

Estudo da Influência de *Buffers* em Linhas de Modelo Misto com Auxílio de Programação Linear Inteira Mista e Simulação

Julio Alberto Godofredo (UTFPR) julio.a.godofredo@gmail.com

Leandro Magatão (UTFPR) magatao@utfpr.edu.br

Resumo:

O presente artigo aborda um estudo de balanceamento e sequenciamento em uma linha de montagem de modelo misto (*Mixed Model Assembly Line*) com *buffers* de armazenagem temporária em linha. O balanceamento foi realizado por meio de um modelo matemático em Programação Linear Inteira Mista (PLIM) implementado no ambiente IBM ILOG CPLEX Optimization Studio. Os estudos de sequenciamento foram desenvolvidos a partir de simulação implementada em VBA (*Visual Basic for Applications*). Objetivou-se analisar a influência do tamanho e posição de *buffers* intermediários na produtividade da linha. O estudo de caso de linha de montagem de bancos de automóveis apresentado por Donnini *et al.* (2010) foi utilizado para condução dos experimentos. Em um estágio inicial o modelo PLIM proposto determina o balanceamento otimizado para a linha produtiva, já considerando a presença de um *buffer* intermediário na linha sem, contudo, tecer considerações acerca da dimensão (tamanho) do *buffer* em análise. Na sequência, fase de simulação, observam-se as diferenças do tempo de ciclo (*tc*) obtido pelo modelo PLIM em relação ao *tc* obtido na simulação. Estas diferenças ocorrem em função do posicionamento e capacidade dos *buffers* adotados na fase de simulação. Os dados extraídos da simulação auxiliam na compreensão do funcionamento da linha de modelo misto em relação às diferentes sequências de produtos a serem produzidas. A combinação balanceamento-simulação possibilitou a análise de condições otimizadas de operação da linha em estudo.

Palavras chave: Balanceamento de Linhas, *Buffer*, Programação Linear Inteira Mista, Simulação.

Study of Buffers' Influence in Mixed Model Lines with the Aid of Mixed Integer Linear Programming and Simulation

Abstract:

The present paper approaches a balancing and sequencing study in a Mixed Model Assembly Line with temporary storage buffers. The balancing was performed by means of a mathematical model of Mixed Integer Linear Programming implemented on IBM ILOG CPLEX Optimization Studio environment. The sequencing investigations were developed from a simulation implemented on VBA (*Visual Basic for Applications*). The goal was to analyze the size and intermediary buffers position influence on line productivity. The case study of automotive seats assembly line presented by Donnini *et al.* (2010) was used to conduct the experiments. In an initial stage the proposed PLIM model determines the optimized line balancing, considering the presence of an intermediate buffer in line, without, however, make considerations on the dimension (size) of the analyzed buffer. In the simulation phase it was observed differences on cycle time (*ct*) obtained by PLIM model in comparison to *ct* obtained by simulation. These differences occur due to the simulation adopted buffers positioning and capacity. The simulation extracted data help understanding the Mixed Model Assembly Line functioning in relation to different goods sequences to be produced. The conjunction balancing-sequencing enabled analysis of assembly line optimized conditions.

Key-words: Line Balancing, Buffer, Mixed Integer Linear Programming, Simulation.

1. Introdução

Devido à globalização, cada vez mais as indústrias modernas necessitam buscar o aperfeiçoamento de suas linhas de montagem com a finalidade de aumentar a produtividade e, consequentemente, poder oferecer produtos a um preço mais competitivo. A tendência é o uso de linhas de montagem onde produtos com características diferentes compartilham a mesma linha produtiva, configurando-se uma Linha de Modelo Misto (*Mixed Model Assembly Line*). Dentre as possíveis formas de melhorias operacionais das linhas está a realização de um balanceamento mais eficiente e adoção de melhores configurações de sequenciamento de entrada dos produtos que são produzidos. Para tanto, conceitos abordados no estudo da Pesquisa Operacional (PO) são de grande valia no intuito de obtenção de soluções otimizadas. Dentre eles estão técnicas de otimização que utilizam uma abordagem denominada Programação Linear Inteira Mista (PLIM ou, do inglês, MILP – *Mixed Integer Linear Programming*). Existem diversos *softwares* especializados em resolver modelos de PLIM. Na literatura já foram propostos modelos de PLIM que aperfeiçoam o balanceamento e sequenciamento de linhas como em Becker e Scholl (2004) e Boysen *et al.* (2009). Porém existe uma gama de características operacionais ainda não abordadas nestes modelos. Além disso, a realização de simulação das linhas onde soluções otimizadas são propostas por modelos matemáticos é de fundamental importância para a validação das respostas obtidas com os modelos, visto que simplificações de modelagem podem ser adotadas. A junção de otimização com simulação pode, desta forma, fornecer *insights* ao pesquisador de como as linhas podem se comportar, de fato, em um modo otimizado. No presente trabalho é analisado um problema de balanceamento numa linha de bancos de automóveis contendo um *buffer* intermediário, cenário descrito por Donnini *et al.* (2010). A partir dos dados do problema real uma solução otimizada é obtida por um modelo PLIM e, na sequência, uma simulação a eventos discretos é realizada para auxiliar no dimensionamento de um *buffer* intermediário.

A sequência do artigo é organizada conforme a seguir descrito. Na seção 2 realiza-se uma breve revisão da literatura acerca de modelos PLIM para balanceamento da linha em estudo. Na seção 3 descreve-se o procedimento metodológico adotado para a realização do trabalho. A seção 4 apresenta os resultados obtidos. Na seção 5 apresentam-se as conclusões obtidas.

2. Revisão da literatura

2.1. Modelos MALBP

Os modelos atuais para realização de balanceamento de linhas de montagem simples são chamados de SALBP (*Simple Assembly Line Balancing Problem*), denominação primeiramente utilizada por Baybars (1986). Os modelos SALBP têm como objetivo diminuir o tempo de ciclo e/ou o número de estações, além de validarem balanceamentos pré-determinados.

Linhas de montagem onde mais de um modelo são produzidos são chamadas de linhas de modelo misto ou multimodelo. Nas linhas multimodelo as peças de um mesmo modelo são agrupadas e entram em blocos (lotes). Já nas linhas de modelo misto diferentes modelos podem ser sequenciados um após o outro, havendo relação direta com a filosofia de “produção puxada”. De acordo com Scholl (1999), a complexidade do balanceamento/sequenciamento de linhas de modelo misto é uma dificuldade a ser superada, dadas as prováveis diferenças de tempos de montagem dos diferentes produtos nos postos de trabalho. A Figura 1 ilustra como cada tipo de linha se comporta.

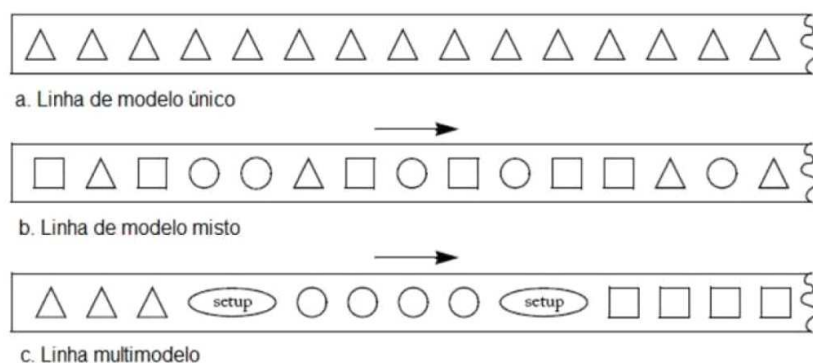


Figura 1 – Linhas de montagem para modelo único e múltiplos modelos
Fonte: Adaptado de Becker e Scholl (2004)

Uma variação dos modelos SALBP, chamada MALBP (*Mixed-model Assembly Line Balancing Problem*), foi criada para resolver problemas em linhas com mais de um modelo. Existem quatro possíveis configurações de modelos MALBP, apresentadas por Becker e Scholl (2004), e que são mostradas na Tabela 1. No presente trabalho é estudado um caso MALPB-2, ou seja, dado um número fixo de estações de trabalho, deseja-se realizar um balanceamento que minimize o tempo de ciclo da linha.

Número de Estações	Tempo	
	Dado	Minimizar
Dado	MALBP-F	MALBP-2
Minimizar	MALBP-1	MALBP-E

Tabela 1 – Comparativo entre tipos de MALBP
Fonte: Adaptado de Becker e Scholl (2004)

2.1. Equações do modelo MALPB-2 para linhas com buffers

Neste trabalho considera-se a presença de *buffers* de armazenagem temporária em linha como elemento para auxiliar na flexibilidade da linha de modelo misto. Denomina-se como *cluster* os subconjuntos de postos de trabalho que são divididos por *buffers*. Por exemplo, na Figura 2, o *buffer* indicado divide a linha em 2 *clusters*, **k1** e **k2**, o primeiro formado pelos postos de trabalho 1 e 2 e o segundo pelos postos 3 e 4.

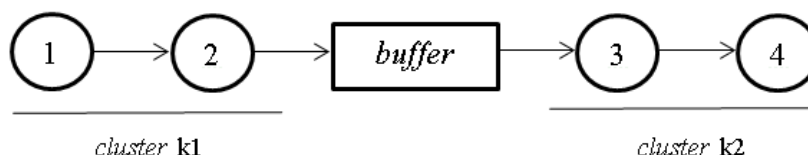


Figura 2 – Definição de cluster
Fonte: Autoria própria

Na sequência, apresenta-se um modelo em PLIM para um caso de MALBP-2, modelo este que foi adaptado do apresentado em Donnini *et al.* (2010). Objetiva-se diminuir o tempo de ciclo com um número de estações determinado. Considera-se, adicionalmente, a presença de

clusters. Um conjunto de seis grupos de restrições, além da função de avaliação, formam o modelo. A variável binária $t_{posto}(t, p)$ indica se a tarefa t está ou não alocada no posto p . A equação 1, então, determina que cada tarefa t , dentro do conjunto de um total de tn tarefas, deve ser alocada em apenas um posto de trabalho p , dentro do conjunto total com pn postos.

$$\sum_{p=1}^{pn} t_{posto}(t, p) = 1, \forall t = 1, \dots, tn \quad (1)$$

O parâmetro $prec(i, j)$ determina se a tarefa i precede a tarefa j , no caso de ser igual a 1. Esse dado é retirado de diagramas de precedências que variam para cada linha estudada. Desta forma, a inequação 2 restringe que cada tarefa j seja realizada no mesmo posto da tarefa i ou em postos que estejam à frente do posto de sua tarefa precedente direta, tarefa i . Ou, seja, a precedência entre i e j deve ser respeitada.

$$\sum_{p=1}^{pn} p \times t_{posto}(i, p) \leq \sum_{p=1}^{pn} p \times t_{posto}(j, p), \quad (2)$$

$$\forall i \in \{1, 2, \dots, tn\}; j \in \{1, 2, \dots, tn\} / prec(i, j) = 1$$

O parâmetro $tempos(t, m)$ indica quanto tempo é utilizado para se realizar cada tarefa t de cada modelo m , dentro do conjunto total de modelo nm . Desta forma, a equação 3 indica que o tempo total que cada modelo ocupa em cada posto é a soma dos tempos das tarefas deste modelo que são realizados neste posto, tempo total sumarizado na variável $dur(p, m)$.

$$\sum_{t=1}^{tn} tempos(t, m) \times t_{posto}(t, p) = dur(p, m), \quad (3)$$

$$\forall p = 1, \dots, pn; m = 1, \dots, nm$$

A variável $tciclo(k, m)$ determina o tempo de ciclo de cada *cluster* k da linha para cada modelo m . O número total de *clusters* é dado pelo número de *buffers* (nb) acrescido de um. A inequação 4 mostra que o tempo de ciclo em cada *cluster* para cada modelo deve ser menor ou igual ao tempo do posto mais ocupado pelo modelo neste *cluster*.

$$dur(p, m) \leq tciclo(k, m), \quad (4)$$

$$\forall k = 1, \dots, nb + 1; p \in Buf_k; m = 1, \dots, nm$$

O parâmetro $prop(m)$ indica a proporção que o modelo m ocupa na linha (*mix* de produção). A inequação 5 determina que o tempo de ciclo em cada *cluster* da linha deve ser a média ponderada do tempo de ciclo de cada modelo neste *cluster*.

$$\sum_{m=1}^{nm} prop(m) \times tciclo(k, m) \leq tc, \forall k = 1, \dots, nb + 1 \quad (5)$$

A variável $diff(k, m)$ é a diferença do tempo de ciclo do modelo em cada *cluster* pelos tempos de duração deste modelo em cada posto deste *cluster*. Em outras palavras, a variável $diff(k, m)$ representa o desequilíbrio dentro dos *clusters* para cada modelo. Isto é representado por meio da inequação 6.

$$tciclo(k, m) - dur(p, m) \leq diff(k, m), \quad (6)$$

$$\forall k = 1, \dots, nb + 1; p \in Buf_k; m = 1, \dots, nm$$

Por fim, a função objetivo, expressão 7 do modelo, é minimizar tc e o somatório de todas as variáveis $diff(k, m)$. Desta forma o modelo procura minimizar o tempo total de ciclo, ao mesmo tempo que procura equilibrar a carga de tarefas dentro de cada *cluster*. É necessário fazer uma ponderação para cada termo da função objetivo para equiparar a importância de um termo em relação ao outro, na proporção adequada. Esses termos de ponderação são determinados experimentalmente. No caso em estudo, considera-se que a minimização de tc é dominante (mais prioritária) em relação à minimização do termo que envolve o grupo de variáveis $diff(k, m)$. Desta forma, um elevado peso foi adotado para o termo que envolve tc . Contudo, ressalta-se que como o modelo é executado até a otimalidade, para um valor minimizado de tc , o somatório mínimo do grupo de variáveis $diff(k, m)$ é também obtido.

$$(minimize)z = (1000 \times tc) + \left(\sum_{m=1}^{nm} \sum_{k=1}^{nb+1} diff(k, m) \right) \quad (7)$$

3. Materiais e métodos

Neste trabalho toda a formulação e resolução do estudo de caso apresentado em Donnini *et al.* (2010), o qual envolve o balanceamento de uma linha de modelo misto de bancos de automóveis contendo um *buffer*, foram refeitos com o auxílio de uma versão acadêmica do *software* IBM ILOG CPLEX Optimization Studio 12.6. O equacionamento matemático foi o apresentado em (1) a (7) na seção anterior. Os tempos computacionais foram inferiores a 15s (execução até a otimalidade) para um computador com Processador Intel® Core™ i3 CPU@3.1GHz, Memória RAM de 4 GB e Sistema Operacional de 64 bits, MS Windows 8.

Depois de resolvido o balanceamento, os dados obtidos foram comparados aos encontrados por Donnini *et al.* (2010). Além disso, com os valores obtidos para a configuração balanceada da linha, foram realizadas simulações computacionais da linha em funcionamento. Nas simulações alguns parâmetros, como o tamanho/posição de *buffers*, foram modificados para que fosse possível observar como a produtividade variava com essas mudanças. A simulação foi programada utilizando o VBA (*Visual Basic for Applications*) do Microsoft Excel 2013.

4. Resultados e discussão

4.1. Balanceamento da linha de bancos de automóveis

A linha de montagem balanceada por Donnini *et al.* (2010) continha dois modelos que deveriam ser produzidos em sete estações de trabalho. O *mix* de produção era de, aproximadamente, 84% para o produto denominado A e 16% para o produto B. Para cada modelo eram realizadas, aproximadamente, 50 tarefas. Além disso, a linha continha um *buffer* intermediário entre as

estações 2 e 3. Após os modelos serem implementados e executados, as tarefas foram alocadas nos postos (balanceamento) de maneira que o tempo de ciclo fosse reduzido e a carga de trabalho fosse equilibrada em cada *cluster*. Os tempos nos postos obtidos após o balanceamento para cada modelo são apresentados na Tabela 2, valores dados em unidades de tempo (u.t.).

Posto	1	2	3	4	5	6	7
Tempo do Modelo A (u.t.)	117	117	129	131	129	130	131
Tempo do Modelo B (u.t.)	216	216	145	154	145	144	147

Tabela 2 – Tempos nos postos (em u.t.) para cada modelo

Nota-se que o produto B demanda mais tempo para sua produção que o produto A. Porém, fazendo uma média ponderada usando os valores do *mix* de produção é observado que o tempo equivalente de cada estação de trabalho é próximo de 130 u.t. Apesar do produto B ocupar mais tempo na linha, uma proporção menor desse produto passa pela mesma (16%).

Os resultados obtidos foram, em essência, os mesmos encontrados por Donnini *et al.* (2010). Os gráficos do balanceamento são apresentados nas Figuras 3 e 4. Observa-se que para o modelo A o tempo de ciclo aumenta após ele passar pelo *buffer* e para o modelo B ocorre o oposto. Assim, os postos 1 e 2 ficam mais “carregados” quando o modelo B passa pela linha e menos carregados na passagem do modelo A. O *buffer* auxilia na compensação das flutuações de tempo oriundas do sequenciamento de diferentes produtos na linha.

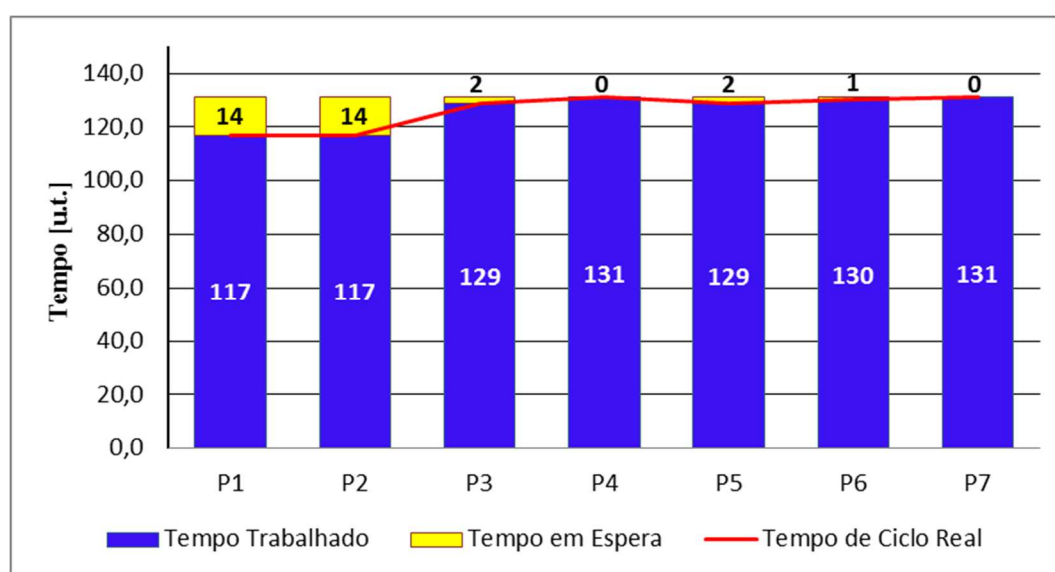


Figura 3 – Diagrama dos tempos dos postos para o produto A – MALBP-2
Fonte: Adaptado de Donnini *et al.* (2010)

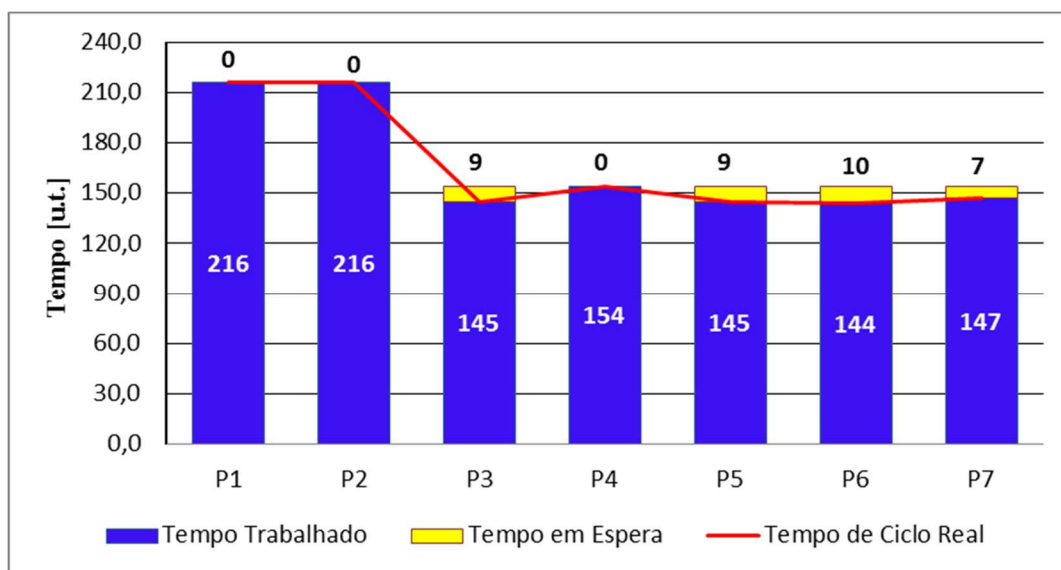


Figura 4 – Diagrama dos tempos dos postos para o produto B – MALBP-2
Fonte: Adaptado de Donnini *et al.* (2010)

4.1. Simulações

Para que se fosse possível analisar a influência de *buffers* na linha foram realizados seis experimentos (E1 a E6), conforme mostrado na Tabela 3. Para cada experimento a configuração inicial de carregamento dos postos da linha era aleatória e a simulação parava após a linha produzir um total de 100 peças. O tempo de ciclo é obtido a partir da divisão do tempo total gasto com a produção pelo número de peças produzidas.

Nos casos que a sequência é dita como randômica os modelos não tinham uma sequência pré-determinada para entrada na linha, porém a probabilidade de entrada correspondia ao *mix* de produção, ou seja, cada peça que entrava tinha 84% de chances de ser do modelo A e 16% de ser do modelo B. A sequência de entrada 1 foi previamente estabelecida como sendo (1,1,1,1,1,2), seguindo aproximadamente o *mix* de produção (6/7 de A e 1/7 de B). Desta forma, em E1 considerou-se a linha sem *buffers* e com sequência de entrada randômica; em E2 foram adotados *buffers* de tamanho, em tese, infinitos, entre cada um dos sete postos; em E3 adotou-se somente um *buffer* de tamanho infinito entre os postos 2 e 3, sequência de entrada randômica; em E4 adotou-se somente um *buffer* de tamanho máximo de duas peças entre os postos 2 e 3, sequência de entrada randômica; em E5 adotou-se somente um *buffer* de tamanho máximo de duas peças entre os postos 2 e 3, sequência de entrada do tipo (1,1,1,1,1,2); e, finalmente, em E6 adotou-se somente um *buffer* de tamanho máximo de uma peça entre os postos 2 e 3, sequência de entrada do tipo randômica. A Tabela 4 mostra os tempos de ciclo encontrados em cada um dos experimentos.

Experimento:	E1	E2	E3	E4	E5	E6
Número de <i>buffers</i>	0	6	1	1	1	1
<i>Buffers</i> entre postos	-	todos	2 e 3	2 e 3	2 e 3	2 e 3
Tamanho de <i>buffer</i>	-	∞	∞	2	2	1
Sequência das peças	randômico	randômico	randômico	randômico	sequência 1	randômico

Tabela 3 – Parâmetros adotados nos experimentos de simulação

Experimento:	E1	E2	E3	E4	E5	E6
Tempo de Ciclo [u.t.]	159	134	141	143	145	148

Tabela 4 – Tempos de ciclo (em u.t.) obtidos nos experimentos de simulação

A Tabela 5 mostra a ocupação dos postos em cada experimento, ou seja, o percentual de tempo que cada posto estava operante durante a simulação. Observa-se que o tempo de ciclo só foi igual ao encontrado no balanceamento no experimento E2, e isso coincidiu com fato de todos os postos estarem ocupados 100% do tempo. Em outras palavras o t_c encontrado no modelo matemático corresponde ao tempo de ciclo máximo teórico, no caso em que todos os postos operam sem pausas. Também nota-se que o maior tempo de ciclo foi quando não havia *buffers* e a ocupação dos postos foi a menor.

Posto	Experimento					
	E1	E2	E3	E4	E5	E6
1	100,0%	100,0%	100,0%	100,0%	100,0%	100,0%
2	94,2%	100,0%	92,1%	92,1%	90,0%	92,5%
3	87,5%	100,0%	98,2%	97,5%	98,1%	93,9%
4	87,0%	100,0%	97,6%	96,9%	96,7%	93,3%
5	85,0%	100,0%	96,0%	95,2%	94,4%	91,7%
6	84,4%	100,0%	95,0%	94,2%	93,0%	90,7%
7	83,4%	100,0%	94,1%	93,2%	91,7%	89,6%

Tabela 5 – Ocupação dos postos (em %) em todos os experimentos

No experimento E3 o *buffer* foi alocado na mesma posição que no modelo matemático e com tamanho infinito. A ocupação do *buffer* no experimento E3 é mostrada na Tabela 6. Por exemplo, durante 26,28% do tempo total da simulação o *buffer* estava ocupado com uma peça. É possível perceber que, neste caso, com o tamanho de *buffer* ilimitado, ele ficou aproximadamente 75% do tempo ocupado por, no máximo, duas peças. Por este motivo os experimentos E4, E5 e E6 foram feitos com tamanho de *buffer* igual a 2, 2 e 1, respectivamente. De modo complementar, em termos práticos, *buffers* de tamanho excessivamente grandes tendem a não ser possíveis de serem implementados devido a restrições físicas de espaço do chão-de-fábrica.

Observando os valores dos tempos de ciclo obtidos em E3, E4 e E6 percebe-se que no caso em que haviam 2 posições no *buffer* o tempo de ciclo encontrado foi próximo do tempo encontrado quando o *buffer* era ilimitado. Em outras palavras, no estudo de caso conduzido, o tamanho de *buffer* necessário pôde ser inferido a partir da ocupação observada do *buffer* configurado para tamanho infinito na simulação da linha. Porém para que esta hipótese seja amplamente validada, experimentos adicionais devem ser conduzidos com outras configurações de linha. Além disso, quando o tamanho de *buffer* foi reduzido para 1 o tempo de ciclo subiu para 148 u.t. (em relação a 143 u.t. de E4). Outro aspecto importante observado é que nos experimentos E4 e E5, em que nos dois casos o *buffer* tinha o mesmo tamanho, porém no primeiro não havia um sequenciamento determinado, os tempos de ciclos foram praticamente os mesmos.

A Figura 4 mostra os gráficos dos tempos de ciclo calculados em função do número de peças produzidas para os experimentos E4 e E5. Percebe-se que, como mostrado anteriormente na Tabela 4, no final da simulação de produção de 100 peças os tempos de ciclo convergiram para um valor próximo de 145 u.t. Contudo, apesar de E4 e E5 terem o mesmo tempo de ciclo no final, no experimento E5, em que havia um sequenciamento padronizado, o valor de t_c

convergiu mais rapidamente para um valor mínimo e apresentou menores variações ao longo do número de peças produzidas. Assim, no experimento realizado, independente do sequenciamento, se as peças entrarem conforme o *mix* programado no modelo, o valor de t_c tende a convergir para um mínimo igual, mas com um sequenciamento adequado observou-se mais uniformidade no tempo de produção e a produtividade converge para o melhor valor possível mais rapidamente.

Ocupação do Buffer (nº de peças)	Tempo [u.t.]	Percentual do Total de Ocupação
0	7304,8	17,09%
1	11234,7	26,28%
2	13536,5	31,66%
3	8753,5	20,47%
4	1885,2	4,41%
5	38,9	0,09%

Tabela 6 – Ocupação do *buffer* no experimento E3

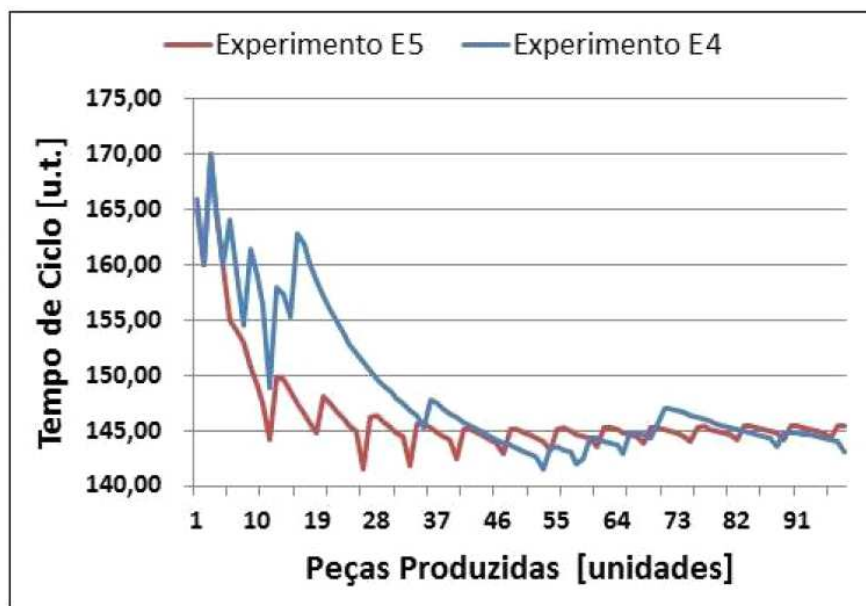


Figura 4 – Tempos de Ciclo x Peças Produzidas nos Experimentos E4 e E5

5. Conclusão

O presente trabalho abordou um estudo de balanceamento e sequenciamento em uma linha de montagem de modelo misto (*Mixed Model Assembly Line*) com *buffers* de armazenagem temporária em linha. O balanceamento foi realizado por meio de um modelo matemático em Programação Linear Inteira Mista (PLIM) implementado no ambiente IBM ILOG CPLEX Optimization Studio. Os estudos de sequenciamento foram desenvolvidos a partir de implementação realizada em VBA (*Visual Basic for Applications*).

O modelo PLIM teve por base a formulação apresentada por Donnini *et al.* (2010). Ao se desenvolver a modelagem matemática e refazer os experimentos de Donnini *et al.* (2010), notou-se que os valores obtidos foram coincidentes, validando o modelo criado. Através das simulações realizadas da linha em operação com diversas configurações de *buffers* notou-se que o valor do tempo de ciclo dado pelo modelo (~130 u.t.) só seria obtido quando todos os postos operassem 100% do tempo ocupados. De acordo com os resultados da simulação, isto só seria possível se um *buffer* intermediário de tamanho ilimitado fosse adotado entre cada um dos postos de montagem. É fato, contudo, que a adoção de um único *buffer* intermediário entre os postos dois e três, com tamanho máximo de duas peças, já contribuiu para a diminuição do tempo de ciclo em relação à condição em que nenhum *buffer* fosse adotado, conforme resultados da Tabela 4.

Nos experimentos realizados observou-se, ainda, que configurações de sequenciamento adequadas podem fazer com que o valor de tempo de ciclo estabilize no patamar mínimo mais rapidamente e fique sujeito a menores variações, conforme Figura 4. A partir da simulação da linha com a capacidade do *buffer* considerada infinita percebeu-se a possibilidade de se sugerir um valor do tamanho de *buffer* prático necessário para manter a produtividade desejada. Entretanto, são necessários mais experimentos com diferentes configurações de linha para validar essa hipótese.

Finalizando, conclui-se que o uso de técnicas conjuntas de balanceamento e simulação pode ser de grande utilidade prática para sugerir uma condição otimizada de uma linha de montagem de modelo misto, conforme indicam os resultados computacionais obtidos.

Referências

- BAYBARS, I.** *A survey of exact algorithms for the simple assembly line balancing problem*, Management Science, vol. 32, n. 8, 1986.
- BECKER, C.; SCHOLL, A.** *A survey on problems and methods in generalized assembly line balancing*, European Journal of Operational Research, vol. 168, p. 694-715, 2004.
- BOYSEN, N.; FLIEDNER, M.; SCHOLL, A.** *Sequencing Mixed-Model Assembly Lines: Survey, Classification and Model Critique*. International Journal of Production Economics, vol. 192, p. 349-373, 2009.
- DANTZIG, G.B.** *Linear programming and extension*, Princeton University Press, Princeton, New Jersey, 1963.
- DONNINI, N.; MAGATÃO, L.; RODRIGUES, L.C.A.** *Balanceamento de uma linha de montagem de bancos de automóveis com buffer intermediário usando programação linear inteira mista*, XLII Simpósio Brasileiro de Pesquisa Operacional, p. 72-83, 2010.
- SCHOLL, A.** *Balancing and Sequencing of Assembly Lines*. 2a. ed. Heidelberg: Physica-Verlag, 1999.