupla pegada diária (em minutos). | 60 |
| <i>tolerancia</i> | Tempo mínimo restante na jornada para finalizá-la com uma pegada simples. | 395 |

Posteriormente, definiram-se as restrições de domínio, utilizadas para decidir se uma tripulação candidata pode ou não ser alocada para cumprir uma jornada da vez em um dia específico. São de natureza chave do problema e não podem ser violadas, devendo ser respeitadas a cada alocação de tripulação a uma jornada. A Função 1 sintetiza todas as restrições de domínio e será utilizada em cada função objetivo do simulador:

$$aloc(trip, jorn, dia) \quad (1)$$

$$\begin{aligned}
 \text{sujeito a} \quad & trip_{dia} = jorn_{dia}; \\
 & trip_{linha2} \langle \rangle \emptyset; \\
 & trip_{tdisp} > jorn_{medio} \vee jorn_{rest} \leq jorn_{medio}; \\
 & trip_{diaAnterior} \cdot Fin + \gamma_{inEntreJornadas} < jorn_{ini}; \\
 & trip_{folga} = false; \\
 & trip_{trab} < 6; \\
 & trip_{tocup1} > jorn_{medio} \vee jorn_{rest} \leq jorn_{medio}; \\
 & jorn_{ini} \leq trip_{inicial} - tEntrePegadas; \\
 & jorn_{ini} \geq trip_{final} + tEntrePegadas; \\
 & trip_{tocup2} > jorn_{medio}; \\
 & qDuplaPegada_{dia} \leq limDuplaPegada; e \\
 & trip_{tExtra} \leq maxHoraExtra.
 \end{aligned}$$

Restrições de desempenho também foram adicionadas, buscando otimizar o resultado do processamento diário das jornadas. A Tabela 2 define cada restrição e o seu objetivo no método.

Tabela 2: Restrições de desempenho

| Restrição | Descrição | Objetivo |
|-----------|--|-------------------------------|
| 1. | Pré-processar o arquivo de jornadas, ordenando-as em linhas / carros / duração (maior para menor). | $min(execTime)$ |
| 2. | Categorizar as jornadas sequencialmente em alocação normal / alocação invertida. | $min(execTime)$ |
| 3. | Preencher os tempos nas jornadas que são iguais ou maiores que o tempo disponível de alocação diário de cada tripulação com pegada simples. | $min(qTrip)$ |
| 4. | Preencher os tempos nas jornadas que são maiores que o tempo disponível de alocação diário de cada tripulação menos o tempo de tolerância com pegada simples. | $min(qTrip)$ |
| 5. | Finalizar as jornadas com tempo restante menor ou igual ao limite de hora extra diário para cada tripulação com horas extra da tripulação já alocada para esta jornada. | $min(tExtra1+tExtra2)$ |
| 6. | Trocar na tabela de alocações as tripulações que foram alocadas como pegada simples e não tem escalação no dia anterior por tripulações que possuem escala no dia anterior e os horários não conflitam com a escalação do dia atual pra facilitar as duplas pegadas. | $min(qTrip)$ |
| 7. | Preencher o limite máximo de duplas pegadas com a mesma tripulação, em uma jornada normal e outra invertida. | $max(qDupla)$ $min(qTrip)$ |

No procedimento de alocação invertida, ao invés da jornada ser alocada em tempo sequencial crescente a alocação é feita em tempo sequencial decrescente em jornadas alternadas, diminuindo assim a quantidade de conflitos entre horários de uma mesma tripulação e facilitando a ocorrência das duplas pegadas.

Com a definição das restrições, foram utilizadas funções objetivo em cada etapa do SAT, de acordo com o tipo de alocação da tripulação. Para alocar uma tripulação com pegada simples, o algoritmo faz primeiramente o escalonamento das jornadas que são caracterizadas com pegada simples, buscando utilizar ao máximo o tempo de trabalho de uma tripulação no dia, conforme demonstrado na Função 2:

$$Min (trip_{td\ disp}(aloc(trip, jorn, dia)) \quad (2)$$

$$\begin{aligned} \text{sujeito a } & trip_{linha1} = \emptyset; \\ & jorn_{rest} \geq jDiaria; e \\ & jorn_{rest} \geq tolerancia. \end{aligned}$$

Caso não consiga finalizar o tempo integral da jornada com alocações com pegada simples (Função 2), o SAT parte para a alocação de uma mesma tripulação com pegada dupla,

lidando com o restante das jornadas ainda não alocadas, que são fracionadas, onde os tempos são menores que os períodos diários integrais das tripulações, diminuindo ao máximo o tempo restante para alocação em cada jornada, conforme a Função 3:

$$\begin{aligned} & \text{Min } (jornA_{rest}, jornB_{rest} (aloc(trip, jornA, dia)aloc(trip, jornB, dia))) \quad (3) \\ \text{sujeito a } & trip_{linha1} = \emptyset; \\ & trip_{linha2} = \emptyset; e \\ & jornA_{rest} + jornB_{rest} \leq jDiaria. \end{aligned}$$

Caso ainda não tenha sido possível finalizar a alocação da jornada, ocorre a alocação de uma tripulação distinta com pegada dupla, respeitando o limite de duplas pegadas para o dia, maximizando a ocorrência de duplas pegadas possível no dia, de acordo com a Função 4:

$$\begin{aligned} & \text{Max } qDupla(aloc(tripA, jorn, dia), aloc(tripB, jorn, dia)) \quad (4) \\ \text{sujeito a } & trip_{linha1} \langle \rangle \emptyset; \\ & trip_{linha2} = \emptyset; e \\ & jorn_{rest} \leq trip_{disp}. \end{aligned}$$

Por fim, ocorre o processamento completo de um dia de jornadas, o qual consiste em alocar todas as jornadas daquele dia às tripulações candidatas, respeitando as restrições de domínio do problema, até que não sobre tempo restante nas jornadas para ser alocado, buscando utilizar o mínimo de tripulações possível, demonstrado na Função 5:

$$\begin{aligned} & \text{Min } qTrip(T, J, dia) \quad (5) \\ \text{sujeito a } & \forall j \in J: aloc(trip, j, dia); e \\ & \forall j \in J: j_{rest} > 0. \\ \text{em que } & T: \text{conjunto de tripulações disponível para o dia;} \\ & j: \text{uma jornada;} e \\ & J: \text{conjunto de todas as jornadas a serem cumpridas no dia.} \end{aligned}$$

Após realizar a alocação completa para um dia de jornadas, é possível avaliar o tempo de execução gasto para gerar a escala do dia correspondente, através da Função 6:

$$\begin{aligned} & tpDia = \frac{100}{qTrip} * execTime(T, J, dia) \quad (6) \\ \text{sujeito a } & qTrip \geq 1; \\ & \forall j \in J: aloc(trip, j, dia); e \\ & \forall j \in J: j_{rest} > 0. \\ \text{em que } & tpDia: \text{tempo de execução para alocar um dia de jornadas;} \\ & T: \text{conjunto de tripulações disponível para o dia;} \\ & qTrip: \text{quantidade de tripulação utilizada para escalonar o dia;} \\ & execTime: \text{tempo de processamento para alocação completa do dia;} \\ & j: \text{uma jornada;} e \\ & J: \text{conjunto de todas as jornadas a serem cumpridas no dia.} \end{aligned}$$

A partir do resultado da Função 6, é possível calcular para cada dia do horizonte de planejamento o tempo de execução total do algoritmo, conforme a Função 7:

$$tpTotal = \left(\sum_{d=0}^{qDias} tpDia \right) \quad (7)$$

sujeito a $qDias \geq 1$;

em que $tpTotal$: tempo de execução para alocar todas as jornadas em todos os dias no horizonte de planejamento; e

$tpDia$: tempo de execução para alocar um dia de jornadas.

A partir da modelagem elucidada, foi criado o Algoritmo SAT, que retorna uma matriz válida preenchida com horários de alocação de tripulações para um dia de jornadas, respeitando as restrições de domínio e com todas as restrições de desempenho ativadas.

4. ALGORITMO GENÉTICO APLICADO AO SAT (AG-SAT)

O SAT resulta em uma solução determinística, gerando uma solução válida para um horizonte de planejamento, porém a qualidade da solução obtida depende da ordem em que as tripulações são processadas a cada dia, visto que a posição em que cada tripulação é processada e escalonada por dia resulta em diferentes soluções. Explorar a quantidade total de combinações possíveis no modelo com uma solução exata se mostrou computacionalmente inviável. A Função 8 representa a quantidade de combinações possíveis:

$$qTrip!^{qDias} \quad (8)$$

em que $qTrip$: quantidade disponível de tripulações para alocação.

Sendo assim optou-se por complementar o método acrescentando uma heurística de otimização estocástica, aplicando um Algoritmo Genético utilizando codificação por permutação e combinado com o simulador desenvolvido. Os objetivos desta etapa se aplicam a cada dia do horizonte de planejamento completo, buscando minimizar o tempo ocioso e quantidade de tripulação necessários para cumprir o horizonte de planejamento e equilibrar a distribuição das horas extras, ociosas e dos dias trabalhados pelas tripulações.

Para calcular a aptidão de uma solução, o algoritmo genético é aplicado a cada dia do horizonte de planejamento gerado pelo SAT, antes do processamento do próximo dia, testando sistematicamente as opções de combinações possíveis e mantendo o melhor resultado diário obtido, conforme a Função 9:

$$Min g = \left(\sum_{i=0}^{qTrip} tdisp * \frac{100}{tdisptot} + \sum_{i=0}^{qTrip} tExtra * \frac{100}{textratot} \right) \quad (9)$$

sujeito a $g \in G$

em que G : conjunto de soluções válidas para um dia de alocação;

$qTrip$: quantidade de tripulação utilizada;

$tDisp$: tempo disponível de cada tripulação;

$tDispTot$: tempo disponível total da tripulação;

$tExtra$: horas extras realizadas por cada tripulação; e

$tExtraTot$: horas extras totais da tripulação.

O algoritmo genético também equilibra a quantidade de dias trabalhados e folgas entre a tripulação no final do processamento do horizonte de planejamento através da realização do rodízio das tripulações escalonadas.

Conforme proposto nesta pesquisa, foi criado o Algoritmo AG-SAT (Algoritmo Genético Aplicado ao Simulador de Alocação de Tripulações), incorporando o algoritmo genético proposto ao SAT, retornando a melhor matriz válida atingida para o dia, de acordo com a Função 9, preenchida com horários de alocação de tripulações para um dia de jornadas, respeitando as restrições de domínio e com todas as Restrições de Desempenho ativadas.

5. RESULTADOS OBTIDOS

Para o processamento e execução do método foi coletada uma base de dados funcional de jornadas, cedida por uma empresa de Transporte Público da cidade de Montes Claros (MG). Na Tabela 3 tem-se as características do banco de dados de jornadas utilizado nos testes:

Tabela 3: Quantidades totais de jornadas utilizadas nos testes.

| Dia | Jornadas | Linhas | Carros | Duração (horas) |
|--------------|------------|-----------|------------|-----------------|
| Dias úteis | 325 | 18 | 64 | 5.160 |
| Sábado | 53 | 17 | 50 | 724 |
| Domingo | 30 | 18 | 30 | 453 |
| TOTAL | 408 | 53 | 144 | 6337 |

Arquivo de jornadas (empresa de transporte público urbano de Montes Claros, 2017).

Os testes foram feitos gerando variações dos Algoritmos SAT e AG-SAT, analisando o comportamento de ambos, em um horizonte de planejamento de 49 dias. Na Tabela 4 são exibidos resultados aplicando o método proposto completo, ou seja, o AG-SAT.

Tabela 4: Desempenho do Método Híbrido Completo

| Parâm. | Q. trip. | Horas extras | Horas ociosas | Dupla pegada | Q. escalas. | Média de trip. não utilizada p/ dia | Tempo de process. |
|--|----------|--------------|---------------|--------------|-------------|-------------------------------------|-------------------|
| Pop: 10 Ger: 5 Cruz: 70% Mut: 2% Elit: Não Exec: 10 | 156 | 650h04 | 817h16 | 847/931 | 6048/6594 | 33/156 | 1442,63s |
| Pop: 10 Ger: 5 Cruz: 70% Mut: 2% Elit: Sim Exec: 10 | 157 | 650h04 | 839h06 | 847/931 | 6048/6594 | 34/157 | 1595,35 |

Os resultados mostrados na Tabela 4 foram os melhores colhidos após diversos experimentos com a variação dos parâmetros do AG-SAT. O aumento da quantidade da população ou gerações influiu em até 30% o tempo de processamento, não impactando na qualidade dos resultados finais. A diminuição das taxas de cruzamento aumentou a quantidade de tripulações necessária para cumprir o horizonte de planejamento em até 10%, devido à baixa rotatividade das tripulações e acúmulo de tripulações com folga no mesmo dia.

O aumento da taxa de mutação influenciou também o tempo de execução, aumentando em até 25% a convergência do algoritmo.

5.1 Tempo de Processamento

A Figura 1 detalha o tempo de processamento de cada dia do horizonte de planejamento pesquisado no melhor cenário do AG-SAT (Tabela 4, primeira linha).

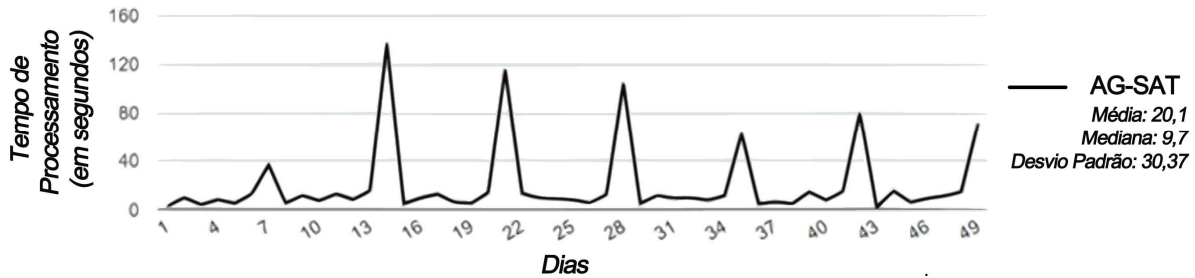


Figura 1: Tempo de processamento de cada dia no AG-SAT.

As informações de processamento do SAT foram omitidas, visto que o tempo para cada dia de processamento deste foi inferior a 1 segundo. Nota-se picos de processamento são percebidos a cada sete dias do horizonte de planejamento. Isto se deve ao fato da distribuição das folgas da tripulação ocorrer nestes dias, sendo que cada tripulação tem pelo menos uma folga aos domingos durante o período, além da necessidade de folgas a cada seis dias consecutivos de trabalho, conforme as restrições de domínio do problema.

5.2 Ocupação das Tripulações

Nota-se na Figura 2 que o objetivo de equilibrar os dias trabalhados de cada tripulação foi alcançado.

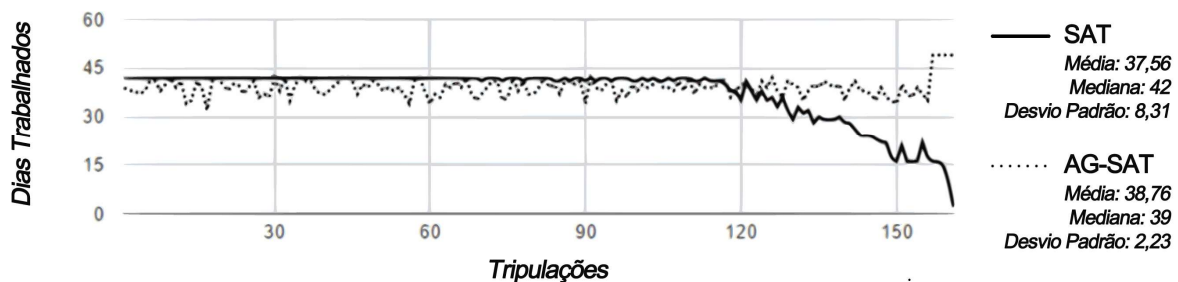


Figura 2: Equilíbrio dos dias trabalhados da tripulação (elaborado pelo autor).

É possível perceber que o SAT ao fazer a alocação sempre com a mesma sequência da tripulação utiliza as últimas tripulações apenas ao final do horizonte, somente quando é necessário gerar uma escala válida. Já o AG-SAT mantém essa variação durante todo os dias, equilibrando a carga diária de cada tripulação.

3.3 Horas Extras da Tripulação

A quantidade das horas extras por tripulação foi minimizada pelo SAT e se manteve igual no AG-SAT, porém percebe-se na Figura 3 que no AG-SAT equilibrou-se a quantidade de horas extras da tripulação.

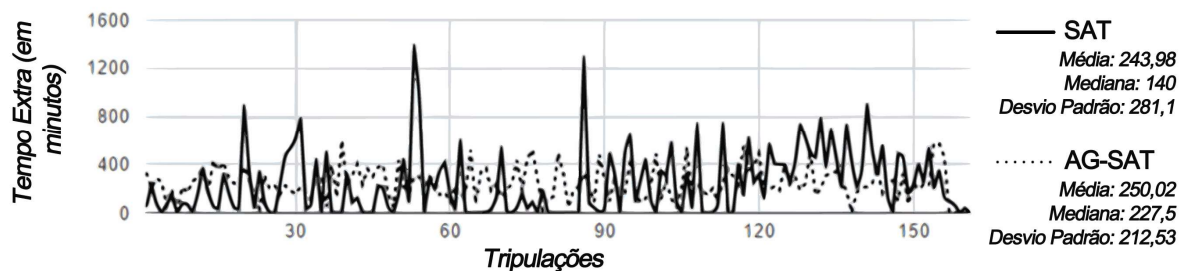


Figura 3: Minimização e equilíbrio das horas extras da tripulação (elaborado pelo autor).

Enquanto que no SAT algumas tripulações tiveram uma alta quantidade de horas extras, no AG-SAT os resultados para cada tripulação tiveram pouca variação.

3.4 Horas Ociosas da Tripulação

A quantidade das horas ociosas por tripulação foi minimizada no AG-SAT, conseqüência também da minimização da quantidade de tripulações necessária para cumprir o horizonte de planejamento, como pode ser conferido na Figura 4:

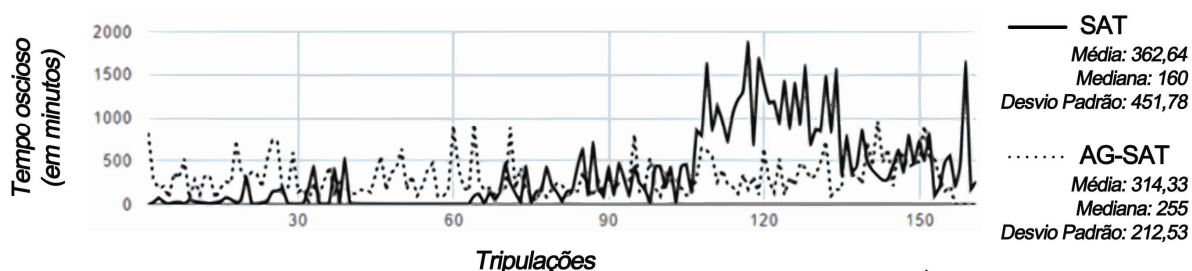


Figura 4: Minimização e equilíbrio das horas ociosas da tripulação (elaborado pelo autor).

Também pode-se notar na Figura 4 que no AG-SAT equilibrou-se a quantidade de horas ociosas da tripulação.

3.5 Comparativo com um cenário real

Na Tabela 5 é possível avaliar os resultados obtidos comparando com um cenário real de alocação utilizado na empresa que cedeu os dados das jornadas (Tabela 3) para esta pesquisa. Foi considerada pelo menos uma folga no domingo dentro do horizonte de planejamento.

Tabela 5: Comparação dos resultados com um cenário real.

| Fonte | Dias | Q. Trip | Horas extras | Horas ociosas | Compensação (HO – HE) | Tempo de processamento |
|---------------|------|---------|--------------|---------------|-----------------------|------------------------|
| AG-SAT | 30 | 156 | 398h19 | 724h21 | + 326h02 | 1388,98s |
| SAT | 30 | 171 | 398h19 | 704h51 | + 306h32 | 15,90s |
| Cenário Atual | 30 | 173 | 1863h44 | 1492h04 | - 371h40 | - |

4. CONCLUSÕES

Este documento teve como principal objetivo propor um método híbrido para otimização da escala de tripulações em ônibus urbano, realizando o escalonamento da

tripulação, atendendo as restrições do problema da programação de tripulações, com os objetivos de minimizar o número de tripulações necessária para cumprir todas as jornadas, minimizar a quantidade total de horas extras e ociosas da tripulação, maximizar o número de duplas pegadas possível, equilibrar a carga de trabalho entre a tripulação e minimizar o tempo de processamento do horizonte de planejamento.

A resolução do PPT desta pesquisa demandou resolver diversas restrições dependentes entre si. Uma das maiores dificuldades foi no desenvolvimento do SAT. A criação deste simulador foi necessária para analisar em um ambiente controlado o comportamento de uma tripulação diante de um horizonte de planejamento com horários pré-definidos.

Neste processo percebeu-se que a otimização do tempo computacional gasto para execução do método seria fundamental para a sua aplicabilidade prática, tendo tanta importância quanto à otimização das variáveis do problema. O modelo apresentado pode ser utilizado em diversos problemas de análise combinatória com restrições de horários conflitantes, pois as suas variáveis são maleáveis e podem ser modificadas de acordo com as restrições do problema.

REFERÊNCIAS

- AMARAL, J.R.D. *Otimização de Tripulações em Ônibus Urbano: um Método Híbrido utilizando Programação com Restrições e Algoritmos Genéticos*. Dissertação de Mestrado. Universidade Estadual de Montes Claros – Unimontes. Disponível em: <https://goo.gl/CvJepX> Acesso em: 13/05/2018.
- CUNHA, C. B. *Aspectos práticos da aplicação de modelos de roteirização de veículos a problemas reais*. Revista Transportes da ANPET – Associação Nacional de Pesquisa e Ensino em Transportes, v. 8, p. 51-74, nov. 2014.
- LEITE, C.; AWAD, J. C. M. *Cidades sustentáveis cidades inteligentes: desenvolvimento sustentável num planeta urbano*. Porto Alegre: Bookman, 2012.
- MAYRINK, V. T. M., SILVA, G. P. *Optimization of crew rostering assignment for public transport systems*. *Journal of Transport Literature*. DOI: 10.1590/S2238-10312013000300009, v. 7 (2013), p. 192-213.
- PRATES, R. F. C., SILVA, G. P. *Otimização da Escala Mensal de Motoristas de Ônibus Urbano Utilizando a Heurística Variable Neighborhood Search*. Revista Transportes, v. 22, nº1, (2014), p. 31–43.

Abstract. *The crew scheduling problem aims to generate workloads from a set of pre-established trips and to schedule a particular crew to fulfill these trips. Solving the problem consists of the generation of scales with the lowest operational cost possible, taking into account the different constraints of the problem. Finding the best scale by decreasing the amount of overtime, downtime, and crew required to meet a planning horizon in an acceptable computational time is a difficult task. The present work had the objective of proposing a Hybrid Method for Optimization of the Scale of Crews in Urban Bus, using Programming with Restrictions and Genetic Algorithms. The tests were performed with real operational data from a public transport company in the city of Montes Claros (MG). The proposed method obtained positive results, generating less costly time scales than those used by the company.*

Keywords: *Crew Programming Problem. Computational Optimization in Public Transport. Hybrid Methods.*