

**UNESP**  
**Faculdade de Engenharia do Campus de Guaratinguetá**

**Guaratinguetá**  
**2010**

ALEXANDRE LEME SANCHES

SEQUENCIAMENTO DE LINHAS DE MONTAGEM MÚLTIPLAS  
EM AMBIENTE DE PRODUÇÃO ENXUTA  
UTILIZANDO SIMULAÇÃO

Tese apresentada à Faculdade de Engenharia  
do Campus de Guaratinguetá, Universidade  
Estadual Paulista, para a obtenção do título  
de Doutor em Engenharia Mecânica na área  
de Transmissão e Conversão de Energia.

Orientador: Prof. Dr. Fernando Augusto Silva Marins.

Co-orientador: Prof. Dr. José Arnaldo Barra Montevechi.

Guaratinguetá  
2010

S211s Sanches, Alexandre Leme  
Sequenciamento de linhas de montagem múltiplas em ambiente de produção enxuta utilizando simulação / Alexandre Leme Sanches - Guaratinguetá : [s.n.], 2010.  
140 f. : il.  
Bibliografia: f. 114-119

Tese (doutorado) – Universidade Estadual Paulista, Faculdade de Engenharia de Guaratinguetá, 2010.  
Orientador: Prof. Dr. Fernando Augusto Silva Marins  
Coorientador: José Arnaldo Barra Montevechi

1. Métodos de linha de montagem I. Título


CDU 658.517

ALEXANDRE LEME SANCHES

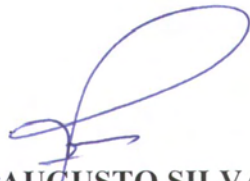
ESTA TESE FOI JULGADA ADEQUADA PARA A OBTENÇÃO DO TÍTULO DE  
“DOUTOR EM ENGENHARIA MECÂNICA”

PROGRAMA: ENGENHARIA MECÂNICA  
ÁREA: TRANSMISSÃO E CONVERSÃO DE ENERGIA

APROVADA EM SUA FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO

  
Prof. Dr. Marcelo dos Santos Pereira  
Coordenador

BANCA EXAMINADORA:

  
Prof. Dr. FERNANDO AUGUSTO SILVA MARINS  
Orientador / Unesp-Feg

  
Prof. Dr. ANTONIO AUGUSTO CHAVES  
Unesp-Feg

  
Prof. Dr. JORGE MUNIZ JUNIOR  
Unesp-Feg

  
Prof. Dr. DAGOBERTO ALVES DE ALMEIDA  
UNIFEI

  
Prof. Dr. RODRIGO SGARPEL  
ITA

Dezembro de 2010

## **DADOS CURRICULARES**

### **ALEXANDRE LEME SANCHES**

<b>NASCIMENTO</b>	<b>01.12.1967 – BRAGANÇA PAULISTA / SP</b>
<b>FILIAÇÃO</b>	<b>Waldemar Sanches Gimenez Maria Cleonice da Silva Leme Sanches</b>
<b>1989/1994</b>	<b>Curso de Graduação Engenharia Mecânica – UNIFEI</b>
<b>2002/2004</b>	<b>Curso de Pós-Graduação - Mestrado Engenharia de Produção - UNIFEI</b>

Dedico este trabalho, de modo especial e com muito amor, ao meu filho Caio, que com seus poucos anos de vida, na sua doce ingenuidade, já é meu grande incentivador.

## AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus pela minha vida, minha família, meus amigos, minha esperança e persistência para atingir meus objetivos.

Aos meus orientadores *Prof. Dr. Fernando Augusto Silva Marins e Prof. Dr. José Arnaldo Barra Montevechi*, pela segurança, sabedoria e amizade.

Ao *Prof. Dr. Dagoberto Alves de Almeida*, pelo incentivo, pelo auxílio na elaboração do tema e pelos comentários e contribuições durante a pesquisa.

Ao *Prof. Dr. Antonio Augusto Chaves* e aos alunos de graduação *Cristiano Ferreira de Almeida e Laís Emy Ishibashi* pela motivação e disposição em colaborar na elaboração e análise dos algoritmos.

Ao *Prof. Dr. Edson de Oliveira Pamplona* pelas oportunidades que me foram apresentadas na carreira acadêmica.

À empresa *Volkswagen Caminhões e Ônibus Ltda.*, em particular na pessoa do *Engenheiro Juliano Martins Moraes*, e à empresa *Maxion Sistemas Automotivos*, em particular na pessoa do *Administrador Marcos Paulo da Silva Pinto*, pelas visitas técnicas viabilizadas, pelos dados utilizados e, principalmente, pelo esclarecimento de dúvidas de ordem prática.

Aos funcionários dos Campi de Itajubá e Guaratinguetá pela dedicação, presteza, simpatia e alegria no ambiente de trabalho.

À *FAPEMIG* pelo suporte à pesquisa.

À *Professora Maria Cleonice da Silva Leme Sanches*, minha mãe, pela revisão do texto.

À *Paola Santiago Prado*, pela compreensão e o apoio de sempre.

“Em algum lugar, algo incrível está esperando para ser descoberto”

Carl Sagan



SANCHES, A. L. **Sequenciamento de Linhas de Montagem Múltiplas em Ambiente de Produção Enxuta Utilizando Simulação**. 2010. 140 f. Tese (Doutorado em Engenharia Mecânica) – Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2010.

## RESUMO

A acirrada competitividade entre as empresas de manufatura, presente no cenário atual, exige a busca por sistemas produtivos cada vez mais eficientes e que estejam inseridos num ambiente de constante aprimoramento. Visando à redução de estoques, e agilidade na produção, o conceito de produção enxuta se apresenta como um conjunto de importantes ferramentas operacionais. Desde que surgiram os conceitos associados à produção enxuta, vários estudos que tratam da utilização eficaz de Linhas de Montagem de Modelos Mistos - LMMM se concentram no sequenciamento de tais linhas. Neste trabalho, linhas de montagem múltiplas para modelos mistos, em ambiente de produção enxuta, são estruturas operacionais nas quais vários fornecedores internos abastecem várias linhas de montagem de modelos mistos simultaneamente, de modo que todas as linhas possam receber peças ou subconjuntos de todos os fornecedores. Para otimizar este sistema, a sequência de programação deve buscar o consumo constante de peças ou subconjuntos, minimizando, assim, o dimensionamento dos *kanbans* e estoques intermediários e, ainda, nivelar a carga de trabalho em cada posto, minimizando as paradas de linha. Baseado no clássico Problema de Monden, que determina o sequenciamento de uma única linha, este trabalho desenvolve um modelo computacional utilizando a Meta-heurística *Simulated Annealing* e a Simulação a Eventos Discretos para o sequenciamento do sistema de abastecimento cruzado com várias linhas. O modelo proposto busca o sequenciamento das linhas de montagem que mais aproxima o consumo real de um consumo ideal constante e atende a um padrão de eficiência predefinido para as linhas.

**PALAVRAS-CHAVE:** Sequenciamento. Linha de Montagem. Simulação. Meta-heurísticas.

SANCHES, A. L. **Sequencing mixed-model multiple assembly lines in lean manufacture environment using simulation.** 2010. 140 f. Thesis (Doctorate in Mechanical Engineering) - Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2010.

### **ABSTRACT**

The fierce competition between manufacturing companies, in this current scenario requires the search for production systems more efficient, and which are embedded in an environment of continuous improvement. Seeking to reduce inventories, and speed of production, the concept of lean production is presented as a range of important operational tools. Since the rise of concepts associated with lean production, several studies dealing with the effective use of assembly lines mixed models - LMMM focus on the sequencing of such lines. In this work, multiple assembly lines for mixed models, in an environment of lean production, are operating structures where several domestic suppliers supply many assembly lines mixed models simultaneously, so that all lines can get parts or subsets of all suppliers. To optimize this system, the programming sequence must seek constant consumption of parts or subassemblies, thus minimizing the scaling of kanbans, and intermediate stocks and also level the workload at each station to minimize line stoppages. Based on the classic problem Monden, which determines the sequencing of a single line, this paper develops a computational model using the Simulated Annealing metaheuristic and Discrete Event Simulation for the sequencing of the supply system with crossed several lines. The proposed model seeks the sequencing of assembly lines that most approximates the actual consumption of a consumption ideal constant and respects a predefined standard of efficiency for the lines.

**KEYWORDS:** Sequencing. Assembly Line. Simulation. Metaheuristics.

## LISTA DE FIGURAS

FIGURA 1 – Push system x pull system.....	32
FIGURA 2 – Linha de montagem de modelos mistos - LMMM.....	40
FIGURA 3 – Consumo de determinado componente “ $a_j$ ” na linha.....	42
FIGURA 4 – Linhas de montagem para GC e para MGC .....	45
FIGURA 5 – Dinâmica da linha de montagem no MGC.....	46
FIGURA 6 – Etapas do desenvolvimento do modelo proposto .....	63
FIGURA 7 – Metodologia da simulação.....	68
FIGURA 8 – Classificação da pesquisa .....	81
FIGURA 9 – Linhas de montagem múltiplas para modelos mistos.....	87
FIGURA 10 – Consumo de determinado componente “ $a_j$ ” em linhas múltiplas.....	88
FIGURA 11 – Consumo de vários componentes em linhas múltiplas.....	88
FIGURA 12 – Divergência entre consumo médio de componentes real e ideal.....	91
FIGURA 13 – Definição da sequência de montagem no software Promodel.....	95
FIGURA 14 – Modelo proposto.....	96
FIGURA 15 – Modelo híbrido envolvendo Simulação e Meta-heurística.....	97
FIGURA 16 – Condicionamento dos dados de entrada pelo software Crystal Ball	102
FIGURA 17 – Modelo de simulação no software Promodel .....	103
FIGURA 18 – Relatório Gráfico - Promodel .....	103
FIGURA 19 – Relatório de Bloqueios - Promodel .....	104
FIGURA 20 – Relatório de Esperas - Promodel .....	104

## LISTA DE TABELAS

TABELA 1 – Matriz de demandas.....	99
TABELA 2 – Matriz de componentes.....	99
TABELA 3 – Tempos de operação por componente/quantidade.....	99
TABELA 4 – Sequências obtidas pelo GCA .....	101
TABELA 5 – Sequências obtidas pelo SA.....	101
TABELA 6 – Correlação “F.O. x Bloqueio Crítico” .....	105
TABELA 7 – Diagrama de consumo – linha1 .....	105
TABELA 8 – Diagrama de consumo – linha2 .....	106
TABELA 9 – Diagrama de consumo – linha3 .....	106
TABELA 10 – Diagrama de consumo geral .....	106

## **LISTA DE QUADROS**

QUADRO 1 – Classificação dos problemas de sequenciamento.....	24
QUADRO 2 – Classificação do método de pesquisa “Modelagem e Simulação”...	80

## LISTA DE ABREVIATURAS E SIGLAS

AD	- Axiomática Descritiva
AG	- Algoritmos Genéticos
AN	- Axiomática Normativa
BDP	- Bounded Dynamic Programming
BT	- Busca Tabu
CIM	- Computer Integrated Manufacturing
ED	- Empírica Descritiva
EDD	- Earliest Due Date
EN	- Empírica Normativa
ERP	- Enterprise Resource Planning
GC	- Goal Chasing
GCA	- Goal Chasing Adaptado
GCE	- Goal Chasing Estendido
GCM	- Goal Chasing Modificado
JIT	- Just in Time
LM	- Lean Manufacturing
LMMM	- Linhas de Montagem de Modelos Mistos
LN	- Lognormal
MS	- Microsoft
PCP	- Planejamento e Controle da Produção
PE	- Produção Enxuta
SA	- Simulated Annealing
SPE	- Sequenciador de Produção Evolutivo
STP	- Sistema Toyota de Produção
TBGC	- Time-Based Goal Chasing
WIP	- Work in Process

## SUMÁRIO

<b>CAPÍTULO 1 - INTRODUÇÃO</b> .....	15
1.1 Considerações Iniciais .....	15
1.2 Linhas de Montagem de Modelos Mistos - LMMM.....	15
1.3 Objetivos.....	19
1.4 Contribuições teóricas e práticas .....	20
1.5 Delimitações da pesquisa .....	21
1.5.1 O “sequenciamento de modelos mistos” ( <i>mixed-model sequencing</i> ) .....	21
1.5.2 O “sequenciamento de automóveis/carros” ( <i>car sequencing</i> ) .....	22
1.5.3 A “programação nivelada” ( <i>level scheduling</i> ) .....	23
1.6 Estrutura do Trabalho .....	26
1.7 Considerações finais.....	26
<b>CAPÍTULO 2 - REVISÃO DA LITERATURA</b> .....	28
2.1 Considerações Iniciais .....	28
2.2 A Filosofia da Produção Enxuta.....	28
2.2.1 Produção “puxada” x “empurrada” (“ <i>pull system</i> ” x “ <i>push system</i> ”).....	31
2.2.2 O sistema <i>kanban</i> .....	33
2.2.3 O dimensionamento de <i>kanbans</i> .....	34
2.3 O problema de sequenciamento de linhas de montagem de modelos múltiplos	36
2.3.1 A evolução dos métodos de sequenciamento de LMMM .....	39
2.4 Programação Dinâmica .....	49
2.5 Meta-heurísticas .....	50
2.5.1 Meta-heurísticas Clássicas.....	52
2.5.2 Recozimento Simulado ( <i>Simulated Annealing</i> ) .....	52
2.5.3 Aplicações de meta-heurísticas em LMMM .....	55
2.6 A Simulação a Eventos Discretos.....	56
2.6.1 Vantagens e desvantagens da simulação .....	56
2.6.2 Simulação estática e dinâmica.....	59
2.6.3 Simulação determinística e estocástica .....	60
2.6.4 Simulação contínua e discreta .....	60
2.6.5 A simulação computacional em ambientes de manufatura .....	61
2.7 Considerações finais .....	61
<b>CAPÍTULO 3 - ASPECTOS METODOLÓGICOS</b> .....	62
3.1 Considerações Iniciais .....	62
3.2 Procedimento metodológico para desenvolvimento do modelo proposto .....	62
3.2.1 Descrição das etapas específicas .....	64
3.3 Softwares utilizados.....	75
3.4 Classificação da pesquisa .....	79
3.5 Considerações finais.....	82
<b>CAPÍTULO 4 - DESENVOLVIMENTO DO MODELO PROPOSTO PARA SEQUENCIAMENTO DE LINHAS MÚLTIPLAS</b> .....	83
4.1 Considerações iniciais .....	83
4.2 Análise das características dos métodos de seq. pesquisados (1ª Etapa) .....	83
4.3 Ajuste do modelo para linhas múltiplas (2ª Etapa) .....	86

4.4 Seleção do método de busca de soluções eficientes (3ª Etapa).....	92
4.5 Elaboração do modelo de simulação computacional (4ª Etapa) .....	93
4.6 Integração lógica das etapas do sequenciamento (5ª Etapa) .....	95
4.7 Considerações Finais .....	97
<b>CAPÍTULO 5 – APLICAÇÃO DO MODELO PROPOSTO PARA SEQUENCIAMENTO DE LINHAS MÚLTIPLAS .....</b>	<b>98</b>
5.1 Considerações Iniciais .....	98
5.2 Exemplo numérico.....	98
5.3 Análise dos resultados .....	106
5.4 Considerações Finais .....	107
<b>CAPÍTULO 6 - CONCLUSÕES E RECOMENDAÇÕES.....</b>	<b>108</b>
6.1 Considerações Iniciais .....	108
6.2 Conclusões sobre o GCA.....	108
6.3 Conclusões sobre o SA.....	109
6.4 Conclusões sobre a simulação a eventos discretos.....	109
6.5 Conclusões sobre o modelo de sequenciamento proposto .....	110
6.6 Conclusões gerais .....	111
6.7 Recomendações para trabalhos futuros .....	112
6.8 Considerações Finais .....	113
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>114</b>
<b>BIBLIOGRAFIA CONSULTADA .....</b>	<b>120</b>
<b>GLOSSÁRIO .....</b>	<b>121</b>
<b>APÊNDICE A – PUBLICAÇÕES GERADAS.....</b>	<b>124</b>
<b>ANEXO A – Relatório Trace – modelo de simulação - Promodel .....</b>	<b>125</b>
<b>ANEXO B – Código fonte em linguagem C – sequenciamento pelo SA.....</b>	<b>134</b>



## CAPÍTULO 1 - INTRODUÇÃO

### 1.1 Considerações Iniciais

A acirrada competitividade entre as empresas de manufatura, presente no cenário atual, exige a busca por sistemas produtivos cada vez mais eficientes, e que estejam inseridos num ambiente de constante aprimoramento.

Visando à redução de estoques e agilidade da produção, o conceito de produção enxuta (PE) ou *lean manufacturing* (LM) se apresenta como um conjunto de importantes ferramentas operacionais. Porém, são necessárias condições específicas e conhecimento detalhado dos processos produtivos, para uma correta adequação às características e condições da produção enxuta.

### 1.2 Linhas de Montagem de Modelos Mistos - LMMM

Desde que surgiram os conceitos associados à produção enxuta, vários estudos, que tratam da utilização eficaz de linhas de montagem se concentram no sequenciamento de tais linhas. Entre os estudos pioneiros pode-se destacar Thomopoulos (1967), que delimitou as fronteiras entre os problemas de balanceamento e sequenciamento de linhas de montagem, seguido por clássicos como Mondem (1983); Miltenburg (1989); Inman e Bulfin (1991), entre vários outros, buscando diferentes enfoques para essa classe de problemas.

Para que as linhas de montagem alcancem um desempenho eficiente, quatro problemas básicos devem ser solucionados (OKAMURA; YAMASHINA, 1979):

- 1) Determinação do tempo de ciclo (*Takt Time*).
- 2) Determinação do número e sequência dos postos de trabalho na linha.
- 3) Balanceamento da linha.
- 4) Sequenciamento, determinação da sequência com que os produtos são programados e inseridos na linha.

O último problema, item quatro, é o foco deste trabalho.

A programação da sequência com que os diversos modelos entram na linha deve ter como objetivo, segundo Miltenburg (1989) e Monden (1983), manter constante a taxa de utilização dos componentes críticos consumidos e, ainda, evitar a sobrecarga dos postos de trabalho, respeitando o tempo de ciclo da linha e evitando paradas indesejadas.

A determinação do tempo de ciclo, ou *takt time*, exige certo cuidado, visto que, quando inferior ao ideal, pode gerar paradas na linha por operações incompletas, e quando superior ao ideal, pode gerar excessiva ociosidade, o que contraria os princípios da produção enxuta.

Muitos são os enfoques para essa classe de problemas, portando faz-se necessária uma primeira apresentação de conceitos básicos e uma classificação dos problemas encontrados na literatura.

O corrente capítulo, introdução, já apresenta certos conceitos elementares, necessários ao entendimento dos objetivos e das delimitações do trabalho, antecipando algumas definições pertencentes ao segundo capítulo, revisão da literatura.

As linhas de montagem, que geralmente constituem a última etapa do processo de manufatura industrial, podem produzir vários modelos distintos, ou seja, variações de um mesmo produto, ou a denominada **família de produtos**, caracterizando as Linhas de Montagem de Modelos Mistos – LMMM.

As LMMM surgiram como resultado de uma imposição do mercado, a cada dia mais sujeito às diversidades e divergências com relação às exigências de clientes, num momento em que a redução de estoques é vista como uma necessidade, quando se trata da sobrevivência das empresas.

Tais linhas de montagem devem possuir flexibilidade suficiente para produzir diferentes modelos, em sequências aleatórias, alternadas, ou especificamente sequenciadas, praticamente sem a necessidade de paradas para preparação ou ajuste (*setup*).

As LMMM, aqui consideradas, trabalham em fluxo contínuo e sem estoque intermediário, sendo que todos os postos de trabalho possuem o mesmo tempo máximo de ciclo e mantêm a mesma sequência de produtos introduzida inicialmente na linha.

A ausência de estoque intermediário indica que cada produto se move diretamente de um posto de trabalho para o posto seguinte e, em certos casos, as operações são executadas com o produto em movimento. Nessas condições, em muitas linhas de montagem são utilizadas esteiras transportadoras ou pontes rolantes como principal meio de transporte interno.

No sequenciamento das LMMM, o horizonte de planejamento pode ser um dia, ou até mesmo um turno, visto que, com base num programa-mestre mensal, que é determinado pelo mercado, é estimada a produção a ser expedida no dia ou turno em questão.

A programação da produção diária é comunicada somente à linha de montagem final, e esta aciona seus fornecedores por um sistema de produção puxada (*pull system*), utilizando o *kanban* como principal recurso.

Um dos principais benefícios do sequenciamento eficiente está no dimensionamento do próprio *kanban*, quando visto como um estoque intermediário, pois devido ao consumo constante da sua produção, não havendo picos de demanda, é possível manter um volume mínimo de material em processo.

A indústria automobilística é considerada, na literatura, como a maior área de aplicação das LMMM, porém tais linhas também são muito encontradas na indústria eletrônica, de eletrodomésticos, de móveis modulares, entre outras.

Um exemplo, na indústria automobilística, poderia ser uma linha de montagem de determinado modelo base, onde podem ser incluídos opcionais como: teto solar, vidros elétricos, câmbio automático, ar condicionado, desembaçadores, bancos de couro, computador de bordo, piloto automático ou sensor de chuva (YANO; RACHAMADUGU, 1991).

Atualmente, muitas estratégias de produção são direcionadas à customização do produto, proporcionando ao cliente uma grande diversidade de opcionais, que resulta em um número expressivo de modelos distintos. Por exemplo, um veículo com 20 opcionais pode permitir mais de um milhão de modelos distintos, ou seja,  $2^{20} = 1.048.576$  modelos.

Nessas condições, se torna inviável a produção baseada em previsão de mercado. A produção deve ter como base a demanda real, exigindo um tempo reduzido entre

pedido e entrega. A minimização desse tempo pode ser decisiva na disputa por frações de mercado, e está diretamente relacionada ao sequenciamento.

Segundo Thomopoulos (1967) e McMullen (2001), o número de diferentes sequências “ $N_{seq}$ ” para determinado período “ $T$ ” (um dia ou um turno) de produção, para os diversos produtos ou modelos “ $A_i$ ” ( $i = 1, 2, 3, \dots, I$ ), é obtido por meio de uma análise combinatória, representada pela equação (1):

$$N_{seq} = \frac{\left(\sum_{i=1}^I Q_i\right)!}{\prod_{i=1}^I (Q_i!)} \quad (1)$$

Sendo:

$Q_i$ : Demanda do Produto  $A_i$  ( $i=1, 2, 3 \dots I$ ) para o período “ $T$ ”.

Por exemplo, existindo seis modelos ( $A_1, \dots, A_6$ ) a serem sequenciados em determinado turno, com as respectivas demandas:  $Q_1 = 42$  unidades,  $Q_2 = 28$  unidades,  $Q_3 = 14$  unidades,  $Q_4 = 2$  unidades,  $Q_5 = 5$  unidades e  $Q_6 = 9$  unidades, deseja-se obter o número de possíveis sequências, para a programação da produção.

O resultado pode ser obtido pela equação (2):

$$N_{seq} = \frac{(42 + 28 + 14 + 2 + 5 + 9)!}{42! \times 28! \times 14! \times 2! \times 5! \times 9!} = 2,87 \cdot 10^{58} \quad (2)$$

Observa-se, neste exemplo, extraído de Thomopoulos (1967), que para um problema com 6 modelos distintos e uma demanda total de 100 unidades, o número de possíveis sequências é  $2,87 \times 10^{58}$ , ou seja, um espaço de soluções de dimensões expressivas.

Mesmo utilizando um recurso computacional, torna-se inviável enumerar e inspecionar todas as possíveis soluções. Portanto, torna-se necessário um estudo específico, pois o sequenciamento pode influenciar, significativamente, a eficiência da linha.

Nesse contexto, a capacidade de flexibilizar e sequenciar produtos nas LMMM aparece como o grande diferencial, o que justifica o volume de pesquisas e publicações destinadas a essa área.

Este trabalho trata do sequenciamento de linhas de montagem múltiplas em ambiente de produção enxuta, uma estrutura operacional composta por linhas de montagem destinadas a modelos diversificados, que são abastecidas por fornecedores comuns.

Buscando proximidade com as situações reais encontradas em ambiente de produção industrial, o trabalho considera as incertezas e aleatoriedades envolvidas nos processos, principalmente no que se refere ao tempo de operação, proporcionando informações relevantes ao pesquisador ou gestor do processo em questão.

Para a busca de soluções satisfatórias para problemas complexos podem ser utilizadas ferramentas específicas como: meta-heurísticas e simulação estocástica.

No caso em questão, a exploração de heurísticas, meta-heurísticas e técnicas de simulação se apresentam com grande potencial de aplicação, conforme se pode observar em Leu et al. (1996), Leu et al. (1997), McMullen (1998), Scholl et al. (1998), McMullen e Frazier (2000), McMullen (2001) e Bautista et al. (1996).

A evolução das técnicas de sequenciamento também gerou trabalhos envolvendo métodos combinados ou adaptados como em Matzka et al. (2009), Shao et al. (2010), Al-Aomar (2010), Jahangirian et al. (2010), Shannon et al. (2010), Fliedner et al. (2010).

### 1.3 Objetivos

O objetivo geral deste estudo é propor um modelo para a obtenção de um sequenciamento eficiente para linhas de montagem **múltiplas** em ambiente de produção enxuta.

Como objetivo específico, pode-se considerar a adequação de ferramentas e métodos para solução de um modelo proposto, sem o comprometimento da obtenção da solução ótima, mas de uma solução viável e eficiente, considerando o ambiente sujeito a incertezas e aleatoriedades.

Considera-se, aqui, que linhas de montagem múltiplas são linhas de montagem de produtos semelhantes, operando simultaneamente e abastecidas pelos mesmos fornecedores internos.

#### 1.4 Contribuições teóricas e práticas

Muitos trabalhos pesquisados focam o sequenciamento de linhas de montagem de modelos mistos (LMMM), podendo-se destacar Monden (1983), Miltenburg (1989), Inman e Bulfin (1991), Kubiak e Sethi (1991), Bautista et al. (1996) e Boysen et al. (2009), porém, o objetivo sempre foi o sequenciamento de linhas únicas.

Estando presente no ambiente industrial, observa-se que, em muitos casos, os fornecedores internos não abastecem somente uma linha ou célula, mas sim múltiplas linhas, o que pode comprometer as metas do sequenciamento, exigindo o superdimensionamento dos *kanbans* devido aos picos de demanda, ou provocando paradas de linha por falta de componentes.

Entre as principais contribuições deste trabalho, pode-se destacar o maior sincronismo entre fornecimento e consumo de componentes, melhor dimensionamento dos *kanbans*, menor probabilidade de paradas de linha, seja por sobrecarga dos postos de trabalho ou por falta de componentes.

A maior aproximação do modelo com a situação real pode ser vista como uma grande contribuição, reforçada pela possibilidade de grande gama de análises, pela simulação a eventos discretos, que envolve a reprodução dos gradativos níveis de aleatoriedade, dependência e consequente complexidade.

Convêm destacar também a utilização da simulação com dupla função: a **função determinística** de validação do modelo matemático proposto e a **função estocástica** de análise dos riscos e incertezas envolvidos nos processos.

Outra contribuição teórica reside na adaptação de uma meta-heurística para o problema de linhas múltiplas utilizando modelagem matricial, expandindo e valorizando a grande capacidade desses métodos de busca com grande evidência na pesquisa operacional atual, além da aplicação conjunta de um modelo estocástico com um modelo determinístico, caracterizando um modelo híbrido totalmente integrado.

## 1.5 Delimitações da pesquisa

Segundo Marconi e Lakatos (2006), delimitar a pesquisa é estabelecer limites para investigação, selecionando um tópico, a fim de impedir que se torne muito extenso ou muito complexo. Portanto, para uma clara determinação dos limites deste trabalho, faz-se necessária uma breve explanação sobre conceitos básicos e classificações dos problemas de sequenciamento em LMMM.

De acordo com Boysen et al. (2009), o problema de sequenciamento em LMMM pode receber três classificações básicas: O **sequenciamento de modelos mistos** (*mixed-model sequencing*), o **sequenciamento de automóveis/carros** (*car sequencing*) e a **programação nivelada** (*level scheduling*).

### 1.5.1 O sequenciamento de modelos mistos (*mixed-model sequencing*)

Com foco no volume e tempo de atividades de cada posto de trabalho, o sequenciamento de modelos mistos considera a dependência entre atividades, a movimentação de operadores e da esteira transportadora, a possibilidade de sobreposição de postos e a flexibilidade de atividades.

Nesse problema, o tempo de ciclo é definido como a média aritmética dos tempos de ciclo de todos os modelos produzidos na linha. Consequentemente, o tempo de processamento de determinados modelos pode ser maior que o tempo médio de ciclo.

Se modelos com alto tempo de processamento forem inseridos seguidamente na linha, isso resultará em sobrecarga de trabalho em determinados postos, e certas operações não estarão concluídas até o limite do posto de trabalho.

Nesse sentido, torna-se necessária também a apresentação das classificações dos postos de trabalho com relação aos seus limites, pois se refere à possibilidade de sobreposição de atividades no decorrer da linha.

Com relação aos limites, na visão de Thomopoulos (1967), os postos de trabalho podem ser abertos ou fechados à direita e à esquerda.

Os diagramas representativos de linhas de produção, geralmente definem o sentido do fluxo do produto da esquerda para a direita, ou seja, o limite esquerdo do posto de trabalho indica o ponto de entrada do produto e o limite direito, o ponto de saída.

Considerando que não existem estoques intermediários, o ponto de saída de um posto de trabalho qualquer é, conseqüentemente, a entrada do posto seguinte, ou tratando-se do último posto, a saída da linha.

Nestas condições, um posto de trabalho aberto à esquerda é um posto com um limite de entrada, que pode ser ultrapassado ainda durante a finalização da operação anterior, ou seja, não é necessária a conclusão da operação anterior para o início da operação do posto em questão, sendo permitida certa simultaneidade na região de transição de postos.

De modo antagônico, o posto de trabalho fechado à esquerda possui um limite rígido que não pode ser ultrapassado pelo produto enquanto a operação anterior não for concluída, implicando, no caso de um atraso na operação anterior, na parada da linha. Para os limites à direita, abertos ou fechados, os conceitos podem ser considerados análogos.

Os postos abertos à esquerda ou à direita, apesar de não restringirem suas atividades por dependência com relação às atividades anteriores ou subseqüentes, possuem a limitação física do alcance do equipamento e instalações necessárias.

### 1.5.2 O sequenciamento de automóveis/carros (*car sequencing*)

Para evitar o intenso esforço na coleta de dados necessária aos problemas de sequenciamento de modelos mistos, o sequenciamento de automóveis tem como objetivo evitar a sobrecarga de trabalho, em cada posto, de uma maneira implícita. Isso é possível formulando um conjunto de restrições e verificando se possíveis seqüências de produtos inseridos na linha respeitam ou minimizam a violação de tais restrições. O principal objetivo desse modelo é a minimização das paradas da linha.

O problema recebe esse nome, sequenciamento de automóveis/carros (*car sequencing*), por representar, em muitos casos, a realidade encontrada no ambiente de



produção de indústrias automobilísticas, onde surgiram os primeiros estudos (DUPLAGA et al., 1996).

### 1.5.3 A programação nivelada (*level scheduling*)

Tendo como principal objetivo o consumo constante de componentes, enquanto as duas primeiras classes de problemas visam à minimização da violação das restrições de capacidade, o problema de programação nivelada presume que o consumo de componentes está diretamente correlacionado com o volume de atividades, relegando a segundo plano a utilização do volume de atividades como critério na determinação do sequenciamento.

Os primeiros métodos de solução do problema de programação nivelada, como o *Toyota Goal Chasing - GC*, proposto por Mondem (1983), foram vistos como limitados, pois determinam a cada entrada, qual o produto que minimiza a função objetivo corrente, não considerando a função objetivo global, apesar de obter apenas uma otimização parcial, ou passo a passo.

O *Goal Chasing* apresenta bons resultados com relação ao nivelamento do consumo de componentes, pois considera exatamente o consumo de cada modelo produzido, além da forte convergência para uma solução eficiente próxima da ótima, ou até mesmo a solução ótima, conforme o caso.

Para solucionar o problema de otimização parcial e obter a otimização global, Miltenburg (1989) propôs um método que otimiza todas as entradas do período, simultaneamente, porém esse método possui limitações críticas, que são descritas no segundo capítulo.

É possível, ainda, considerar o nivelamento, não só dos componentes de primeiro nível, que são os componentes fornecidos diretamente à linha de montagem final, mas também os componentes de níveis superiores, que são os enviados aos fornecedores.

Miltenburg (1989) destaca o interesse no nivelamento em vários níveis, expandindo a idéia aos *kanbans* de matéria prima e até mesmo aos fornecedores externos, visto que todos são influenciados, mesmo que indiretamente e de forma moderada, pelo sequenciamento das linhas de montagem finais. Sempre considerando

que todos os modelos programados possuem, aproximadamente, o mesmo número e *mix* de componentes.

O trabalho de Inman e Bulfin (1991) segue as condições do método de Miltemburg (1989) e apresenta um algoritmo semelhante denominado *Earliest Due Date – EDD*, relativamente rápido no aspecto computacional e que também considera o mesmo número e *mix* de componentes para todos os modelos produzidos.

Por considerar essas aproximações indevidas, este trabalho segue a linha de Mondem (1983) e considera o consumo exato de componentes de cada modelo, o que pode proporcionar maior confiabilidade e precisão ao método proposto no quarto capítulo.

Deve se considerar, também, que este trabalho está focado somente no nivelamento de componentes de primeiro nível, por questões como aplicabilidade e equilíbrio entre tempo computacional e qualidade da resposta. Além do que, o consumo de componentes de níveis superiores pode estar amortecido pelo efeito de estoque intermediário exercido pelos *kanbans*.

O nível de detalhamento do problema de sequenciamento pode ser classificado como alto, médio e baixo e, ainda, considerar a sobrecarga de trabalho e nivelamento do consumo. O Quadro 1 apresenta o nível de detalhamento dos problemas de sequenciamento, conforme cada meta considerada. Os três problemas apresentados possuem certas características comuns, o que, inevitavelmente, resulta em comparações e conclusões também comuns, como é detalhado no decorrer do trabalho.

Quadro 1 - Classificação dos problemas de sequenciamento (adaptado de BOYSEN et al., 2009)

Nível de detalhamento	Foco na sobrecarga de trabalho	Foco no nivelamento do consumo
Alto	Sequenciamento de modelos mistos	—
Médio	Sequenciamento de automóveis	—
Baixo	—	Programação nivelada GC – foco no consumo de componentes
		Programação nivelada EDD – foco na otimização simultânea

Também podem ser encontrados, na literatura, problemas híbridos, como o apresentado em Drexl e Kimms (2001), combinando características dos problemas de programação nivelada com as restrições dos problemas de sequenciamento de automóveis.

Outra delimitação importante está representada pela condição de demandas totais diárias iguais para todas as linhas. Tratando-se de linhas de montagem de modelos semelhantes, ou famílias de produtos, os processos também devem ser semelhantes e seus tempos aproximados. Portanto, as quantidades totais produzidas em cada linha por unidade de tempo, um dia ou um turno, podem ser consideradas, aproximadamente, iguais.

O objeto central de estudo deste trabalho é o problema de **programação nivelada** (*level scheduling*), sendo expandido para linhas múltiplas, porém respeitando as principais características do problema de programação nivelada.

O modelo de sequenciamento deve priorizar duas metas básicas, alinhadas com a filosofia da produção enxuta, para proporcionar eficiência à linha de montagem: o consumo constante de componentes e o nivelamento da carga de trabalho em cada posto.

Desta forma, os fornecedores internos têm reduzidas as suas pausas ou os seus picos de demanda, o que permite um melhor dimensionamento dos *kanbans*, e ainda é possível se obter maior eficiência com relação ao tempo efetivo de operação, evitando sobrecarga ou sobreposição de atividades dos postos de trabalho.

Outras características do problema em estudo são: postos de trabalho abertos à direita e à esquerda com flexibilidade dos produtos em relação à linha, evidência no nivelamento do consumo para primeiro nível de componentes e tempo de operação estocástico.

Para desenvolvimento do modelo proposto, este trabalho utiliza como base o método apresentado por Monden (1983), o *Toyota Goal Chasing - GC*, que se apresenta como um método heurístico viável para solução do problema de sequenciamento de linhas únicas.

## 1.6 Estrutura do Trabalho

O primeiro capítulo, introdução, aborda o ambiente em que o trabalho está inserido, apresentando o problema de pesquisa, os conceitos básicos mais relevantes, os objetivos, as contribuições teóricas e práticas, as classificações, as delimitações do problema e as condições de contorno envolvidas.

No segundo capítulo é apresentada uma revisão da literatura, embasando os conceitos mais específicos utilizados no desenvolvimento do modelo proposto para linhas de montagem múltiplas.

No terceiro capítulo são mostrados os aspectos metodológicos, focando a classificação e complexidade da pesquisa, incluindo as características e justificativas da utilização da simulação a eventos discretos e das meta-heurísticas clássicas.

Devido à utilização de conceitos avançados, o capítulo relacionado aos aspectos metodológicos sucede o capítulo relacionado à revisão da literatura.

O quarto capítulo apresenta o desenvolvimento do modelo proposto, seguido pelo quinto capítulo, onde é realizada uma aplicação do modelo proposto, e é apresentada a análise dos resultados obtidos.

Finalizando, no sexto capítulo são listadas as conclusões, generalizações e propostas para futuras contribuições para trabalhos na área.

## 1.7 Considerações finais

A intenção de um capítulo introdutório é criar um ambiente propício para o desenvolvimento de determinado tema, mostrar as aspirações dos pesquisadores e proporcionar uma visão geral do conteúdo pesquisado. Porém, devido à extensão e complexidade do tema deste trabalho, torna-se necessária a apresentação de certos conceitos, já neste capítulo, pois são considerados fundamentais para a compreensão dos capítulos subsequentes.

Boysen et al. (2009) apresentam, em sua pesquisa, as variações que os problemas de sequenciamento podem abranger. Portanto, apesar das delimitações e explicações

sobre o problema em questão, apresentadas neste capítulo, certos detalhes e particularidades serão explorados de forma gradativa, nos momentos pertinentes.

## CAPÍTULO 2 - REVISÃO DA LITERATURA

### 2.1 Considerações Iniciais

Para uma clara compreensão do modelo de sequenciamento proposto neste trabalho, faz-se necessário um embasamento teórico detalhado. Com essa finalidade, este capítulo aborda a essência da filosofia da produção enxuta, com base nos conceitos associados à produção puxada e o *kanban*. Aborda, também, os principais métodos de sequenciamento encontrados na literatura, sua evolução e suas respectivas aplicações. Seguem-se, então, os conceitos relacionados as meta-heurísticas, os algoritmos mais utilizados para o sequenciamento de linhas de montagem múltiplas. Finalizando, é apresentada a simulação a eventos discretos, sua importância para problemas da área de produção, incluindo os elementos básicos de um software de simulação, o condicionamento dos dados de entrada, a verificação e a validação de modelos simulados.

### 2.2 A Filosofia da Produção Enxuta (*Lean Manufacturing*)

No trabalho **A genealogia da produção enxuta** (*The genealogy of lean production*), de Holweg (2007), é abordada a filosofia e evolução da produção enxuta, sob a ótica da globalização da indústria automobilística, com crescente domínio das empresas orientais. Desse trabalho, podem ser destacados três conceitos elementares aplicados nos problemas de sequenciamento: o *heijunka*, o *jidoka* e o *kanban*.

O nivelamento da produção, também conhecido no Sistema Toyota como *heijunka*, é a programação equalizada por meio do sequenciamento da montagem final em um padrão lógico, que permite a combinação de itens distintos de forma a garantir um fluxo contínuo de produção, nivelando também a demanda de recursos.

O *heijunka* permite a programação da produção em lotes unitários, proporcionando, assim, a minimização dos inventários e o nivelamento do consumo de recursos. Matzka et al. (2009) também reforçam a idéia de que o *heijunka* é o elemento chave do Sistema Toyota de Produção, pois nivela o consumo de componentes

minimizando assim os estoques intermediários (*Work in Process - WIP*). O *heijunka* estabelece, portanto, uma condição para o correto funcionamento do sistema de abastecimento. Sem o *heijunka*, os fornecedores internos podem sofrer picos ou pausas em suas demandas, gerando desabastecimento e ociosidade, situações evitadas no Sistema Toyota.

O *Jidoka* ou *Autonation*, que significa **autonomia do operador**, ou ainda **autonomação** (automação com toque humano), consiste em facultar ao operador o poder de decisão de interromper o processo, sempre que ocorrer alguma situação anormal ou quando a demanda for atingida (MILTEMBURG, 2001).

A autonomação pode ser aplicada em operações manuais, mecanizadas ou automatizadas. Ela permite que a linha seja parada no caso de detecção de falhas, gerando ação imediata para correções. Essa intervenção, segundo Monden (1983), valoriza a atuação do operário e estimula a aplicação de melhorias.

Para Xiaobo e Ohno (2000), devido à autonomação e à possibilidade de paralisação das LMMM pelos operadores, no ambiente de produção enxuta, o sequenciamento da produção aparece como fator relevante, por ser fortemente influente na sobrecarga de trabalho de cada posto.

O *kanban*, considerado por Ohno (1997) o principal elemento da produção enxuta, atua como um amortecedor da produção, ao mesmo tempo em que tem a função de limitar e controlar o estoque intermediário. No dimensionamento do *kanban*, é presumida a demanda constante do componente em questão.

Nos casos com alguma oscilação de demanda, o *kanban* deve ser dimensionado pelo pico de demanda, ou ser projetado com determinada margem de segurança excedente, aumentando o estoque intermediário, o que reforça a necessidade de um sequenciamento eficiente.

A filosofia da produção enxuta, alinhada aos conceitos do *Just in Time*, é reconhecida como uma evolução do Sistema Toyota de Produção – STP, e pode ser entendida como uma prática guiada por um conjunto de técnicas, que visam à redução de todo desperdício que pode ocorrer no processo de fabricação (OHNO, 1997).

Operacionalmente, a redução do desperdício é alcançada produzindo-se somente os itens necessários, na quantidade necessária e no momento exato. Para isso, é

fundamental detectar, prontamente, a demanda e atendê-la, sem a necessidade de estoques.

Para tanto, o *JIT* utiliza três técnicas fundamentais:

- a) Troca rápida de ferramenta.
- b) Layout celular.
- c) Sistema *kanban* (*Pull System*).

A troca rápida de ferramenta permite flexibilidade de produção e, se aliada ao Layout celular, permite cumprir rapidamente o ciclo produtivo, uma vez que as fases de fabricação estão integradas pelas células de manufatura. Desta forma, pode-se atender com maior rapidez às necessidades do mercado (MONDEN, 1983).

Por ser um sistema que puxa a produção, o sistema *kanban* informa aos estágios anteriores do processo de fabricação, a demanda que ocorre no último estágio, geralmente, a expedição. O *kanban* atua como um elo entre os estágios do processo e a demanda final do produto.

Se forem adotados lotes de transferências pequenos, a comunicação entre os diversos estágios fica bastante rápida, permitindo que todas as etapas do processo de fabricação sejam informadas sobre a demanda real, quando necessário.

Monden (1983) argumenta que o propósito das práticas *Just in time*, na Toyota, é obter lucro com a atividade de manufatura, por meio da redução de custo ou melhoria da produtividade. Esta meta pode ser atingida por meio da eliminação de desperdícios relacionados ao excesso de recursos de produção como: mão-de-obra, equipamentos, estoque de matérias-primas e excesso de estoque de produto acabado.

Estes procedimentos, associados a outras práticas como: o controle de qualidade em todas as etapas de fabricação, a flexibilidade de mão-de-obra na execução de tarefas multifuncionais e a organização da área de trabalho, constituem um processo de melhoria contínua do sistema produtivo, conhecido na Toyota por *kaizen*.

A Toyota enfatiza, ainda, o combate às ineficiências geradas pelos altos custos, provenientes do tempo excessivo gasto na preparação de máquinas (*setup*), do tempo de parada programada de equipamentos, em caso de manutenção preventiva, e do tempo de parada imprevisto, relacionado à manutenção corretiva.



A filosofia da produção enxuta se resume, ainda, na implementação de algumas práticas de trabalho e métodos que tornam a manufatura mais flexível. Podem ser citados: a produção em fluxo contínuo, sinalizada de acordo com a demanda de mercado, pequenos lotes de produção que reduzem o *lead time* de fabricação, redução do tempo de *setup*, mão-de-obra multifuncional e **layout celular flexível**, que explora a similaridade de itens fabricados em uma mesma célula.

A recomendação da produção enxuta, quanto à disposição celular do layout orientado ao produto, é comentada por Schomberger (1983), que aponta cinco principais vantagens no uso de células de trabalho em forma de “U”, em oposição à orientação em linha reta: flexibilidade de mão-de-obra, disponibilidade da equipe de trabalho para resolver problemas locais, retrabalho de peças defeituosas na mesma célula, desobstrução da área de produção para trânsito de pessoal e de veículos de transporte de materiais, maior contato entre os operadores e menores percursos para os produtos em processo.

De uma maneira geral, pode-se afirmar que a característica principal da produção enxuta consiste em fornecer a quantidade estritamente necessária de um item, em tempo hábil e no devido local de processamento, atendendo às expectativas de qualidade e quantidade do item de consumo.

De acordo com Monden (1983), o princípio básico do sistema de produção enxuta ainda continua o mesmo desde a sua implantação na Toyota. Porém, sofreu adaptações ao longo de décadas de utilização, tomando novas formas e incorporando recursos provenientes da tecnologia integrada pelo uso do computador (*Computer Integrated Manufacturing - CIM*) com os sistemas de informação (*Enterprise Resources Planning - ERP*), conectando em rede a função manufatura com os departamentos de marketing, financeiro, recursos humanos, compras, engenharia, clientes e fornecedores.

### 2.2.1 Produção “puxada” x “empurrada” (“*pull system*” x “*push system*”)

No que se refere ao mecanismo de acionamento da produção, pode-se dizer que a manufatura na produção enxuta caracteriza-se por puxar a produção (*pull*). Cada

estágio subsequente na linha de montagem solicita, conforme a taxa de produção, os componentes ou subconjuntos dos estágios precedentes.

A demanda é ajustada a uma taxa de produção diária real, conforme a configuração da linha de montagem. Esta demanda de produção diária representa determinada quantidade fixa de cartões *kanban* no sistema que, periodicamente, solicitam as peças das células precedentes.

Por outro lado, por se tratar de um sistema de controle visual, permite a decisão sobre as necessidades de fabricação sem a elaboração e consulta de relatórios, atribuindo, ao pessoal da fábrica, a tarefa de decidir o que produzir a partir das indicações do quadro *kanban*. Passa a ser desnecessária a intervenção do setor de Planejamento e Controle da Produção - PCP nas decisões sobre o que, quanto e quando produzir. Assim, o sistema *kanban* manifesta-se como um eficiente instrumento de informação para comandar com rapidez a reposição dos estoques e, principalmente, em situações de diversificação de produtos, a partir da montagem de componentes padronizados, tornando-se um instrumento que possibilita rápido atendimento ao cliente, sem que seja necessária a manutenção de grandes estoques de produtos e componentes.

A Figura 1 ilustra o sistema produtivo puxado (*pull system*), comparado com o sistema empurrado (*push system*) no que se refere à programação da produção.

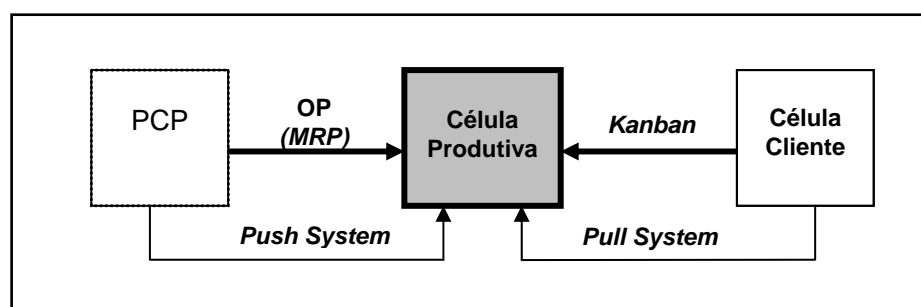


Figura 1 – Push system x pull system (SANCHES et al., 2007)

O estoque pode ser definido como uma consequência da diferença entre o ritmo de abastecimento e o ritmo de demanda. Caso esses ritmos não sejam exatamente iguais, o estoque é inevitável.

Reforçando a idéia de que o estoque é realmente necessário, o custo de gestão do inventário vem a se tornar um fator de competitividade da empresa, portanto, deve ser bem administrado.

A filosofia da produção enxuta evoluiu após anos de ajustes e, atualmente, procura minimizar os estoques entre estágios intermediários do processo, por meio de intervenções que se traduzem, finalmente, em redução do número de cartões *kanbans* no sistema.

### 2.2.2 O sistema *kanban*

Sendo a base da produção enxuta, o sistema *Kanban* foi desenvolvido na década de 60, pelos engenheiros da *Toyota Motors Company*, com o objetivo de tornar simples, rápidas e automáticas as atividades de programação e controle da produção (OHNO, 1997).

O *kanban* é também o mecanismo de controle da produção ou de movimentação de materiais no sistema *Just in Time*.

Esta palavra de origem japonesa significa um sinalizador visual, tal como uma etiqueta ou cartão. No contexto do sistema de manufatura enxuta, o *kanban* representa um cartão contendo um código que identifica o item, a descrição do item, a quantidade movimentada do item ou a quantidade a ser produzida em um posto de trabalho, podendo apresentar um código de barras para interface com o sistema de informações.

Atualmente, o *Kanban* não é apenas uma simples ferramenta de controle de inventário, mas também um recurso que proporciona flexibilidade e agilidade às células de alto *mix* (SCHOMBERGER, 2002).

A palavra *Kanban* significa, literalmente, registro, cartão ou placa visível, que pode ser substituído por outro sistema de sinalização, como luzes, caixas vazias ou até locais vazios demarcados.

Os cartões *kanban* são colocados pelo cliente interno em quadros *Kanban*, para indicar a necessidade de um item em determinada quantidade. Quando não há nenhum cartão no quadro, não se deve produzir nenhuma peça. O *Kanban* permite agilizar a entrega e a produção de peças, além de limitar e controlar o estoque intermediário (*WIP*).

Pode ser empregado em indústrias montadoras, desde que a demanda não oscile excessivamente, fato que pode ser amenizado pelo sequenciamento eficiente. Os *Kanbans* físicos (cartões ou caixas) transitam entre os locais de armazenagem e produção, substituindo formulários, ordens de produção e outras formas de solicitação de peças (SCHOMBERGER, 2002).

O sistema de manufatura enxuta procura trabalhar com lotes menores e maior frequência de abastecimento. A implementação de *kanbans* no ambiente de produção enxuta, visa à redução gradativa no tamanho dos contêineres até um limite ideal, de maneira a obter um lote de tamanho unitário. Isto equivale a um perfeito sincronismo entre taxas de produção e suprimento, com eliminação total de inventário em processo.

A experiência da Toyota mostra que variações de demanda de até 10 % podem ser contornadas, mudando apenas a frequência de transferência de material de um estágio ao outro (OHNO, 1997).

A inclusão ou retirada de cartões no sistema exige o recálculo dos parâmetros do sistema *Kanban*, envolvendo demanda, tempo de resposta e tamanho do contêiner. Por isso, deve ser evitada sempre que possível.

Deve-se verificar a confiabilidade dos dados de previsão de demanda a médio prazo, a fim de ajustar a produção diária à capacidade do sistema, e manter sempre o foco em ações de melhoria no processo de manufatura.

### 2.2.3 O dimensionamento de *kanbans*

As técnicas mais citadas na literatura para calcular o número de *kanbans* e implementar o *JIT* são a programação matemática e a simulação (OHNO, 1997).

O número de contêineres e respectivos cartões *kanban*, que devem controlar o fluxo de produção de um determinado item, podem ser definidos, segundo Ohno (1997) pela equação (3):

$$N = \frac{D}{Q} \times T \times (I + S) \quad (3)$$

Sendo:

D: Demanda;

Q: Quantidade de unidades por contêiner;

T: Tempo de resposta de sistema;

S: Fator de segurança.

O Tempo de Resposta do Sistema “T”, também denominado *Lead Time* de Ciclo de Abastecimento, pode ser decomposto em Tempo de Espera “T<sub>e</sub>”, Tempo de Preparação “T<sub>p</sub>”, Tempo de Operação “T<sub>o</sub>” e Tempo de Movimentação “T<sub>m</sub>”.

O Tempo de Operação geralmente é obtido como tempo unitário “T<sub>ou</sub>”, portanto deve ser multiplicado pela capacidade do contêiner “T<sub>o</sub> = T<sub>ou</sub> · Q”. Então, o número de *kanbans* pode ser definido pela equação (4):

$$N = \frac{D}{Q} \cdot (T_e + T_p + T_t + (T_{ou} \cdot Q)) \cdot (I + S) \quad (4)$$

Sendo:

D: Demanda;

Q: Quantidade de unidades por contêiner;

T<sub>e</sub>: tempo de espera;

T<sub>p</sub>: tempo de preparação;

T<sub>ou</sub>: tempo de operação unitário;

T<sub>t</sub>: tempo de movimentação;

S: fator de segurança.

Diversos autores têm proposto métodos alternativos para determinação do número de *kanbans*, considerando diferentes aspectos e objetivos, porém sempre respeitando os princípios do *JIT*.

Quando se inicia a fabricação de um item, espera-se que o volume a ser produzido preencha todas as caixas, ou seja, que todos os *kanbans* presentes no sistema sejam fabricados, minimizando assim o número de *setups*.

A dificuldade da utilização do sistema *kanban* surge com as incertezas inerentes aos processos produtivos, como as oscilações de demanda, dispersões no tempo de execução de tarefas e até oscilações em quantidades produzidas, devido a problemas de qualidade, mesmo considerando que a filosofia da produção enxuta prevê a minimização de tais incertezas.

Na intenção de solucionar o problema, geralmente, utiliza-se um fator de segurança no dimensionamento determinístico, conforme equação (3), que amplia o número de cartões *kanban* de forma empírica, o que pode afastar o dimensionamento do *kanban* da situação ótima.

### 2.3 O problema de sequenciamento de linhas de montagem de modelos múltiplos

No sistema de produção enxuta, o programa-mestre de produção, que define o programa de montagem final, geralmente opera com horizonte de planejamento de um a seis meses. No mês corrente, o programa-mestre é distribuído em programas diários, com o objetivo de nivelar a carga de trabalho para máquinas e operadores e garantir as entregas com a diversidade que o mercado exige.

Na produção puxada, procura-se limitar o nível de estoques entre as unidades fornecedoras e as consumidoras utilizando para isso o *kanban*. Nesse sistema, os componentes são produzidos para suprir as necessidades da linha de montagem final, sendo esta a responsável pelo acionamento da produção desses componentes.

Considerando que o dimensionamento dos *kanbans* deve utilizar como base a demanda constante, para que esse dimensionamento seja mínimo, preservando a idéia de minimização de estoques intermediários, o sequenciamento das linhas de montagem surge como um dos principais fatores de redução de estoques, considerando que tal

sequenciamento é responsável pela definição do ritmo e constância no qual os componentes são consumidos.

Por outro lado, outros fatores também podem ser influenciados pelo sequenciamento das linhas de montagem, como por exemplo, a carga de trabalho em cada posto de trabalho da linha.

Mesmo considerando tais linhas balanceadas, esse balanceamento pode não ser plenamente eficiente, visto que a LMMM deve estar balanceada para vários modelos e cada modelo exige o seu balanceamento específico. Portanto, de modo geral, o balanceamento terá uma eficiência relativa.

O nivelamento do consumo de componentes pode compor, juntamente com a carga de trabalho designada para os postos, as diretrizes básicas para o sequenciamento.

Na visão de Fisher e Ittner (1999), uma operação excessivamente extensa não é problemática, desde que as operações precedentes ou subsequentes permitam a compensação da carga de trabalho excessiva. Para que essa compensação seja efetiva, deve ser aceita a possibilidade de ligeiras antecipações ou atrasos, que não comprometam a movimentação geral da linha de montagem.

Conforme apresentado por Tsai (1995), em muitas linhas os produtos são montados sobre esteiras rolantes, e os operadores se movem sobre a esteira, juntamente com o produto, enquanto realizam a operação.

Cada operador pode realizar a sua operação somente enquanto o produto estiver próximo ao seu posto de trabalho, com limites definidos na linha. Caso o operador não termine sua operação até que a esteira atinja o limite do seu posto, ele pode parar a linha (autonomação - sistema japonês), ou pode desviar o produto, para que a operação se complete fora da linha (sistema americano), sem que ocorra necessariamente a paralisação do sistema.

Em certas empresas com tecnologia de produção avançada como, por exemplo, a indústria automobilística francesa *Renault S. A.*, além da movimentação padrão da esteira com relação aos postos de trabalho, existe a possibilidade de movimentação do produto em relação à esteira.

Essa flexibilidade gera uma possibilidade de movimento também em relação aos outros produtos na linha, amortecendo as diferenças entre tempos de operação e permitindo que a linha complete sua carga diária de trabalho, antes mesmo do término do turno. Essa característica pode ser muito bem representada nos modelos de simulação a eventos discretos.

No trabalho de Korkmaz e Meral (2001), o problema de sequenciamento é tratado como uma otimização multicriterial, onde o nivelamento do consumo de componentes e da carga de trabalho compõe uma única **função objetivo**, porém, no caso, o enfoque torna-se puramente determinístico.

Analisando pelo aspecto matemático, o problema de sequenciamento pode ser visto como um problema de **Otimização Combinatória** com variáveis discretas, no qual o número de possíveis soluções cresce exponencialmente com o número de produtos a serem sequenciados e com a demanda diária da linha, caracterizando um problema complexo. (MCMULLEN, 2001).

Quando é considerada tal complexidade, o problema de sequenciamento de LMMM pode, também, ser definido como um problema intratável pelos métodos de otimização tradicionais, devido ao elevado tempo computacional exigido para sua solução.

Nesse contexto, surgem os novos métodos e técnicas de otimização combinatória, visando à redução de tempo computacional, sendo que os mais encontrados na literatura para sequenciamento de LMMM são a Programação Dinâmica e as Meta-heurísticas Clássicas, como em Leu et al. (1996), Leu et al. (1997), McMullen (1998), Scholl et al. (1998), McMullen e Frazier (2000), McMullen (2001) e Bautista et al. (1996).

Considerando ainda as particularidades dos problemas de sequenciamento de LMMM, segundo Duplaga et al. (1996), em certos casos, o sequenciamento deve ser alterado no decorrer do período de produção, para evitar paradas de linha ou falta de componentes, devido a incertezas e atrasos que podem ocorrer. Portanto, com o intuito de evitar alterações emergenciais de programação, pode ser interessante analisar a probabilidade dessa linha sofrer atrasos, e buscar sequências que operem dentro de um perfil de eficiência definido pela administração.



Para essa análise, faz-se necessária a aplicação de uma ferramenta específica para situações de risco e incerteza em ambiente de produção, como por exemplo, a simulação a eventos discretos.

### 2.3.1 A evolução dos métodos de sequenciamento de LMMM

Na pesquisa realizada, observa-se que os primeiros estudos relacionados ao sequenciamento de LMMM são creditados a Wester e Kilbridge (1963 apud THOMOPOULOS, 1967).

Até então, problemas semelhantes eram dedicados somente ao **balanceamento** de tais linhas, onde se entende por balanceamento a alocação de tarefas sequenciais aos respectivos postos de trabalho, visando à redução de ociosidade nos postos não **gargalos** e a consequente eficiência da linha.

Com a evolução das técnicas de balanceamento, verificou-se que, para relativa diversidade de produtos montados na mesma linha, o sequenciamento tornou-se um fator de extrema relevância, quando considerado o fornecimento interno de componentes e a ausência de estoques intermediários na linha.

A globalização dos conceitos relacionados à **produção enxuta** e ao sistema *Just in Time* colocou o sequenciamento das linhas de montagem na condição de tarefa primordial para se alcançar as metas estratégicas da empresa. O balanceamento tornou-se uma tarefa básica, por gerar a eficiência local. Já o sequenciamento se tornou um diferencial, por influenciar a programação de toda a fábrica.

O estudo de Thomopoulos (1967) combina balanceamento com sequenciamento e esboça uma primeira tentativa de busca de uma solução eficiente por um método heurístico. Já Lumus (1995) apresenta três regras básicas para sequenciamento de LMMM em ambiente de produção enxuta:

a) Na primeira regra, a programação diária deve estabelecer uma proporção entre os produtos a serem montados e intercalar os produtos, minimizando as quantidades e respeitando as proporções. Por exemplo, se dois produtos “A” e “B” devem ser produzidos nas proporções de 80% e 20%, respectivamente, a sequência pode ser: AAAABAAAAB...

b) Na segunda regra é priorizada a minimização das trocas de produto, ou seja, devem ser montados todos os produtos “A” e, posteriormente, todos os produtos “B”: AAAA AAAABB...

c) A terceira regra é direcionada pela demanda (*demand driven*), ou seja, os produtos são inseridos na linha conforme são solicitados pelos clientes ou pela expedição.

Na realidade, tais regras, com relativa simplicidade, não se mostram eficientes. A primeira regra torna-se inaplicável quando existe grande diversidade de produtos e de demandas.

A segunda regra não está alinhada com os objetivos reais da produção enxuta e a terceira regra torna aleatório o sequenciamento, podendo gerar seqüências extremamente prejudiciais ao sistema.

#### O método de Monden (*Toyota Goal Chasing - GC*)

Monden (1983) propõe um modelo matemático para determinação do sequenciamento eficiente de linhas de montagem de modelos mistos (Figura 2). Essa linha de montagem é abastecida por vários *kanbans* de componentes, porém é tratada como bloco único, ou seja, comprimento zero e, como consequência, considera-se que todos os componentes são consumidos simultaneamente.

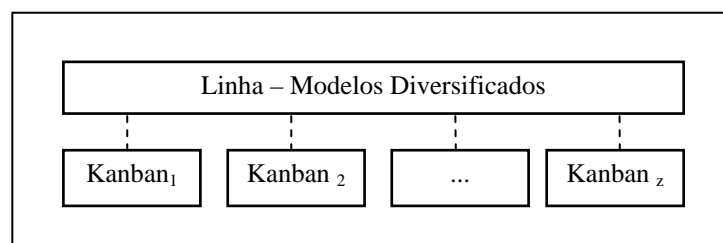


Figura 2 – Linha de montagem de modelos mistos - LMMM

Existem, basicamente, duas metas a serem focadas para a determinação da seqüência:

**Primeira Meta:** Nivelar a carga de trabalho em cada posto “ $l$ ”, respeitando a condição, ou restrição, apresentada na equação (5).

A equação (5) representa a média ponderada dos tempos de montagem em cada posto de trabalho, ou seja, ela considera que, se um processo atrasa, esse atraso pode ser compensado no produto seguinte, desde que **em média**, o tempo de operação em cada posto não ultrapasse o tempo de ciclo da linha:

$$\text{Para cada posto } (l) \left\{ \frac{\sum_{i=1}^I Q_i T_{il}}{\sum_{i=1}^I Q_i} \right\} \leq C \quad (5)$$

$Q_i$ : Demanda Diária do Produto  $A_i$  ( $i=1, \dots, I$ ).

$T_{il}$ : Tempo de Operação do Produto  $A_i$  no posto.

$C$ : Tempo de Ciclo (*Takt Time*) =  $\frac{\text{Tempo total de operação por período}}{\sum_{i=1}^I Q_i}$ .

A restrição apresentada na equação (5) pode ser determinística ou estocástica, dependendo das características consideradas para a variável **tempo de operação**.

Se a variável tempo, considerando individualmente o tempo de operação dos diversos postos de trabalho, apresenta pequenas variações, a primeira meta pode ser tratada como uma condição de parada ou não-parada da linha.

Se tal tempo de operação possui variações significativas, o tempo pode ser tratado como uma variável aleatória, e a equação (5) pode ser adaptada, agora, não mais para determinar uma condição binomial de parada, mas sim para determinar a eficiência da linha com relação ao tempo efetivo de operação.

A primeira meta considera que os diferentes produtos/modelos não possuem o mesmo tempo de operação em cada posto. Alguns podem, até mesmo, consumir um tempo de operação superior ao tempo de ciclo, em certos postos.

A linha de montagem deve ser capaz de ajustar tais diferenças sem necessariamente provocar uma parada. Porém, se produtos com alguma operação

relativamente extensa forem programados repetidamente, os atrasos e paradas de linha serão inevitáveis, devido ao acúmulo localizado de carga.

**Segunda Meta:** Manter constante a taxa de utilização dos componentes críticos consumidos pela linha.

Considera-se aqui, que os componentes críticos são aqueles que necessitam de um controle mais rigoroso na constância do consumo, por serem produtos de alto custo, ou que consomem muito espaço físico, ou com processos complexos, ou de difícil deslocamento, ou até mesmo, por exigência dos fornecedores externos.

O método de Monden consiste em minimizar as distâncias entre o consumo real e o consumo ideal de componentes, representadas na Figura 3.

Considera-se, também, que existe alta correlação entre o número de componentes consumidos e o tempo de operação em cada posto de trabalho, permitindo, assim, modelar o nivelamento do consumo de componentes em detrimento do nivelamento do tempo de operação, tratando o mesmo apenas como uma restrição.

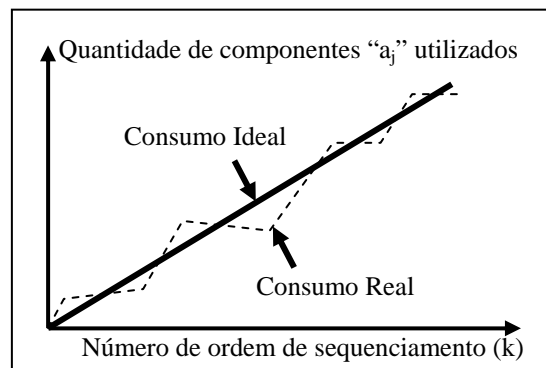


Figura 3 – Consumo de determinado componente “aj” na linha.

O modelo matemático, para busca de um sequenciamento eficiente, descrito por Monden (1983), é representado pela equação (6):

$$\min D_{ki} = \sqrt{\sum_{j=1}^J \left( \frac{K \cdot N_j}{Q} - X_{j,k-1} - b_{ij} \right)^2} \tag{6}$$

Sendo:

$K$ : Contador de tempo normalizado ( $k = 1, \dots, K$ );

$N_j$ : Consumo ideal do componente “j” por unidade de tempo;

$j$ : Índice de componente ( $j = 1, \dots, J$ );

$A_i$ : produtos a serem programados na linha de montagem;

$Q_i$ : Demanda Diária do Produto “i”;

$i$ : Índice de Produto ( $i = 1, \dots, I$ );

$X_{j,k}$ : Consumo parcial do componente “j” até o momento “k”;

$B_{i,j}$ : Consumo do componente “j” no momento “k”;

$D_k$ : Distância entre o consumo real e o consumo ideal no momento “k”.

Algoritmo do Método *Goal Chasing* (MONDEN, 1983)

---

1º. Passo: Faça  $k = 1$ ,  $X_{j,k-1} = 0$ ,  $S_{k-1} = \{1, 2, \dots, I\}$

2º. Passo: Faça  $A_i^*$  o produto que minimiza a distância  $D_k$  no momento  $k$ .

$$D_{ki}^* = \min_i \{D_{ki}\}, i \in S_{k-1}. \text{ (calcule } D_k \text{ para todos os produtos } A_i\text{).}$$

3º. Passo: Se todas as unidades do produto  $A_i^*$  foram incluídas na sequência:

Então: Faça  $S_k = S_{k-1} - \{i^*\}$ .

Senão: Faça  $S_k = S_{k-1}$ .

4º. Passo: Se  $S_k = 0$  (conjunto vazio):

Então: Fim do algoritmo.

Senão: Faça  $X_{j,k} = X_{j,k-1} + b_{i^*j}$  ( $j = 1, \dots, J$ ), volte para o passo 2 e faça  $k = k+1$ .

---

Steiner e Yeomans (1993) defendem a idéia de que se pode permitir uma distância “ $D_k$ ” não mínima, em determinado momento “k”, com o objetivo de se obter uma somatória de distâncias “ $\Sigma(D_k)$ ” mínima, ou seja, uma minimização global do somatório em detrimento de certa minimização parcial individual.

### Críticas ao *Toyota Goal Chasing*

O modelo descrito por Monden e sua heurística podem obter uma solução satisfatória, porém, é considerado um **método míope** (BAUTISTA et al., 1996), por otimizar o momento corrente sem considerar os momentos futuros, mesmo tendo em vista a convergência natural do modelo. Apesar desta limitação, o *Goal Chasing – GC* pode ser visto como um primeiro passo para o desenvolvimento de trabalhos mais complexos.

O problema de sequenciamento de LMMM, com o objetivo de obter uma demanda regular de recursos, é um problema que pode ser modelado de várias maneiras. Miltenburg (1989), Inman e Bulfin (1991) e Kubiak e Sethi (1991) apresentam variações e evoluções ao modelo proposto por Monden.

Conforme apresentado por Celano et al. (2004), o *Toyota Goal Chasing* foi aprimorado, apresentando duas principais variações: o *Goal Chasing Estendido (Extended Goal Chasing – EGC)* e o *Goal Chasing Modificado (Modified Goal Chasing – MGC)*.

#### O *Goal Chasing Estendido - GCE (Extended Goal Chasing – EGC)*

O principal trabalho da linha de pesquisa do EGC pode ser considerado o artigo de Miltenburg (1989), seguido por Inman e Bulfin (1991), o que também é reforçado por Celano et al. (2004). Miltenburg (1989) propôs um método que otimiza, simultaneamente, todas as entradas do período, ou seja, determina toda a sequência a ser inserida na linha, porém sem considerar o consumo individual de componentes entre as entradas.

A grande crítica a esse método refere-se ao fato de que ele admite que todos os modelos programados possuam, aproximadamente, o mesmo número e *mix* de componentes, o que pode ser uma aproximação arriscada e até indevida na grande maioria dos casos. Sem essa aproximação, a otimização simultânea se torna inviável devido à dependência do consumo de componentes do produto anterior, no período

“k”, para a programação do modelo seguinte, período “k+1”. Considerando que esta tese busca generalização, consistência e precisão, o método *EGC* não é utilizado.

O *Goal Chasing* Modificado - *GCM* (*Modified Goal Chasing – MGC*)

Outra importante variação do *Toyota Goal Chasing*, conforme Xiaobo e Zhou (1999) e Celano et al. (2004), é denominada *Goal Chasing* Modificado (*Modified Goal Chasing – MGC*).

No *MGC*, a linha de montagem não é considerada um bloco único como ocorre no *GC*, ou seja, a linha é subdividida em postos de trabalho e cada posto possui suas respectivas dimensões e tempos de operação individuais, o que implica em entradas de componentes em locais e momentos específicos. A Figura 4 mostra as características que diferenciam o *GC* do *MGC*, no caso de uma linha com três postos de trabalho onde são sequenciados três modelos.

Convém destacar que a esteira deve manter velocidade constante. Conseqüentemente, o tempo de operação deve ser dimensionado pela extensão do posto de trabalho, ou seja, para o balanceamento das operações o dimensionamento dos postos deve estar ajustado.

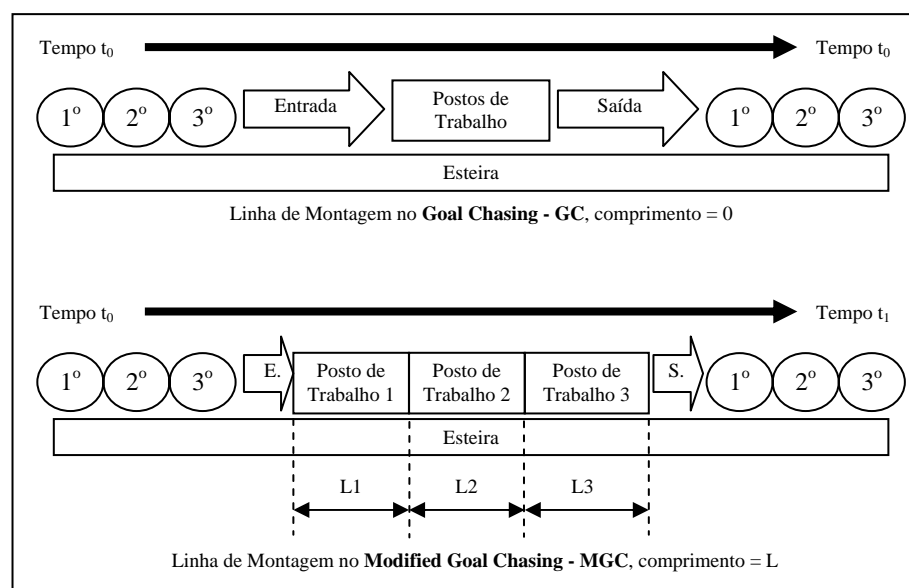


Figura 4 – Linhas de montagem para *GC* e *MGC* (CELANO et al., 2004)

Quando considerada a aplicação prática e o foco na filosofia da produção enxuta, o *MGC* apresenta um detalhamento desnecessário, visto que a meta é o consumo constante de componentes, e não o momento ou local exato onde o componente é inserido na linha.

Porém, quando são analisadas as variações de tempo das operações e é aplicada a simulação a eventos discretos, pode ser necessário o detalhamento, conforme esquema do *MGC*, de modo que se analise a lógica condicional das interdependências e se determine como cada variação afeta o sistema todo.

Xiaobo e Zhou (1999) destacam ainda, como meta a ser considerada, a sequência de entrega solicitada pelo mercado que, dependendo da área de atuação da empresa, pode ser exigente a ponto de divergir da sequência que nivela o consumo de componentes e a carga de trabalho na linha de montagem.

Para a simulação a eventos discretos, que é uma simulação dinâmica, o decorrer do tempo assume grande importância, pois cada evento elementar está associado a determinado momento cronológico, devendo ser considerado na estrutura do problema. A Figura 5 mostra a movimentação da esteira em função do tempo, e como a variável velocidade, combinada com a extensão do posto de trabalho, determina o tempo disponível para a execução da operação.

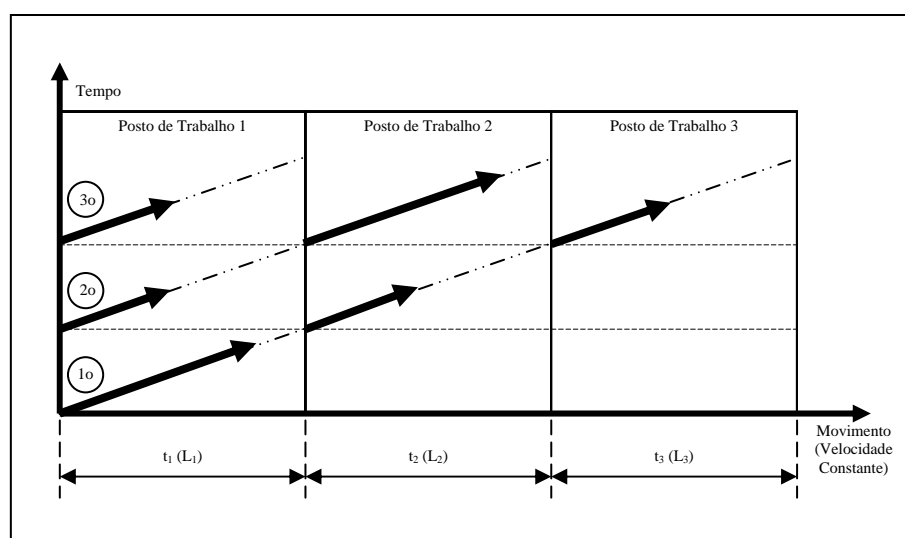


Figura 5 – Dinâmica da linha de montagem no *MGC* (adaptado de XIAOBO; ZHOU, 1999)



Durante as visitas técnicas e as pesquisas, que fundamentam este trabalho, foi observado que, em determinados casos, os produtos podiam realizar pequenos deslocamentos sobre a esteira, ou seja, os produtos podem estar em qualquer ponto entre o seu antecessor e seu sucessor. Esse recurso gera certo amortecimento nas variações dos tempos de operação entre os postos. Portanto, neste trabalho, essa característica é considerada, até porque esse amortecimento pode ser representado com fidelidade no modelo de simulação computacional.

#### *O Goal Chasing Baseado no tempo de operação (Time-based Goal Chasing – TBGC)*

Com o objetivo de considerar as diferenças inevitáveis entre os tempos de operação nas LMMM, Kurashige et al. (1999) desenvolveram uma variação do *Toyota Goal Chasing*, considerando, no sequenciamento, as dispersões entre os tempos de operação e não somente no consumo de componentes.

O TBGC considera momentos irregulares para a entrada de produtos na linha, ou seja, produtos que envolvem operações que consomem mais tempo exigem um intervalo maior até a inserção do produto subsequente, o que não ocorre no GC onde a entrada de produtos na linha segue intervalos regulares.

Outra característica de destaque no trabalho de Kurashige et al. (1999) é a possibilidade de um sequenciamento simétrico, ou seja, a determinação da sequência com que os produtos entram na linha inicia-se, simultaneamente, pelos primeiros e pelos últimos momentos da jornada a ser programada, finalizando no centro da jornada.

#### *O Toyota Goal Chasing na Hyundai Motor Company*

A eficiência, até então, do Toyota Goal Chasing, fez com que o método fosse utilizado até mesmo por grandes concorrentes da Toyota, conforme relata Duplaga et al. (1996). Segundo os autores, a *Hyundai Motor Company* utilizou e aprimorou o *Goal Chasing* com o objetivo de simplificar os cálculos envolvidos.

Os engenheiros da *Hyundai* observaram que, nos casos de componentes com apenas duas opções como: transmissão manual ou automática, não havia necessidade de se inserir os dois componentes no modelo matemático, pois o nivelamento de um já implicaria naturalmente no nivelamento da outra opção. Exemplificando, se todos os modelos, obrigatoriamente, devem receber um modelo de transmissão, o nivelamento do consumo de transmissões automáticas tem como consequência o nivelamento de transmissões manuais.

Assim, muitos componentes, que se enquadram nessa situação, podem ser excluídos do modelo matemático, simplificando a lógica utilizada, reduzindo significativamente o volume de cálculos e o tempo computacional.

#### Os métodos atuais de sequenciamento

Atualmente, vários estudos têm abordado o sequenciamento de LMMM, inserindo características como *setup*, exigências de sequenciamento do mercado, deslocamento de operadores, simultaneidade de operações, entre outras.

Estudos recentes também têm abordado este assunto para exploração de recursos de busca de soluções satisfatórias, utilizando meta-heurísticas ou métodos otimizantes, como por exemplo: Algoritmos Genéticos – AG (*Genetic Algorithms – GA*) em Leu et al. (1996), *Beam Search* em Leu et al. (1997), *Tabu Search* em McMullen (1998) e Scholl et al. (1998), *Simulated Annealing* em McMullen e Frazier (2000), *Ant Colony* em McMullen (2001) e a *Bounded Dynamic Programming (BDP)* em Bautista et al. (1996). Este último pode ser visto como uma evolução da **Programação Dinâmica**, considerada eficiente, apesar da relativa complexidade de cálculos, para o caso de sequenciamento de LMMM.

Sumichrast et al. (2000) propõem um modelo denominado Sequenciador de Produção Evolutivo - SPE (*Evolutionary Production Sequencer - EPS*) com o objetivo de maximizar a produção em LMMM. O SPE visa, basicamente, à minimização do ciclo de produção para atingir 100% da programação, para isso, permite maior flexibilidade no deslocamento dos produtos no decorrer da linha.

Se a sequência da produção permite que todos os postos de trabalho operem com certa ociosidade, o ciclo de produção pode ser reduzido e a produção total da jornada de trabalho concluída em menor tempo. A busca de tal objetivo, segundo Sumichrast et al. (2000), é possível por meio de um algoritmo evolutivo.

## 2.4 Programação Dinâmica

Segundo Cormen et al. (2002), a Programação Dinâmica é uma técnica que busca a solução de um problema, combinando a solução de subproblemas sequenciais, que podem apresentar soluções dependentes, ou seja, cada decisão pode afetar a decisão seguinte.

Um algoritmo de Programação Dinâmica resolve cada subproblema uma única vez e registra sua resposta, evitando assim o trabalho de recalcular a resposta sempre que o subproblema é encontrado. Para evitar recálculos, a Programação Dinâmica pode iniciar o processo de otimização pelas etapas finais do problema e, assim, seguir até as etapas iniciais.

Esse procedimento evita que decisões precedentes afetem as decisões subsequentes, porém as decisões subsequentes devem considerar todas as possibilidades das decisões precedentes. Dessa forma, a cada passo do problema, no decorrer da sequência temporal em que as atividades são executadas, já é conhecida a melhor solução de um ponto qualquer até o final do problema.

A grande vantagem da Programação Dinâmica reside na possibilidade de se reduzir o volume de cálculos, ou seja, reduzir um problema com tempo computacional exponencial para um problema com tempo computacional polinomial, tornando tratável um problema até então intratável.

Held e Karp (1962) aplicaram, de forma pioneira, a Programação Dinâmica em problemas de permutação e sequenciamento como o Problema do Caixeiro Viajante, Problemas de Programação de Produção e Balanceamento de Linhas de Montagem, porém, devido às limitações dos computadores da época, os problemas de grandes dimensões consumiam excessivo tempo de processamento, restringindo a aplicação a problemas relativamente simples.

Atualmente, apesar do desenvolvimento tecnológico na área computacional, o inconveniente encontrado na aplicação da Programação Dinâmica, no sequenciamento de LMMM, reside no volume de restrições de demanda, ou seja, na grande dependência entre as soluções dos subproblemas, exigindo abordagens específicas e funções de recorrência complexas, como observado em Celano et al. (2004).

Quando o problema de sequenciamento de LMMM simples se expande para linhas múltiplas, mesmo utilizando a Programação Dinâmica, pode voltar a se tornar intratável, pois a dimensão **número de linhas de montagem** amplia significativamente o espaço de solução, o que pode enquadrar o problema em um grupo também intratável de problemas, denominado por Cormen et al. (2002) de algoritmos de tempo superpolinomial.

Devido às características descritas, a aplicação da Programação Dinâmica cede espaço à aplicação das meta-heurísticas clássicas, sendo estas mais simples, com possibilidade de flexibilidade da função objetivo e com uma boa qualidade de solução, com relação ao tempo computacional necessário.

## 2.5 Meta-heurísticas

Conforme apresentado por Hillier e Lieberman (2006), um método heurístico é um procedimento que pode encontrar uma boa solução viável para determinada classe de problemas, mas que não é, necessariamente, uma solução ótima.

Em um método heurístico não se pode garantir a qualidade da solução obtida, porém, quando bem elaborado, geralmente, é capaz de fornecer uma solução, no mínimo, próxima da solução ótima. O procedimento também deve ser relativamente eficiente para tratar problemas de grandes dimensões.

Um procedimento heurístico, geralmente, é um algoritmo iterativo completo, em que cada iteração conduz à busca por uma nova solução que, eventualmente, pode ser melhor que a melhor solução encontrada previamente.

Quando o algoritmo termina, por um critério de parada predefinido, a solução apresentada é a melhor solução obtida em qualquer iteração, não necessariamente na última.

O grande problema encontrado nos métodos heurísticos reside na possibilidade do método ficar preso em regiões de ótimos locais, deixando de explorar regiões mais promissoras com relação a boas soluções ou a solução ótima. Para resolver esse problema foram desenvolvidas as meta-heurísticas.

A palavra **meta-heurística** tem sua origem em duas palavras gregas, o prefixo **meta**, que indica uma extensão além de determinado limite e a palavra **heurística**, que significa descobrir, encontrar (GLOVER, 1986).

A otimização combinatória pode ser considerada uma das áreas de maior importância, dentro da pesquisa operacional, para as aplicações das meta-heurísticas, devido às dimensões e à complexidade que essa classe de problemas pode atingir.

A Lógica das meta-heurísticas consiste em aprimorar os procedimentos de determinada heurística, com o objetivo de evitar a estagnação em regiões de ótimos locais.

Segundo Blum e Roli (2003), dois conceitos muito importantes em metaheurísticas são a **intensificação e diversificação**. Estas são as duas forças que determinam o comportamento de uma meta-heurística.

Entende-se por diversificação a busca por novas regiões a serem investigadas, ou seja, uma busca horizontal. Já a intensificação compreende a exploração de determinada região buscando as melhores soluções, ou seja, uma busca vertical.

Dificuldades encontradas nas aplicações das meta-heurísticas (BLUM; ROLI, 2003; GLOVER, 1986; MCMULLEN, 2001):

- Determinar se a melhor solução encontrada é um ótimo local ou global.
- Tratar problemas cujo espaço de soluções apresenta vários ótimos locais.
- Balancear **diversificação e intensificação**.

### 2.5.1 Meta-heurísticas Clássicas

Considerando a aplicação proposta neste trabalho, são consideradas três das meta-heurísticas clássicas mais utilizadas em problemas semelhantes, segundo a literatura: Algoritmos Genéticos - AG, *Simulated Annealing* - SA e Busca Tabu - BT. O que pode ser reforçado pelo trabalho apresentado por McMullen (2001).

A seguir é apresentada uma revisão da literatura abordando o método *Simulated Annealing*, mostrando suas principais características, visando à aplicação no caso em questão: o sequenciamento de linhas de montagem múltiplas.

### 2.5.2 Recozimento Simulado (*Simulated Annealing* - SA)

Inicialmente proposta por Kirkpatrick et al. (1983), a meta-heurística *Simulated Annealing* é considerada uma das primeiras meta-heurísticas encontradas na literatura. O *Simulated Annealing* busca uma analogia com o processo de maleabilização ou recozimento, utilizado para estabilizar fisicamente um metal, ou seja, reduzir a dureza e conseqüente fragilidade do metal tornando-o mais maleável e mais resistente a impacto.

Nesse processo, o metal é aquecido a uma temperatura acima do seu ponto de fusão e, logo em seguida, é realizado um resfriamento controlado, até que o ponto de solidificação seja atingido, gerando uma estrutura estável (KIRKPATRICK et al., 1983).

De acordo com Chaves (2009), em um problema de otimização combinatória, a função objetivo é representada pelo nível de energia, uma solução viável é um estado do sistema, o parâmetro de controle é a temperatura e a melhor solução para o problema é o estado de mínima energia (situação estável e livre de concentrações de energia).

A lógica do SA consiste em permitir, de forma controlada, movimentos que resultem em soluções piores que a solução corrente, visando à exploração de novas regiões do espaço de soluções e, conseqüentemente, evitar a estagnação em ótimos locais.

O SA parte de uma solução inicial qualquer e de uma temperatura inicial predeterminada. A cada temperatura é aplicado o algoritmo de *Metropolis* (METROPOLIS et al., 1953), até que seja atingido o equilíbrio térmico nessa temperatura.

Conforme apresentado por McMullen e Frazier (2000), uma solução vizinha da solução corrente é gerada aleatoriamente a cada iteração. Se essa solução for melhor que a solução corrente, prossegue a busca a partir do vizinho gerado. Caso contrário, se a solução for pior, existe uma probabilidade de aceitação dessa solução, que é função do valor da temperatura corrente.

A representação da função de probabilidade de se aceitar uma solução que seja pior que a solução corrente, é dada por  $e^{-\Delta/T}$ , sendo  $\Delta$ , a variação do valor da função objetivo ao deslocar-se da solução corrente para uma solução vizinha, e T a temperatura corrente.

Como resultado dessa função, em temperaturas mais altas, há uma grande probabilidade de se aceitar uma solução pior, e essa probabilidade é significativamente reduzida em temperaturas baixas.

Para a obtenção da estabilização, a temperatura é reduzida de forma controlada, seguindo uma taxa de resfriamento preestabelecida e um número máximo de iterações.

O critério de parada é definido pela temperatura, o final do processo se dá quando a temperatura se aproxima de zero e nenhuma solução de piora é mais aceita, evidenciando o encontro de um ótimo local.

Assim, o *Simulated Annealing* possui um comportamento aleatório no início do processo de busca e converge para um método de refinamento, à medida que a temperatura se aproxima de zero, pois a probabilidade de aceitar movimentos de piora diminui. Portanto, o critério de resfriamento é o responsável pelo balanceamento entre diversificação e intensificação (CHAVES, 2009).

Pseudocódigo do *Simulated Annealing* (DOWSLAND 1993 apud MARCONE et al., 2002):

---

Seja “ $s_0$ ” uma solução inicial, “ $T_0$ ” a temperatura inicial, “ $\alpha$ ” a taxa de resfriamento e “ $SA_{Max}$ ” o número máximo de iterações para atingir o equilíbrio térmico.

$s \leftarrow s_0$       {solução corrente}

$s' \leftarrow s$       {melhor solução obtida até então}

$T \leftarrow T_0$       {temperatura corrente}

IterT  $\leftarrow 0$     {número de iterações na temperatura T}

enquanto (critério de parada não for satisfeito) faça

enquanto (IterT <  $SA_{max}$ ) faça

        IterT  $\leftarrow$  IterT + 1;

        Gere um vizinho qualquer  $s' \in N(s)$ ;

$\Delta = f(s') - f(s)$ ;

se ( $\Delta \leq 0$ );

então

$s \leftarrow s'$ ;

                se  $f(s') < f(s^*)$  então  $s^* \leftarrow s'$ ;

senão

                Tome  $x \in [0, 1]$

se  $x < e^{-\Delta/T}$  então  $s \leftarrow s'$

fim-se

fim-enquanto

$T \leftarrow \alpha \cdot T$ ;

    IterT  $\leftarrow 0$

fim-enquanto

Retorne  $s^*$ ;

fim SA

---



### 2.5.3 Aplicações de meta-heurísticas em LMMM

Trabalhos como os apresentados por Bautista et al. (1996), Tamura et al. (1999) e Celano et al. (2004) comparam técnicas de sequenciamento de LMMM, envolvendo heurísticas, meta-heurísticas e programação dinâmica, destacando para cada caso específico as vantagens e desvantagens de cada método.

Nesses estudos, evidencia-se a relativa simplicidade das meta-heurísticas, com relação à Programação Dinâmica, e seu potencial de ampla exploração do espaço de soluções, destacando, então, sua superioridade com relação a certos métodos heurísticos.

Shao et al. (2010) aplicaram as meta-heurísticas AG e SA para sequenciamento multicriterial de LMMM e obtiveram resultados expressivos utilizando **função desejabilidade**, apresentada por Derringer e Suich (1980).

O conceito associado à função desejabilidade pode ser incorporado à função objetivo, para ponderar os critérios de sequenciamento utilizados, sintetizando todos os critérios em uma única variável.

Al-Aomar (2010) integra características otimizantes das meta-heurísticas com o realismo inerente à simulação, considerando um sistema onde a meta-heurística e a simulação trocam informações até a obtenção de uma solução satisfatória.

Autores como Glover (1986), Leu et al. (1996, 1997), Scholl et al. (1998), McMullen (1998, 2001), McMullen e Frazier (2000), Chaves (2009), Shao et al. (2010), Al-Aomar (2010) entre outros, continuamente, aplicam, aprimoram e combinam métodos de busca, visando à qualidade da solução e redução do tempo computacional necessário.

De uma forma geral, as meta-heurísticas apresentam grande flexibilidade com relação à função objetivo, independentemente da linearidade da mesma, desde que o problema trate de uma otimização combinatória.

## 2.6 A Simulação a Eventos Discretos

Simulação é uma técnica que utiliza modelos computacionais para representar características de uma instalação real, ou proposta, sob investigação, com o objetivo de avaliar o comportamento desse sistema sob diversas condições.

Para Law e Kelton (2000), simulação computacional é a representação de um sistema real por meio de um computador, para a posterior realização de experimentos para avaliação e melhoria de seu desempenho.

Isto significa a importação da realidade para um ambiente controlado, onde se pode estudar seu comportamento sob diversas condições, sem os riscos físicos e os custos envolvidos em um estudo convencional.

A simulação é utilizada para descrever e analisar o comportamento de sistemas pelo enfoque da causalidade, ou seja, relações de causa e efeito.

A simulação permite a análise de condições de difícil equacionamento, sujeitas a extensas lógicas condicionais, o que a torna adequada a problemas de análise de operações em ambiente de produção.

### 2.6.1 Vantagens e desvantagens da simulação

Os modelos de simulação apresentam inúmeras vantagens em relação aos modelos analíticos. Kelton (1999) afirma que o uso da simulação na realização de experimentos possibilita um conjunto de oportunidades, muitas vezes impossível em situações rotineiras de uma empresa.

Banks et al. (2005) destacam algumas vantagens da simulação:

- Tomada de decisão: a simulação possibilita testar todos os aspectos de uma proposta de mudança com menor consumo de recursos;
- Desacelerar e acelerar o tempo: podem ser analisados fenômenos momentâneos, variando a velocidade de simulação;

- Explorar possibilidades: uma das grandes vantagens da simulação, uma vez que se tenha desenvolvido um modelo de simulação válido, é a possibilidade de explorar novas políticas, processos operacionais ou métodos, sem uma experimentação direta no sistema real, o que envolveria maiores custos;
- Diagnóstico de problemas: as organizações modernas possuem processos complexos, sendo impossível considerar todas as interações que ocorrem em um local em dado momento. Com a simulação, é possível avaliar os efeitos das variáveis e suas interações sobre o sistema;
- Identificação das restrições: a presença de gargalos na produção causa efeitos indesejados. A simulação é capaz de conduzir o analista do problema às restrições pontuais;
- Desenvolvimento do conhecimento: o estudo de simulação promove o entendimento de como o sistema realmente funciona;
- Visualização da planta: conforme o recurso computacional utilizado, é possível visualizar as operações sob diferentes pontos de vista;
- Construção de um consenso: é possível, com a simulação, evitar inferências na aprovação de projetos, pois estas podem ser simuladas e seus efeitos demonstrados;
- Investimentos com sabedoria: o custo típico de um estudo de simulação é, substancialmente, menor do que o total gasto na implementação de estudos de projetos reais;
- Treinamento da equipe: modelos de simulação podem proporcionar excelentes treinamentos quando projetados para tal. Esta estratégia é menos problemática e onerosa que o treinamento no local de trabalho.

Apesar das inúmeras vantagens de sua utilização, a simulação apresenta algumas desvantagens. Segundo Law e Kelton (2000), as desvantagens da simulação são:

- Os modelos são, em geral, relativamente onerosos e consomem tempo para serem desenvolvidos;
- Cada rodada de um modelo de simulação estocástico só estima as características verdadeiras do sistema para um número particular de parâmetros de entrada. Assim, serão necessárias várias rodadas independentes para cada conjunto de parâmetros a serem estudados. Por essa razão é que os modelos de simulação, geralmente, não são tão eficientes em relação à otimização;
- O grande volume de dados gerados por um estudo de simulação faz com que as pessoas confiem no modelo mais do que deveriam. Se um modelo não for a representação adequada de um sistema, não importa qual será a informação obtida, a real utilidade será mínima.

Já de acordo com Banks et al. (2005), as desvantagens da simulação são as seguintes:

- A construção de modelos requer treinamento especial: a simulação é uma arte aprendida com a experiência. É altamente improvável que, dois modelos do mesmo sistema, construídos por pessoas diferentes, sejam iguais;
- A modelagem e a análise podem tomar muito tempo e exigir altos investimentos: restringindo, indevidamente, os recursos para a modelagem e análise, podem ser obtidos resultados insuficientes;
- A simulação pode ser utilizada inapropriadamente: como em casos onde uma solução analítica é possível, ou mesmo preferível.

Strack (1984) recomenda a utilização da simulação quando:

- Não há uma formulação matemática completa para o problema;
- Não há método analítico para a solução do modelo matemático;
- Resultados são mais acessíveis por simulação que por qualquer outro método analítico;
- Não existe habilidade técnica para a resolução do modelo matemático por técnica analítica ou numérica;
- Torna-se necessário observar o desenvolvimento do processo desde o início até o seu término;
- São necessários detalhes específicos do sistema;
- A experimentação na situação real apresenta inúmeros obstáculos ou não é possível.

A maneira como a simulação computacional do sistema trabalha, seja com o uso de distribuições estatísticas, considerando-se o tempo ou não, ou por meio do uso de variáveis discretas ou contínuas, está diretamente relacionada com o tipo de simulação utilizada (HARREL et al., 2000).

As classificações mais usuais são: dinâmica ou estática; determinística ou estocástica; discreta ou contínua. A seguir são apresentadas, com mais detalhes, essas classificações.

### 2.6.2 Simulação estática e dinâmica

Para Law e Kelton (2000), simulação estática é a representação de um sistema em um dado momento. Um exemplo clássico pode ser representado pela simulação de Monte Carlo, bastante utilizada na área financeira e na logística. Já a simulação dinâmica é definida como a representação de um sistema no decorrer do tempo. Esse

tipo de simulação é apropriado para a análise de sistemas de manufatura e serviços, que descrevem o andamento de operações com o passar do tempo.

### 2.6.3 Simulação determinística e estocástica

Na visão de Law e Kelton (2000), os modelos de simulação são determinísticos quando as variáveis de entrada assumem valores exatos. Assim, os resultados ou saídas desse tipo de simulação serão sempre os mesmos, independentemente do número de **replicações**.

Os modelos estocásticos permitem que as variáveis de entrada assumam valores dentro de uma distribuição de probabilidades a ser definida pelo modelador.

Os resultados gerados pelos modelos estocásticos são diferentes a cada replicação, em função da natureza aleatória das variáveis de entrada no modelo. Em consequência, a variável de saída a ser analisada também pode ser ajustada a uma **função densidade de probabilidade**, a partir do conjunto de valores obtidos das diversas corridas.

### 2.6.4 Simulação contínua e discreta

Para Strack (1984), a simulação contínua modela sistemas onde as variáveis mudam continuamente de valor, ou seja, podem assumir infinitos valores dentro de determinado intervalo. É utilizada em estudos que consideram os sistemas constituídos por um fluxo contínuo de informações.

Já a simulação discreta caracteriza-se por variáveis de estado onde as mudanças ocorrem de maneira descontínua, ou seja, sofrem mudanças bruscas. Em outras palavras, a variável em questão só pode assumir um número finito de valores dentro de determinado intervalo.

Na prática, não existe variável contínua em ambiente de simulação computacional. A cada corrida há sempre uma discretização da variável contínua ou,

ao contrário, o condicionamento de dados, aproximando um conjunto de valores discretos a uma função densidade de probabilidade padronizada.

### 2.6.5 A simulação computacional em ambientes de manufatura

Pesquisadores como Harrell et al. (2000) defendem a possibilidade de aplicação da simulação nas mais diferentes áreas, desde a área de serviços, até a área de saúde. No entanto, neste trabalho a linha de estudo é a manufatura.

Segundo Banks et al. (2005), a simulação tem se tornado uma das técnicas mais populares para se analisar problemas complexos em ambientes da manufatura. A simulação é uma ferramenta muito útil, uma vez que prevê o comportamento de sistemas complexos, indicando os movimentos e interações dos componentes desse sistema.

## 2.7 Considerações finais

A integração criteriosa dos conceitos apresentados neste capítulo, leva ao embasamento necessário à elaboração da metodologia de pesquisa, apresentada no terceiro capítulo, e do modelo de sequenciamento de linhas múltiplas, apresentado no quarto capítulo.

Partindo da síntese dos conceitos apresentados, o modelo de sequenciamento desenvolvido busca uma aplicação abrangente e prática, visando ao ambiente real de manufatura.

O modelo proposto apresenta uma série de procedimentos lógicos, que relacionam a filosofia da produção enxuta e os conceitos básicos de sequenciamento de LMMM únicas às técnicas de busca de soluções eficientes e à simulação a eventos discretos.

## CAPÍTULO 3 - ASPECTOS METODOLÓGICOS

### 3.1 Considerações Iniciais

Toda pesquisa científica é motivada por uma dificuldade, uma dúvida, um questionamento, que evolui de forma organizada e se caracteriza como um **problema de pesquisa**.

Os problemas de pesquisa, num ambiente científico, principalmente os que envolvem modelos quantitativos, exigem métodos que busquem soluções precisas e confiáveis e, se possível, com alto poder de generalização, proporcionando assim, contribuição à comunidade científica e ações consideradas de interesse geral para a sociedade.

Este trabalho exige uma atenção especial com relação aos aspectos metodológicos, pois segue duas linhas metodológicas distintas que se integram, visando à solução de um problema com certa complexidade.

Pode-se definir este trabalho como a análise de um problema de otimização determinístico, sujeito a condições ou restrições estocásticas definidas por um perfil de eficiência preestabelecido.

As seguir são apresentados os métodos e recursos utilizados no decorrer deste trabalho.

### 3.2 Procedimento metodológico para desenvolvimento do modelo proposto

O procedimento, a seguir, descreve e caracteriza, em seis etapas específicas, a metodologia para desenvolvimento do **sequenciamento de linhas de montagem múltiplas em ambiente de produção enxuta utilizando simulação**.

São consideradas já concluídas até aqui, as etapas de: definição do problema de pesquisa, delimitação do tema, definição dos objetivos e levantamento do referencial teórico, consideradas partes integrantes da metodologia de pesquisa.

A Figura 6 ilustra as etapas do desenvolvimento do modelo proposto:



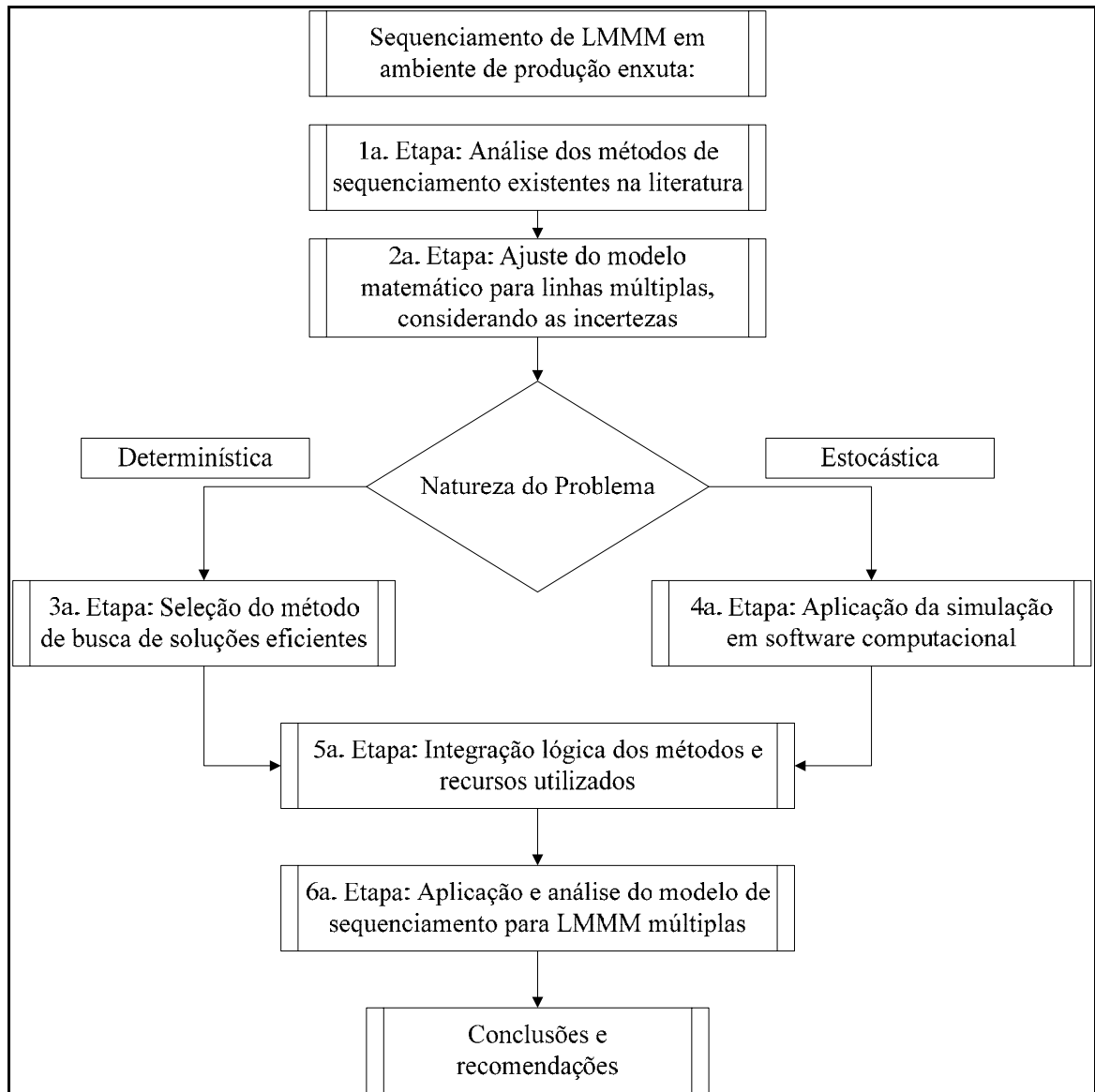


Figura 6 – Etapas do desenvolvimento do modelo proposto.

O recente trabalho de Jahangirian et al. (2010) destaca a tendência atual de desenvolvimento de métodos híbridos, combinando a simulação a eventos discretos com diversas técnicas de busca e otimização.

Na visão dos autores, após sessenta anos de aplicações da simulação computacional, nas diversas áreas do conhecimento, os softwares específicos focaram o realismo e a fidelidade com relação às situações observadas em campo, enquanto o espaço de soluções dos problemas tratados, principalmente na área de manufatura, cresce de forma expressiva, exigindo técnicas auxiliares.

### 3.2.1 Descrição das etapas específicas

#### 1ª Etapa: Análise dos métodos de sequenciamento existentes na literatura

A análise das características dos métodos de sequenciamento, existentes na literatura, pode ser considerada o ponto de partida no desenvolvimento do modelo proposto. Tal análise visa à seleção dos conceitos e recursos relevantes a serem adaptados às linhas de montagem múltiplas.

Os métodos de sequenciamento, encontrados na literatura, apresentam muitas variações com relação às características operacionais e aos objetivos. Para maior eficiência, devem ser inseridas no modelo proposto, apenas as características relevantes, de forma que o trabalho possa apresentar contribuições científicas e ser realmente aplicado em ambiente de produção, sem excessiva complexidade.

Por se tratar de um problema, até então não encontrado na literatura, quando consideradas as condições de multiplicidade de linhas e aspecto estocástico das variáveis, deve-se focar com maior intensidade a **validade interna** do modelo, ou seja, a coerência entre o conceito, ou modelo conceitual, e o modelo matemático desenvolvido, buscando uma solução consistente, no mínimo, para o caso estudado.

A **validade externa** também é desejada, porém não deve ser prioridade num trabalho pioneiro, ou seja, a aplicação do modelo desenvolvido em casos semelhantes, também é um objetivo, porém com menor ênfase.

Para análise de características gerais dos métodos de sequenciamento são pesquisados, com prioridade, artigos tipo *survey* ou *review*, por apresentarem grande abrangência e características de comparação e síntese, como em Boysen et al. (2009), Celano et al. (2004) e Bautista et al. (1996).

#### 2ª Etapa: Ajuste do modelo matemático para linhas múltiplas

O ajuste do modelo matemático de linhas únicas para linhas múltiplas deve proceder com a análise do comportamento de cada variável envolvida,

individualmente e em conjunto, visando à obtenção da **função objetivo** e das respectivas **restrições**.

Esta etapa do desenvolvimento do modelo exige uma análise criteriosa e certa experiência em ambiente de produção. Aqui, é observado, em detalhes, o comportamento de cada variável do modelo de sequenciamento de linhas únicas e comparado ao comportamento esperado para linhas múltiplas.

Para melhor entendimento do comportamento das variáveis envolvidas são realizadas visitas técnicas em empresas da área automobilística e autopeças, onde são encontradas condições semelhantes às do estudo, sendo coletados os dados necessários à análise, ao modelo matemático e à simulação computacional.

Para ratificação das observações em campo, o processo é, então, simulado em software específico, contribuindo não só com o entendimento, mas também com o equacionamento e a verificação do comportamento das variáveis em questão.

### 3ª Etapa: Seleção do método de busca de soluções eficientes

Para a seleção do método de busca de soluções eficientes, devem ser analisadas as características operacionais de cada técnica e verificar a aderência ao problema em estudo. Portanto, é necessária a definição clara dos critérios dessa seleção.

Analisando o caso em questão, podem ser destacados três critérios básicos:

- A **complexidade** do método, considerando o respectivo tempo computacional necessário, em relação à **qualidade da solução** e tendo em vista a afinidade do usuário com o mesmo;
- A necessidade de **penalidades** e extensão da função objetivo;
- As regras de movimentos e **estruturas de vizinhanças**.

Esta análise deve focar, conforme encontrado na literatura, as heurísticas e meta-heurísticas clássicas e, também, os métodos otimizantes compatíveis.

#### 4ª Etapa: Aplicação da simulação em software computacional

Esta etapa deve considerar o condicionamento dos dados de entrada, verificação e validação do modelo computacional, definição do número e condições das corridas e análise das respostas, conforme a metodologia específica para simulação a eventos discretos apresentada a seguir.

#### Metodologia para simulação a eventos discretos

A modelagem computacional de um sistema é uma tarefa que exige certo empenho por parte do modelador, porém, quando conduzida com o planejamento necessário, pode proporcionar muitos benefícios.

A simulação computacional envolve mais que a utilização de um software. Trata-se de um projeto que requer um planejamento prévio de cada uma de suas etapas, além de vasto conhecimento do sistema a ser simulado.

Segundo Strack (1984), os resultados insatisfatórios do uso da ferramenta ocorrem em função da não consideração do aspecto humano da simulação. O sucesso depende da habilidade do usuário em analisar as saídas, identificar alternativas de projeto e novas configurações para melhoria do modelo.

Grande parte dos trabalhos de simulação mal sucedidos tem como principal causa a ausência de um planejamento condizente com a importância de seu estudo. Por isso, simular requer não só o conhecimento de um software específico, mas também pessoas com conhecimento dos passos a serem seguidos, bem como experiência nas áreas estatística e operacional.

A metodologia nos trabalhos de simulação busca sistematizar as atividades de desenvolvimento, facilitando a integração entre software, modelador e usuário, evitando, assim, desperdício de tempo, recursos financeiros e resultados indesejados.

Um exemplo de uma metodologia de simulação foi proposto por Chwif (1999) e Chwif e Medina (2006). Nesta metodologia, os autores afirmam que o desenvolvimento de um modelo de simulação compõe-se, basicamente, de três etapas:

- Concepção ou formulação do modelo;
- Implementação do modelo;
- Análise dos resultados do modelo.

Na fase de **concepção** deve-se entender claramente o sistema a ser simulado e os seus objetivos. Deve-se, também, decidir, com clareza, qual será o escopo do modelo, suas hipóteses e seu nível de detalhamento.

Os dados de entrada também são coletados nesta fase. Ainda nesta fase, o modelo que está na mente do analista (modelo abstrato) deve ser documentado de acordo com alguma técnica de representação de modelos de simulação, a fim de torná-lo um modelo conceitual, de modo que outras pessoas envolvidas no projeto possam também entendê-lo.

Na segunda fase, **implementação**, o modelo conceitual é convertido em um modelo computacional, por meio da utilização de alguma linguagem de simulação ou de um simulador comercial.

Com o avanço tecnológico dos simuladores atuais, a etapa de implementação já não consome tanto tempo como no passado. O modelo computacional implementado deve ser comparado frente ao modelo conceitual, com a finalidade de avaliar se sua operação atende ao que foi estabelecido na etapa de concepção. Alguns resultados devem ser gerados para a verificação e validação do modelo computacional.

Na terceira fase, **análise**, o modelo computacional está pronto para a realização dos experimentos, dando origem ao modelo experimental ou modelo operacional. Nesta etapa são efetuadas várias **corridas** do modelo e os resultados da simulação são analisados e documentados.

A partir dos resultados, conclusões e recomendações sobre o sistema podem ser geradas. Caso necessário, se o resultado da simulação não for satisfatório, o modelo pode ser modificado e este ciclo é reiniciado.

Chwif e Medina (2006) destacam que as fases apresentadas não devem ser interpretadas como uma sequência linear. Em um estudo prático, podem ocorrer

diversas iterações e reinserções no processo, à medida que o entendimento do problema sofre alterações.

A grande limitação para o uso da otimização em simulação é o número de variáveis, sendo seu desempenho reduzido, expressivamente, diante de um modelo com alto número de variáveis a serem manipuladas.

Geralmente, as metodologias de otimização em simulação partem de um modelo já existente e validado.

O primeiro passo é a definição das variáveis de decisão, ou seja, as variáveis que afetam a função objetivo do problema. Em seguida, se define a própria função objetivo, que pode visar à maximização ou minimização, cujo resultado será avaliado pelos algoritmos de otimização na busca de um valor ótimo. O passo seguinte é a definição das restrições do problema, seguido pela configuração de alguns parâmetros, como: número de replicações, precisão e critério de parada.

Mitroff et al. (1974) propõem um modelo cíclico, onde a simulação passa pelas etapas de modelagem conceitual, modelagem computacional, experimentação, conclusão e implementação. Um modelo adaptado de Mitroff et al. (1974), aplicado neste trabalho, é apresentado na Figura 7.

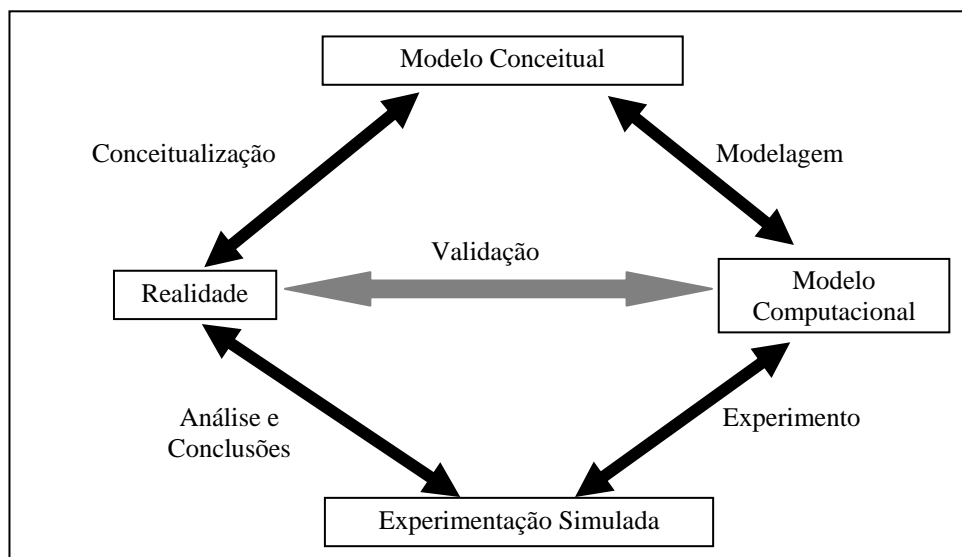


Figura 7 – Metodologia da simulação (adaptado de MITROFF et al., 1974).

A simulação e a otimização são elementos da Pesquisa Operacional, muito difundidos, porém, somente há pouco mais de uma década, estes métodos vêm sendo utilizados conjuntamente (LAW e KELTON, 2000).

Este uso conjunto se mostra bastante eficiente e relativamente simples, sendo aplicável nas mais diversas áreas.

Em um estudo de otimização em simulação, tão importante quanto conhecer o funcionamento dos softwares é a sequência e o modo de execução de cada etapa do projeto. Assim, é preciso que se siga uma metodologia de implementação bastante consistente e devidamente planejada.

Em geral, a técnica da simulação não produz resultados exatos, mas produz boas aproximações, quando há dificuldade ou mesmo impossibilidade de se conseguir a solução experimental do problema.

Um modelo de simulação tem caráter descritivo, com maior ênfase ao desenvolvimento de um sistema que se adapte à realidade, do que a um processo de cálculo que leve a um ponto ótimo.

Convém destacar que, nenhum objeto existente na realidade pode ser explicado corretamente, examinando-se suas partes independentemente. É importante que seja realizado um estudo global de todas as partes do sistema.

### Verificação e validação de modelos (V&V)

Na interpretação de Law e Kelton (2000), a **validação** é um teste de representatividade entre o sistema simulado e o sistema real, ou seja, as variáveis reais em questão devem estar fielmente representadas no modelo simulado, apresentando comportamento e características semelhantes às reais, nas condições específicas do estudo.

Já o termo **verificação**, de acordo com Balci (1997), está relacionado com a linguagem de programação e significa a checagem da existência de erros de programação (*bugs*) no modelo.

A validação relaciona o modelo real com o simulado, enquanto a verificação relaciona o modelo computacional com suas regras próprias de programação.

Para Balci (1997), os processos de validação e verificação devem ser quase que contínuos durante a modelagem e simulação. Verificar e validar somente no final do projeto pode tornar sua correção inviável, comprometendo todo o trabalho.

De acordo com Chwif e Medina (2006), duas importantes regras da verificação e validação devem sempre ser levadas em consideração:

a) não há como garantir que um modelo seja 100% válido. O que se busca é um aumento da confiança no modelo, ou mesmo uma representação satisfatória do sistema;

b) não há como garantir que um modelo seja totalmente livre de *bugs*. Embora o modelo possa ser verificado para determinadas circunstâncias, não há como garantir que, para quaisquer circunstâncias, funcione conforme a situação real.

Balci (1997) também apresenta algumas considerações sobre o processo de verificação e validação de modelos:

a) a verificação e a validação devem ser conduzidas ao longo de todo o ciclo da modelagem e simulação;

b) os resultados da verificação e validação não devem ser considerados como uma variável binária, onde o modelo é absolutamente correto ou incorreto;

c) um modelo de simulação é construído, respeitando os objetivos do projeto de modelagem e simulação, e sua credibilidade é julgada de acordo com a adequação a esses objetivos;

d) a verificação e validação requerem independência para prevenir algum tipo de favorecimento por parte de quem está desenvolvendo o modelo;

e) a verificação e validação constituem um processo difícil, que exige criatividade e perspicácia;

f) o teste completo de um modelo de simulação é impossível;



g) a verificação e validação devem ser planejadas e documentadas;

h) erros devem ser detectados com a maior antecedência possível, no ciclo da modelagem e simulação;

i) um problema bem formulado é essencial para a aceitabilidade dos resultados do ciclo de modelagem e simulação.

### Técnicas para verificação de modelos

A literatura apresenta algumas técnicas de verificação de modelos como segue:

a) **verificação e implementação modular** (CHWIF; MEDINA, 2006; BANKS, 1998): a confecção do modelo em partes permite uma verificação focada em cada fração do modelo construída, o que facilita a identificação de erros e uma ação corretiva mais clara para o modelador. Esta técnica de verificação complementa a idéia de que o modelo não deve ser construído diretamente na sua forma final. Deve ser aprimorado, na medida em que o modelador realiza, progressivamente, verificação das versões anteriores;

b) **uso de valores constantes**, ou médios, comparados a cálculos nominais (CHWIF; MEDINA, 2006; KLEIJNEN, 1995): tomar os valores médios de cada distribuição e considerar estes valores como determinísticos pode facilitar a identificação de erros de programação, pois podem ser comparados a resultados conhecidos, esperados ou equacionados;

c) **utilização do depurador** (CHWIF; MEDINA, 2006; BANKS, 1998): a presença de um depurador no software de simulação viabiliza e simplifica o processo de verificação;

d) **uso da animação gráfica** (CHWIF; MEDINA, 2006; KLEIJNEN, 1995; BANKS, 1998): considerada uma poderosa ferramenta de verificação, a animação permite ao modelador acompanhar o processo na tela do computador. Eventos como o

abastecimento de postos de trabalho, formação de filas, trajeto de itens, movimentação de funcionários, paradas para manutenção, podem ser visualizados por meio da animação gráfica, permitindo ao modelador verificar a presença de possíveis falhas na programação. A animação permite não apenas a visualização de erros de programação, mas também erros conceituais, o que também implica no processo de validação.

Segundo Kleijnen (1995), o analista pode avaliar alguns resultados intermediários da simulação e comparar esses resultados com os dados de saída da simulação. O processo de captura de todos os resultados intermediários do modelo computacional, de forma automática, é chamado de *tracing*.

O autor também aponta como técnica de verificação, a comparação dos dados de saída da simulação com resultados analíticos, quando possível. Esta ação é facilitada pelo uso de valores constantes para as variáveis de entrada.

Banks (1998) também indica como técnica de verificação, a documentação da construção e execução do modelo, por meio da presença de comentários nas linhas de programação.

### Técnicas para validação de modelos

Para apresentação das principais técnicas de validação encontradas na literatura, faz-se necessária uma primeira classificação com relação à participação humana no processo (SARGENT, 2005). Nesse aspecto, as técnicas podem ser classificadas como subjetivas ou objetivas.

Entre as técnicas subjetivas, destacam-se o teste de Turing, a duplicação de modelos, a comparação com modelos prévios e a validação face-a-face, descritas a seguir:

#### Técnicas de validação subjetivas

a) **Teste de Turing** (ou validação **Black-box**) (KLEIJNEN, 1995; SARGENT, 2005): seleciona-se um especialista no sistema que foi simulado e apresenta-se a ele

dois conjuntos de dados de saída, um vindo do sistema real e outro do modelo de simulação. Caso o especialista não consiga distinguir se os resultados são oriundos do sistema real ou do modelo de simulação, então o modelo poderia, em certa extensão, ser considerado válido;

b) **duplicação de modelos** (CHWIF e MEDINA, 2006): prática eficiente, porém onerosa, pois supõe a existência de duas equipes independentes desenvolvendo modelos do mesmo sistema. Desta forma, se as duas equipes (operando de forma independente), desenvolverem modelos similares, este seria um bom indicador da validade do modelo;

c) **comparação com modelos prévios** (CHWIF e MEDINA, 2006): desenvolvimento de um modelo a partir de outro já validado;

d) **validação face-a-face** (KLEIJNEN, 1995; SARGENT, 2005; CHWIF e MEDINA, 2006): discussão com especialistas, objetivando alcançar com eles o aval para o uso do modelo. Nesta técnica de validação, Kleijnen (1995) defende a utilização da animação, como forma de apresentação do modelo aos especialistas.

Halachmi et al. (2001) defendem a análise visual do modelo, denominada *eyeballing*, como ferramenta para validação, pois a sequência das operações observadas, em diferentes velocidades, pode permitir a percepção de semelhanças ou divergências entre o modelo real e o computacional. Nesse conceito também podem ser inseridos os relatórios gráficos, por também utilizarem recursos relacionados à percepção visual.

#### Técnicas de validação objetivas

Entre as técnicas objetivas, destacam-se os testes de hipóteses estatísticos, que comparam dados **reais** ou **resultados analíticos**, com dados resultantes do modelo simulado. No caso de projetos em estudo, onde ainda não existem dados reais, podem ser utilizados dados esperados ou idealizados para a comparação com os dados simulados (KLEIJNEN, 1999).

Os testes estatísticos mais indicados para validação de modelos de simulação, segundo Kleijnen (1995), Kleijnen (1999), Banks et al. (2005), Sargent (2005) e Chwif e Medina (2006), são os seguintes:

- a) Intervalo de confiança para diferenças;
- b) Teste “t”;
- c) Teste “t” pareado;
- d) O teste de Wilcoxon.

#### 5ª Etapa: Integração lógica dos métodos e recursos do modelo de sequenciamento

A elaboração do procedimento para integração lógica dos métodos e recursos utilizados deve definir o sequenciamento da produção que melhor se enquadra a um ranking de sequências e ao perfil de eficiência definido pelo usuário. Para melhor compreensão, tal procedimento deve ser apresentado em um fluxograma.

Nesta etapa, uma decisão estratégica deve ser tomada, ou seja, deve-se definir como ponderar dois objetivos de naturezas distintas. O primeiro objetivo refere-se ao nivelamento do consumo de componentes e o segundo, refere-se à eficiência da linha.

A eficiência da linha pode ser monitorada pelas divergências entre os tempos de operação. O quarto capítulo apresenta, em detalhes, a integração lógica dos métodos e recursos utilizados no modelo de sequenciamento proposto, incluindo o fluxograma de integração.

#### 6ª Etapa. Aplicação e análise do modelo de sequenciamento

Somente uma aplicação prática, utilizando exemplos numéricos, pode explicitar detalhes do modelo proposto.

Devido à extensão e complexidade deste tópico, o mesmo é abordado em um capítulo específico, o quinto capítulo, onde é apresentado um problema abrangente, ilustrando os detalhes do modelo proposto.

Conforme o estudo de Jahangirian et al. (2010), 44% das pesquisas sobre simulação, encontradas na literatura entre 1996 e 2007, apresentam casos hipotéticos, utilizando dados artificiais ou adaptados, de forma a buscar maior generalização do método e exploração das soluções obtidas. O problema apresentado no quinto capítulo trata de um caso observado em campo e adaptado para o formato acadêmico, ou seja, com redução do volume de dados e exploração de detalhes do modelo proposto.

### 3.3 Softwares utilizados

A evolução das técnicas de análise e pesquisa está diretamente associada à evolução dos recursos computacionais. No caso, visando à velocidade de processamento e rigor nos cálculos, pode ser indicada a aplicação de softwares específicos para as diversas funções apresentadas.

Por questões de afinidade, disponibilidade e adequação, os softwares específicos utilizados neste trabalho são os seguintes:

**Promodel** (PROMODEL, 2002): é uma ferramenta de simulação a eventos discretos, criada para modelar uma grande diversidade de sistemas de manufatura, abrangendo desde pequenas células até produção em grande escala, incluindo os sistemas de manufatura flexível e produção *Just in Time*.

Foi desenvolvido para o sistema operacional Windows, com uma interface gráfica intuitiva e orientada a objeto, o que simplifica muito a programação. O software permite que diferentes partes do modelo sejam implementadas por diferentes equipes e, posteriormente, sejam agrupadas em um grande modelo único.

Ele tem capacidade de animação, criação de macros, geração de relatórios e análise de cenários. A partir da modelagem, é possível identificar restrições de capacidade em sistemas produtivos e aprimorá-los visando à situação ideal.

O Promodel surgiu com a redução dos custos de equipamentos eletrônicos, e em consequência da demanda das empresas, que necessitavam de ferramentas que auxiliassem na resolução de problemas envolvendo aleatoriedade e interdependência.

Algumas características deste simulador, em relação aos demais, são: interface amigável e boa interação com o usuário, possibilidade de utilização de biblioteca de rotinas previamente construídas não sendo necessários profundos conhecimentos de programação, elaboração de relatórios em tempo real sobre a simulação executada e possibilidade de utilização de ferramentas estatísticas para tratamentos dos dados de aspecto estocástico.

O Promodel é fornecido em conjunto com outros dois programas: o **Simrunner**, para otimização de modelos, e o **Statfit**, para condicionamento de distribuições de probabilidade.

Para a construção de um modelo, o Promodel utiliza os seguintes elementos: **Locais, Entidades, Chegadas, Recursos e Processos**, encontrados no menu **Construir** do programa, cujas definições e funções básicas são as seguintes:

a) Os **Locais** representam pontos fixos no sistema onde se realizam os processos. São utilizados para representar elementos como: máquinas, postos de trabalhos, estoques, transportadores e filas. Para cada local pode-se definir: capacidade, quantidade, setups, manutenção e variáveis estatísticas, além de regras de seleção das entidades na entrada e regra de filas para as saídas;

b) As **Entidades** são os itens a serem processados pelo sistema representando matéria-prima, produtos, paletes, pessoas ou documentos. Para cada entidade pode-se definir velocidade, variáveis estatísticas e imagem. Elas podem ser agrupadas ou divididas ao longo do processo produtivo e são movimentadas por meios próprios ou por meio de recursos;

c) As **Chegadas** definem a entrada ou criação das entidades dentro do modelo. Para cada chegada são definidos a entidade, a quantidade, a frequência e o período,

bem como se pode especificar uma lógica de chegada. As chegadas podem ser definidas, também, por meio de um arquivo de chegada externo;

d) Os **Recursos** são os elementos utilizados para movimentar as entidades, executar operações, realizar manutenção dos locais ou outras atividades. Os recursos podem representar pessoas ou equipamentos. Em um sistema pode haver recursos estáticos ou dinâmicos. Os recursos estáticos não são dotados de movimento. Os recursos dinâmicos, por sua vez, podem se mover sobre um caminho definido por meio de uma rede;

e) Os **Processos** consistem em uma tabela onde são definidas as operações sofridas pelas entidades em cada local específico e uma sequência de rotas para movimentação destas entidades. É nesse elemento que se especificam os recursos necessários para realização de determinada tarefa, as esperas ocorridas ou outras lógicas como junção, carregamento, acúmulo, etc.

O Promodel dispõe, ainda, de elementos auxiliares como a função **Custo**, que define as taxas de custo dos recursos, dos locais e das entidades e a função **Turnos**, para definição de turnos de trabalho.

Essa ferramenta é utilizada neste trabalho com o objetivo de analisar as probabilidades de paradas das linhas, devido às dispersões nos tempos das operações, visto que uma das características deste software é a de operar com variáveis aleatórias em situação de alta interdependência.

**O Crystal Ball:** é um software instalado como uma ferramenta adicional do Microsoft Excel. Sua característica básica é operar variáveis aleatórias utilizando os fundamentos do método de Monte Carlo.

Como recurso auxiliar, apresenta uma vasta galeria de distribuições de probabilidade contínuas e discretas, com o objetivo de aleatorizar variáveis determinísticas, ou condicionar conjuntos de dados a variáveis aleatórias com distribuições contidas na galeria. Para a função de condicionamento de conjuntos de

dados, o **Crystal Ball** pode ser equiparado ao **Statfit**, incluso no pacote **Promodel**, porém sua galeria de distribuições, seus testes de significância e sua operacionalidade superam o **Statfit**, por isso, neste trabalho é dada a preferência ao **Crystal Ball**.

Como alternativa ao Promodel, podem ser utilizados outros softwares de simulação como o **Arena**, o **Witness** ou o **Simul8**.

**O Minitab:** é um software estatístico de ampla utilização em diversas áreas da indústria e em instituições de ensino e pesquisa. Trabalha em sistema de planilha, é altamente amigável e fornece toda a conceituação teórica necessária para sua utilização. O Minitab é utilizado neste trabalho para os testes de hipóteses estatísticos, aplicados na validação dos modelos de simulação.

**A Linguagem “C”:** é uma linguagem de propósito geral, sendo adequada à programação estruturada. No entanto, é mais utilizada para compiladores, analisadores léxicos, bancos de dados e editores de texto. Pertence a uma família de linguagens cujas características são: portabilidade, modularidade, compilação separada, recursos de baixo nível, geração de código eficiente, confiabilidade, regularidade, simplicidade e facilidade de uso (HANCOCK e KRIEGER, 1985). Devido a tais características, a linguagem “C” é aplicada neste trabalho, para a implementação da meta-heurística utilizada para o sequenciamento.

**O MS Office Excel:** de ampla utilização em meios empresariais e acadêmicos, a planilha eletrônica Excel apresenta vários recursos adequados aos cálculos executados neste trabalho, principalmente na aplicação do Goal Chasing. O Excel permite a aplicação automática de valores em funções, pelo simples arraste das células da planilha. Permite, também, a automatização de procedimentos por meio de **Macros** e, ainda, a ordenação e busca de resultados mínimos ou máximos, sendo de ampla utilização neste trabalho.



### 3.4 Classificação da pesquisa

Segundo Bertrand e Fransoo (2002), observam-se na literatura duas classes distintas de pesquisa baseada em modelos quantitativos: a pesquisa **axiomática** e a pesquisa **empírica**.

A **pesquisa axiomática** é dirigida a modelos de problemas previamente padronizados. Nesse caso, a preocupação fundamental do pesquisador é obter soluções para o modelo em questão e assegurar que tais soluções ajudem a esclarecer a estrutura do problema descrito.

A pesquisa axiomática quantitativa produz conhecimento sobre o comportamento de certas variáveis do modelo, baseado em premissas sobre o comportamento de outras variáveis. Ela pode também produzir conhecimento sobre como manipular certas variáveis, visando a um desempenho específico de outras, ou ainda basear-se em modelos existentes para desenvolver outros para problemas mais complexos, ainda não estudados.

Os pesquisadores dessa linha olham para o problema em questão, pelo ponto de vista do modelo matemático e, a partir dele, buscam melhorias e soluções. A pesquisa axiomática quantitativa, assim como a empírica, pode ainda ser subdividida em **normativa** ou **descritiva**, conforme o Quadro 2.

A pesquisa axiomática normativa desenvolve padrões, normas, políticas estratégias e ações para melhorar resultados disponíveis na literatura.

A pesquisa apresentada neste trabalho pode ser classificada como pesquisa **axiomática normativa**, no que se refere à primeira meta do modelo, ou seja, o sequenciamento com base no consumo constante de componentes. Já a segunda meta, a avaliação do perfil de eficiência das linhas, pode ser enquadrada como pesquisa **empírica descritiva**, visto que a mesma tem como base uma condição predefinida e a verificação do comportamento do sistema, por meio de um modelo de simulação, classificando a situação como aceitável ou não com relação a essa condição.

Quadro 2: Classificação do método de pesquisa “Modelagem e Simulação” (BERTRAND; FRANSOO, 2002).

Modelagem e Simulação		
	Descritiva	Normativa
Empírica	ED	EN
Axiomática	AD	AN

Reforçando a definição, na primeira meta, um modelo já existente é desenvolvido ou adaptado às condições específicas das linhas múltiplas, caracterizando a pesquisa **axiomática normativa**, enquanto na segunda meta é realizada uma análise empírica simulada, para identificação do perfil de eficiência das linhas, caracterizando a pesquisa **empírica descritiva**.

No desenvolvimento e aplicação da metodologia, os efeitos do sequenciamento são expressos em valores numéricos, classificando o trabalho como pesquisa quantitativa, sendo então analisados tanto pelo ponto de vista determinístico quanto pelo estocástico. Neste último, aplicando a simulação a eventos discretos pelo software Promodel.

A essência desse trabalho consiste no ajuste de variáveis de um modelo conhecido para um modelo expandido proposto, considerando as incertezas por meio da simulação, cujo objeto de estudo é uma situação mais abrangente: as linhas múltiplas.

O ajuste do modelo se dá pela análise individual das variáveis envolvidas, e a validação do modelo é obtida pela simulação determinística com a aplicação do software Promodel. Na pesquisa em questão, a importância da exploração e descrição dos métodos se sobrepõe à do objeto de pesquisa. A abordagem, no caso, sendo quantitativa, destaca a evidência de relações causais, operacionalização de conceitos e conclusões que, conforme o enfoque, podem ser generalizadas.

O conceito de pesquisa descritiva também se aplica ao caso, pois esta tem como objetivo a descrição de características de determinado processo estudado, ou o estabelecimento de relações entre variáveis.

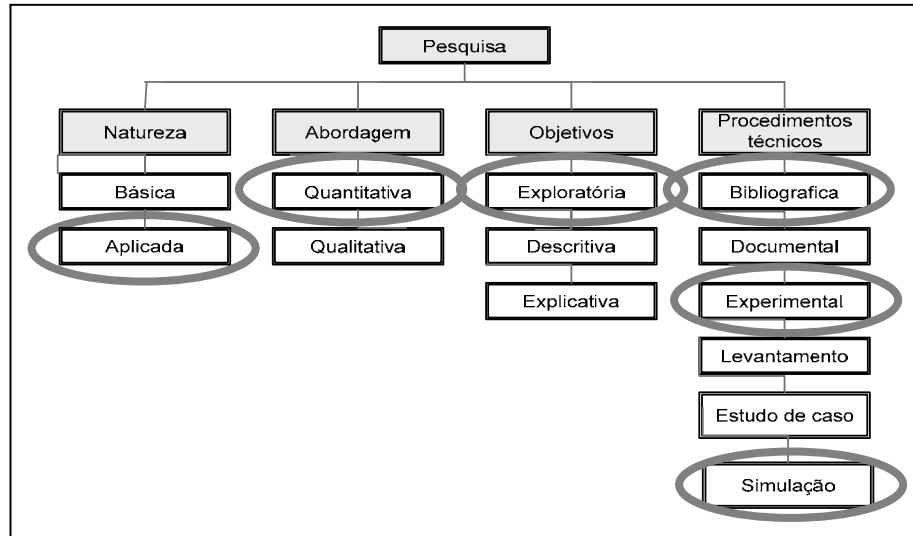


Figura 8 – Classificação da Pesquisa (adaptado de LEAL, 2008).

Algumas pesquisas descritivas vão além da simples identificação da existência de relações entre variáveis. Pretendendo-se determinar a natureza dessa relação, tem-se, então, uma pesquisa descritiva que se aproxima da explicativa. Há casos de pesquisas, que, embora definidas como descritivas, proporcionam uma nova visão do problema, o que as aproxima das pesquisas exploratórias.

### O *DOE* e o sequenciamento

Quando o tema em questão está relacionado à experimentação, otimização e simulação, um dos tópicos que surge em pauta é o **Delineamento de Experimentos** ou *Design of Experiments – DOE*. Porém, no caso de sequenciamento de linhas de montagem, a aplicação do DOE torna-se inviável pelos seguintes motivos:

a) O número de **fatores** e de **níveis** pode gerar uma quantidade de corridas tão expressiva, que impossibilite a aplicação do *DOE*, mesmo considerando os experimentos fracionados.

b) O sequenciamento com o objetivo de nivelamento do consumo é um processo totalmente determinístico. Portanto, não havendo aleatoriedade, não se justifica a aplicação do *DOE*. Convém destacar aqui, que a simulação, sim, possui aspecto estocástico, porém é utilizada apenas para verificação da eficiência associada a um

sequenciamento já estabelecido e não para estabelecer o sequenciamento. A simulação não é utilizada neste trabalho como um método de otimização.

c) A verificação da significância de fatores, que também pode ser obtida por meio do *DOE*, não é relevante, pois o método de otimização do sequenciamento possui como fatores, os momentos “k”, em que os diferentes modelos entram na linha, ou seja, todos são igualmente significantes. Já os níveis que podem ser assumidos por cada fator, são os diferentes produtos ou modelos a serem inseridos nas linhas a cada momento “k”.

Por exemplo, uma linha com uma demanda diária total de 20 itens (fatores), com 5 modelos distintos (níveis), em um *DOE* fatorial completo geraria  $5^{20} = 9,54 \cdot 10^{13}$  corridas, sendo necessário, ainda, desconsiderar as corridas inviáveis, que desrespeitam as restrições de demandas individuais.

### 3.5 Considerações finais

A importância metodológica de um trabalho de pesquisa, principalmente a pesquisa quantitativa, pode ser justificada pela necessidade de consistência dos modelos propostos e pela confiabilidade dos métodos de solução. Nesse aspecto, a metodologia apresentada procura, de forma abrangente, detalhar os métodos, técnicas e recursos utilizados, com a finalidade de alinhar cada ferramenta com o objetivo final do trabalho.

Somente a detalhada descrição das etapas do desenvolvimento do método proposto, inserida num contexto de rigor matemático e conceitual, pode embasar conclusões sólidas, verdadeiras contribuições à comunidade científica.

## CAPÍTULO 4 - DESENVOLVIMENTO DO MODELO PROPOSTO PARA SEQUENCIAMENTO DE LINHAS MÚLTIPLAS

### 4.1 Considerações iniciais

O presente capítulo mostra a execução das etapas do desenvolvimento do modelo proposto para sequenciamento de linhas múltiplas. Com base na delimitação do tema, apresentada no primeiro capítulo, na fundamentação teórica do segundo capítulo e na base metodológica do terceiro capítulo, os conceitos encontrados na literatura são analisados e concatenados, de forma a buscar uma solução eficiente para um problema de sequenciamento real, observado na área de manufatura de empresas que operam em ambiente de produção enxuta.

Considerando os aspectos determinísticos e estocásticos, quando necessário, o modelo a seguir se apresenta como uma proposta de um procedimento para auxiliar no sequenciamento de LMMM múltiplas em ambiente de produção enxuta, considerando as incertezas envolvidas nos tempos de operação e utilizando, entre outros recursos, a simulação a eventos discretos.

### 4.2 Análise das características dos métodos de sequenciamento pesquisados (1ª Etapa)

A revisão bibliográfica mostrada no segundo capítulo apresenta grande ênfase nas **metas** do sequenciamento das linhas de montagem e no **nível de detalhamento** dessas linhas. São essas as características em destaque neste tópico.

#### As metas do sequenciamento

Com base nos trabalhos pesquisados e nas observações de campo, as metas definidas para o modelo de sequenciamento proposto, são duas:

**Primeira Meta:** Nivelamento do consumo de componentes, considerando cada linha como bloco único, ou seja, comprimento zero, seguindo a definição de Boysen et

al. (2009) como problema de **programação nivelada**, conforme os princípios da produção enxuta, e com base nos conceitos do método *Goal Chasing*, equacionado por Mondem (1983).

A primeira meta consiste em minimizar a somatória das distâncias “ $D_k$ ” para o máximo valor de “ $k$ ”, onde “ $D_k$ ” representa a distância entre o consumo real e o ideal a cada momento normalizado “ $k$ ” ( $k = 1, 2, \dots, K$ ) do período de produção.

Essa meta pode ser considerada como uma adaptação da primeira meta do *GC*, proposto por Mondem (1983), onde a minimização das distâncias ocorre parcialmente a cada momento de entrada de um produto na linha, ou seja, a cada momento “ $k$ ”.

**Segunda Meta:** Adequação estocástica da eficiência da linha, estabelecendo um perfil aceito. Nestas condições, a eficiência da linha é avaliada por uma variável quantitativa contínua, o **tempo improdutivo**, que pode ser representada por valores absolutos como horas, minutos e segundos, ou por valores relativos, como porcentagem da jornada diária. Neste trabalho, são utilizados valores relativos, por apresentar compatibilidade com os relatórios emitidos pelo software Promodel.

A eficiência da linha pode ser monitorada pelas divergências entre os tempos de operação dos postos de trabalho, caracterizados pelas **esperas** ou pelos **bloqueios** que podem ocorrer na linha.

Entende-se, aqui, por **espera**, o tempo improdutivo no qual determinado posto, que já concluiu sua operação, aguarda a chegada do próximo produto em processo. E, por **bloqueio**, o tempo improdutivo no qual determinado posto de trabalho, que já concluiu sua operação, aguarda a liberação do posto seguinte para o envio do produto corrente em processo.

O aspecto estocástico da simulação pode ser explicado pelo seguinte exemplo: supondo que determinada sequência de produção seja simulada, ainda de forma determinística, ou seja, tempos com dispersão nula e uma única corrida, o resultado do tempo total ineficiente será um único valor.

Já considerando a simulação estocástica, ou seja, tempos representados por variáveis aleatórias e “ $n$ ” corridas, os tempos totais ineficientes serão representados

por “n” valores, e tais valores podem ser condicionados a uma variável aleatória, com distribuição de probabilidade padronizada.

Neste trabalho, é utilizada a análise por intervalos de confiança bicaudais, por questão de adequação ao software Promodel, visto que o mesmo não fornece intervalos de confiança monocaudais. Porém, nos casos onde há necessidade da análise monocaudal, pode-se estabelecer a relação de equivalência entre intervalos mono e bicaudais.

Um valor extremo de um intervalo de confiança bicaudal simétrico, com nível de significância “ $\alpha$ ”, é igual ao extremo de um intervalo de confiança monocaudal com nível de significância “ $\alpha/2$ ” (TRIOLA, 2005).

Exemplificando: se determinado intervalo de confiança bicaudal, com NS=10% possui um valor extremo superior igual a “10,15”, para os mesmos dados, o intervalo de confiança monocaudal, com NS=5% também terá como limite superior o valor “10,15”.

Portanto, para que a restrição da segunda meta seja respeitada, as variáveis aleatórias, tempo de espera (*waiting*) e tempo de bloqueio (*blocked*), apresentadas no relatório da simulação, devem exibir como resposta um intervalo de confiança com limite superior máximo, menor que um valor predefinido e um nível de significância (ou confiança) controlado.

#### O nível de detalhamento da linha

O nível de detalhamento da linha, ou seja, a divisão da linha em postos de trabalho, também depende de cada meta.

O nivelamento do consumo, primeira meta, pode ser obtido com baixo nível de detalhamento (bloco único), pois simplesmente busca o consumo constante de componentes, independentemente do local ou momento específico em que o componente é solicitado.

A análise estocástica da carga de trabalho, por outro lado, necessita da divisão da linha de montagem em postos, definindo claramente a ordem das tarefas e os

respectivos tempos estocásticos, devido à interdependência entre as operações de cada posto, conforme apresentado por Xiaobo e Ohno (1997).

Os detalhes relativos aos postos de trabalho devem ser apresentados na etapa de simulação computacional, conforme a Figura 4, apresentada no segundo capítulo, no método caracterizado como *Modified Goal Chasing – MGC*.

#### 4.3 Ajuste do modelo para linhas múltiplas (2ª Etapa)

Para a modelagem matemática do problema de sequenciamento para linhas múltiplas é realizado o ajuste e adaptação, a partir do modelo de linhas únicas, que é um problema previamente conhecido.

Nessas condições, deve-se analisar o comportamento de cada variável envolvida, individualmente e em conjunto, visando à obtenção da **função objetivo** e das respectivas **restrições** para linhas múltiplas.

O modelo matemático utilizado como base para o desenvolvimento do modelo proposto é o *Goal Chasing* para a primeira meta, e o *Extended Goal Chasing* para a segunda meta, ou seja, diferem, conceitualmente, pelo nível de detalhamento necessário da linha de montagem.

Considerando a primeira meta, o modelo proposto para sequenciamento de linhas de montagem múltiplas é representado pela Figura 9, onde se observa que todos os fornecedores (*kanbans*) podem abastecer simultaneamente todas as linhas. Dessa forma, o consumo de cada componente deve ser representado pela somatória dos consumos individuais de cada linha a cada momento “k”.

O principal objetivo é determinar simultaneamente para todas as linhas, a sequência com que os produtos devem ser programados, visando à máxima aproximação do consumo constante de componentes e a adequação a um perfil de eficiência predeterminado para as linhas.



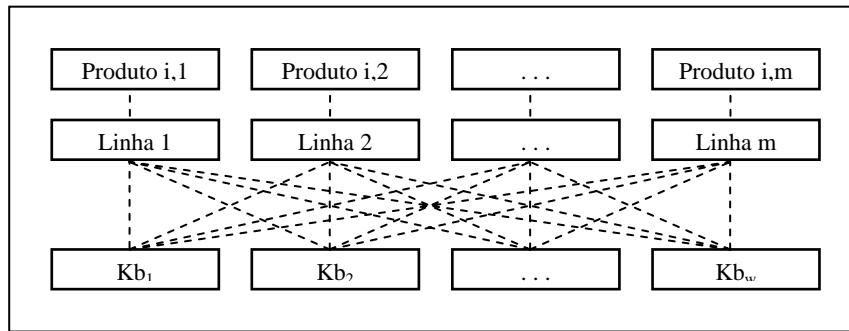


Figura 9 – Linhas de montagem múltiplas para modelos mistos

A partir desse ponto, o problema toma dois rumos distintos: o primeiro segue a linha da pesquisa **axiomática normativa**, onde é analisado o modelo de Monden para linhas únicas, sendo realizados os ajustes e adaptações necessários para a programação das linhas múltiplas, o segundo, segue a linha de pesquisa **empírica descritiva**, com a aplicação da simulação a eventos discretos.

#### Análise dos ajustes necessários

Primeiramente, deve-se inserir no modelo proposto a variável relativa ao índice de linha “m” ( $m = 1, \dots, M$ ). Os demais ajustes são descritos a seguir.

K: Contador de tempo normalizado ( $k = 1, 2, \dots, K$ ). O contador de tempo normalizado (número inteiro) não necessita de ajustes, pois conforme citado no primeiro capítulo, considera-se, aqui, a condição de demandas totais diárias iguais para todas as linhas.

Tratando-se de linhas de montagem de modelos semelhantes, os processos também podem ser semelhantes e seus tempos aproximados. Portanto, as quantidades totais produzidas por período (dia ou turno), em cada linha, podem ser consideradas aproximadamente iguais. Ainda cabe destacar que, como consequência, ocorre a igualdade representada pela equação (7), ou seja, a somatória das demandas em cada linha é igual ao máximo valor de “k”.

$$K = \sum_{i=1}^I Q_{im}, \quad \forall m \tag{7}$$

$N_j$ : Consumo ideal do componente “j” por unidade de tempo. Esse pode ser considerado o principal ajuste do modelo proposto, pois trata da principal distinção entre o modelo de linhas únicas com relação a linhas múltiplas, o **consumo múltiplo**, conforme ilustrado na Figura 10. O consumo ideal total é dado pela somatória dos consumos de todas as linhas, conforme apresentado na equação (8):

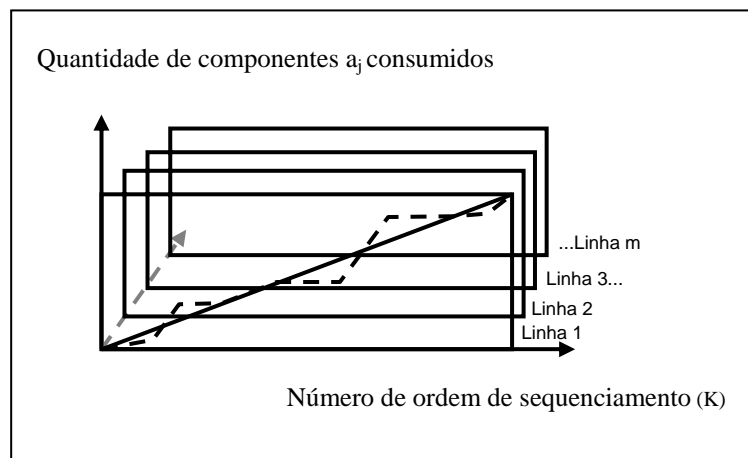


Figura 10 - Consumo de determinado componente “a<sub>j</sub>” em linhas múltiplas

$$[N_j] = \sum_{m=1}^M [Q_{im}] [b_{imj}] \tag{8}$$

Expandindo o conceito para todos os componentes  $a_j$ , é possível visualizar a extensão do problema pela Figura 11.

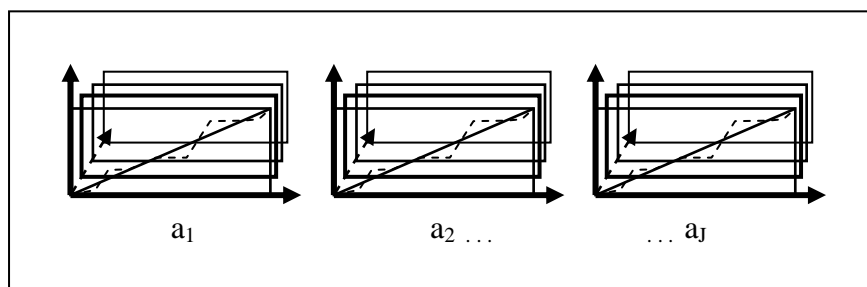


Figura 11 - Consumo de vários componentes em linhas múltiplas.

$j$ : Índice de componente ( $j = 1, \dots, J$ ). O índice de componente é uma variável que não sofre alteração no modelo proposto, pois o número e diversidade de componentes não é função do número de linhas de montagem.

$A_i$ : produtos a serem programados na linha de montagem. Essa variável deve ser ajustada, recebendo um índice para sua respectiva linha de montagem, considerando que, no modelo proposto, não há possibilidade de alteração de linha. Cada produto só pode ser montado na sua respectiva linha “ $m$ ”, não sendo intercambiável. Portanto, a variável “ $A_i$ ” passa a ser “ $A_{im}$ ” no modelo proposto.

$Q_i$ : Demanda Diária do Produto “ $i$ ”. Como consequência da inserção do índice “ $m$ ” na variável “ $A_i$ ” (tornando-se “ $A_{im}$ ”) a demanda de cada produto também deve receber o índice de linha, ou seja, torna-se “ $Q_{im}$ ”.

$i$ : Índice de Produto ( $i = 1, \dots, I$ ). O índice de produto não sofre alteração, no modelo proposto, mas, quando necessário, passa a ser acompanhado do índice de linha “ $m$ ”.

$b_{i,j}$ : Consumo do componente “ $j$ ” no momento “ $k$ ”. Também como consequência da inserção do índice “ $m$ ”, o consumo de cada componente deve receber o índice de linha, ou seja, torna-se “ $b_{ijm}$ ”.

$X_{j,k}$ : Consumo parcial do componente “ $j$ ” até o momento “ $k$ ”. O consumo parcial de cada componente passa a ser computado, não mais com o consumo “ $b_{ij}$ ”, mas sim com a somatória, para todas as linhas, dos consumos “ $b_{ijm}$ ”, conforme mostrado pela equação (9):

$$X_{jk} = X_{j,k-1} + \sum_{m=1}^M b_{i^*jm} \quad (9)$$

$D_{k,i,m}$ : Distância entre o consumo real e o consumo ideal no momento “ $k$ ”, adotando para o momento “ $K$ ” a entrada do produtos “ $i$ ” para cada linha “ $m$ ”, representado pela equação (10).

$$\min D_{k,i,m} = \sqrt{\sum_{j=1}^J \left( \frac{K \cdot N_j}{\left( \sum_{i=1}^I \sum_{m=1}^M Q_{im} \right) / M} - X_{j,k-1} - \sum_{m=i}^M b_{ijm} \right)^2} \quad (10)$$

Tratando-se de demandas totais iguais para todas as linhas, caso em questão, a equação (10) pode ser simplificada para a equação (11):

$$\min D_{k,i,m} = \sqrt{\sum_{j=1}^J \left( \frac{K \cdot N_j}{Q} - X_{j,k-1} - \sum_{m=i}^M b_{ijm} \right)^2} \quad (11)$$

O modelo proposto pela equação (11), aqui denominado **Goal Chasing Adaptado GCA**, ainda apresenta características de otimização parcial, não sendo, até então, apresentada a otimização global para a soma das distâncias  $D_k$ .

Convém destacar, também, que, na aplicação do método GCA, para o sequenciamento de cada produto  $A_{im}$ , faz-se necessária a restrição apresentada na equação (12).

$$\sum_k A_{im} \leq Q_{im} \quad (12)$$

Tal restrição, inexistente no caso de linhas únicas, deve-se à possibilidade de divergência do consumo total com relação à demanda de determinado produto, ou seja, **em média**, o componente programado está limitado à demanda máxima, mas, individualmente, pode estar excedendo tal demanda. A Figura 12 ilustra a situação descrita.

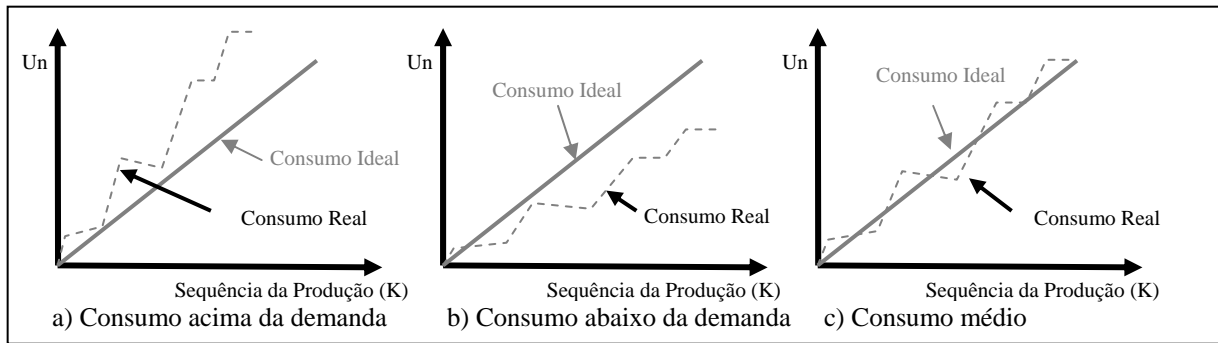


Figura 12 – Divergência entre consumo médio de componentes real e ideal.

### A otimização global

Para a otimização global, ou seja, a minimização da soma das distâncias “ $D_k$ ”, apresentada na equação (13), faz-se necessária a aplicação de uma ferramenta computacional avançada, que supere as limitações do GCA.

$$\min \sum_{k=1}^K D_k \quad (13)$$

Para a verificação da condição da segunda meta, é utilizada a simulação a eventos discretos, onde os tempos de operação são tratados como variáveis aleatórias contínuas, condicionadas a partir de dados coletados e inseridos no sistema modelado no software Promodel.

Tal situação pode ser avaliada nos relatórios do Promodel pelas variáveis **espera** (*Waiting*) e **bloqueio** (*Blocked*). A espera indica que determinado posto de trabalho terminou sua operação corrente e está aguardando a chegada do próximo produto em processo. A espera pode ser considerada uma ineficiência sem grandes consequências por ser pontual e não afetar a linha toda. Já o bloqueio indica que certo posto está parado devido à sobrecarga do posto subsequente, ou seja, se o posto seguinte não termina sua operação dentro de determinado limite de tempo, o posto anterior fica impossibilitado de enviar o produto corrente, o que pode, em certas condições, paralisar a linha toda.

#### 4.4 Seleção do método de busca de soluções eficientes (3ª Etapa)

As heurísticas analisadas, que são basicamente variações do Toyota Goal Chasing, exigem pouco tempo computacional e apresentam implementação relativamente simples, porém a eficiência e qualidade da resposta são discutíveis, devido à otimização parcial a cada momento “k” de entrada de um novo produto na linha.

A programação dinâmica se apresenta como um método eficiente para problemas de sequenciamento. Porém, sua implementação e elaboração para sequenciamento de linhas múltiplas pode se tornar excessivamente complexa, apesar de possuir a característica de transformar um problema intratável, com tempo computacional exponencial, em um problema tratável, com tempo computacional polinomial ou superpolinomial (CORMEN et al., 2002).

Devido à extensão e complexidade do problema em estudo, caso com linhas múltiplas, os métodos relacionados à programação dinâmica perdem espaço para as meta-heurísticas, que apesar de não garantirem soluções ótimas, são mais práticas, mais ágeis em termos computacionais e permitem maior facilidade no ajuste e adaptação das **funções objetivo**.

Também convém destacar que as meta-heurísticas podem ser aplicadas em problemas de qualquer dimensão, mesmo nos problemas considerados intratáveis pela programação dinâmica.

Entre as principais meta-heurísticas encontradas na literatura, com possibilidade de aplicação no sequenciamento de LMMM, pode-se destacar a **Busca Tabu**, como apresentado em McMullen (1998), o *Simulated Annealing*, em McMullen e Frazier (2000) e os **Algoritmos Genéticos**, em Leu et al. (1996).

Com relação à qualidade da resposta, para seleção de uma das meta-heurísticas apresentadas, é necessária a aplicação de todas elas e, conseqüentemente, a comparação dos resultados obtidos, o que não é o objetivo deste trabalho.

Apesar da possibilidade de utilização de qualquer uma das três meta-heurísticas citadas, o *Simulated Annealing* é o método selecionado, simplesmente por uma questão de afinidade.

O SA consiste em partir de uma solução inicial qualquer e, em uma dada temperatura, realizar movimentos aleatórios na solução corrente gerando soluções vizinhas. Computa-se a variação da função objetivo a cada solução vizinha gerada. Se esta melhorar a solução corrente, o movimento é aceito, caso contrário, existe uma probabilidade de aceitação do movimento que é menor em temperaturas baixas.

A representação de uma solução do problema consiste em uma matriz na qual cada linha, dessa matriz, indica uma linha de montagem e a posição de cada elemento, nas linhas da matriz, indica a ordem na qual determinado produto é inserido nessas linhas de montagem.

Para compor vizinhanças distintas, dois movimentos aleatórios podem ser realizados, a troca de posição de dois produtos aleatórios na mesma linha, ou a realocação de um produto aleatório de forma que os demais produtos subsequentes ocupem sua posição.

Na aplicação do SA, para aprimorar o processo de convergência, são utilizados diferentes parâmetros conforme a temperatura corrente, possibilitando um resfriamento mais lento no início do processo e um maior número de iterações em baixas temperaturas. Tais parâmetros são: Para temperaturas maiores que 1000,  $\alpha = 0,98$  e  $SA_{\max} = 1000$ . Para temperaturas maiores que 100 e menores ou iguais a 1000,  $\alpha = 0,95$  e  $SA_{\max} = 1500$ . Para temperaturas menores ou iguais a 100,  $\alpha = 0,92$  e  $SA_{\max} = 2000$ . Além disso, para um melhor refinamento da solução, o processo realiza um aquecimento, ou seja, quando a temperatura se torna inferior a 0,01, a mesma é elevada a 20% da temperatura inicial.

#### 4.5 Elaboração do modelo de simulação computacional (4ª Etapa)

O trabalho desenvolvido nesta etapa toma como base a proposta apresentada em Shannon et al. (2010), onde são descritas sugestões para implementação da filosofia e estratégias da produção enxuta, em modelos de simulação a eventos discretos, permitindo a adaptação a qualquer software de simulação.

De posse do modelo conceitual teórico e dos dados condicionados pelo software **Crystal Ball**, passa-se à elaboração do modelo computacional pelo software

**Promodel**, sendo, então, definidos os elementos: Locais, Entidades, Recursos, Processos e Chegadas.

**Locais:** São definidos como locais, os *kanbans* de componentes e os postos de trabalho que compõem cada linha, devidamente identificados. Nesta fase da elaboração do modelo, é interessante inserir um contador visível, disponível no Promodel, na galeria de recursos denominada **Imagens**, desabilitando o comando **Novo**. Também deve ser inserida a informação sobre a capacidade de cada *kanban*, na coluna **Cap**. As demais colunas devem permanecer inalteradas.

**Entidades:** As entidades são os componentes “ $a_j$ ” e os produtos a serem programados em cada linha “ $A_{im}$ ”. As demais informações dessas entidades, como **velocidade, estatísticas e anotações** são desnecessárias.

**Recursos:** Não é necessária a inserção de recursos como mão de obra ou esteiras, visto que o objetivo do modelo é analisar a interdependência entre as operações, considerando a aleatoriedade dos tempos de execução dessas operações. A partir da informação de velocidade da esteira e comprimento do posto de trabalho, pode-se simplificar o modelo e limitar a operação pela variável tempo disponível, visto que o mesmo é uma consequência da velocidade da esteira e comprimento do posto.

**Processos:** Os processos se iniciam com as entradas dos componentes nos *kanbans* e dos produtos a serem montados nas respectivas linhas. Conforme a necessidade de componentes, os comandos *Join* e *Wait* indicam as peças que serão unidas e o tempo dessa união, considerando o tempo de operação como variável aleatória, com distribuição de probabilidade definida e condicionada a partir de dados coletados.

**Chegadas:** São as responsáveis pelo sequenciamento. É por meio das chegadas agendadas em mês, dia, ano, hora e minuto, que se define a sequência com que os produtos são inseridos na linha. Para determinado modelo simulado, não é necessário alterar a lógica para alterar a sequência, basta alterar o momento de entrada dos produtos para uma nova análise. A Figura 13 mostra as janelas do Promodel destinadas ao agendamento das entradas.



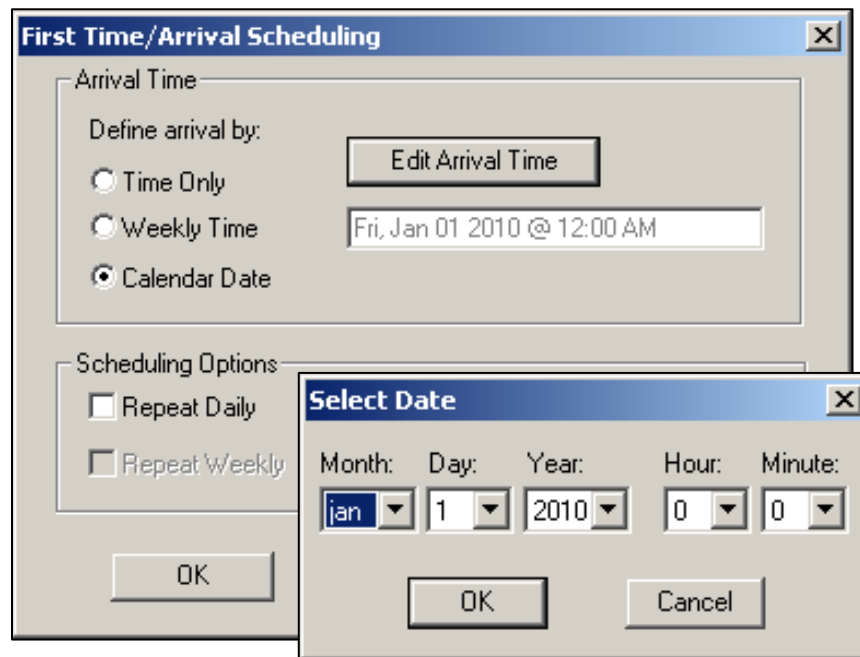


Figura 13 – Definição da sequência de montagem no software Promodel.

#### 4.6 Integração lógica das etapas do sequenciamento (5ª Etapa)

Com base nos objetivos da produção enxuta, o método deve focar, de forma prioritária, o nivelamento do consumo de componentes e adotar um tempo de ciclo baseado nos tempos padronizados de produção, não sendo utilizadas as dispersões no cálculo do tempo de ciclo.

Também se deve considerar que o sequenciamento focado no nivelamento de componentes, automaticamente, auxilia no amortecimento das sobrecargas de trabalho nos postos de trabalho mais críticos.

O algoritmo, apresentado pela Figura 14, integra as duas metas do problema por meio de um processo iterativo, onde ambas as metas devem ser alcançadas. Caso a primeira sequência do ranqueamento não satisfaça a segunda meta, pode-se adotar a seguinte no ranking, o que já pode influenciar o tempo total ineficiente e atender a segunda meta.

Dessa forma, a lógica pode ser seguida até que se satisfaçam as duas metas. Um inconveniente desse algoritmo é a dificuldade de implementação integral em

programação computacional, pois a simulação envolve alteração de parâmetros específicos como datas e horários de produção, que exigem a intervenção humana por parte do usuário ou programador do Promodel.

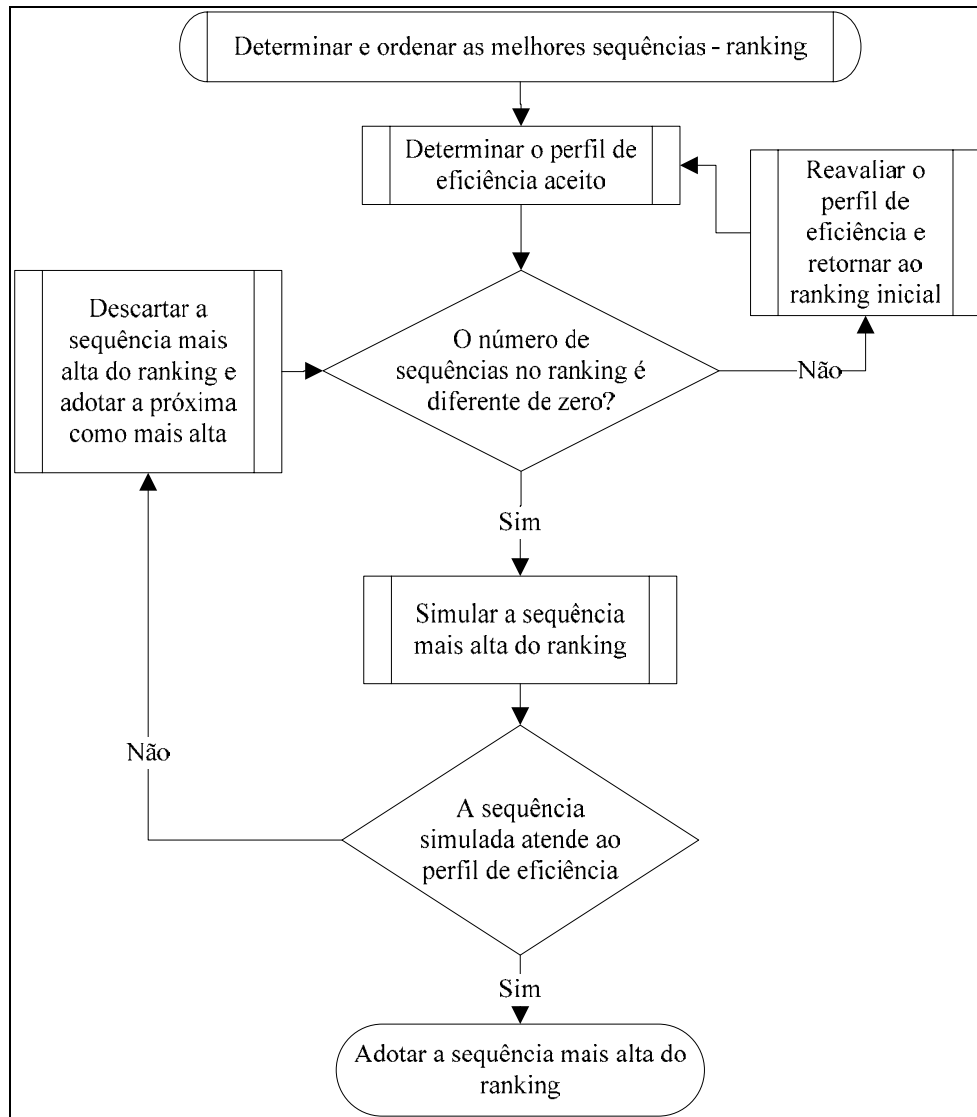


Figura 14 - Modelo proposto

De forma semelhante ao modelo proposto, o recente trabalho de Al-Aomar (2010), destaca, conforme Figura 15, o sistema de realimentação de informações, possível num modelo híbrido, coincidentemente envolvendo a simulação a eventos discretos, o *Simulated Annealing* e o ambiente de produção enxuta.

O autor apresenta um sistema cíclico envolvendo a otimização da alocação de recursos, visando à minimização de *WIP* e *Takt Time*, onde as respostas ótimas e sub-

ótimas são simuladas pelo software **Witness**, para verificação de restrições de ordem prática e realimentação de dados para um aprimoramento da resposta obtida pelo *Simulated Annealing*.

O modelo apresentado por Al-Aomar (2010) pode ser considerado um reforço à validação e generalização do modelo proposto neste trabalho, por apresentar o uso integrado de ferramentas semelhantes para otimização e simulação, e também por estar inserido num contexto de produção enxuta.

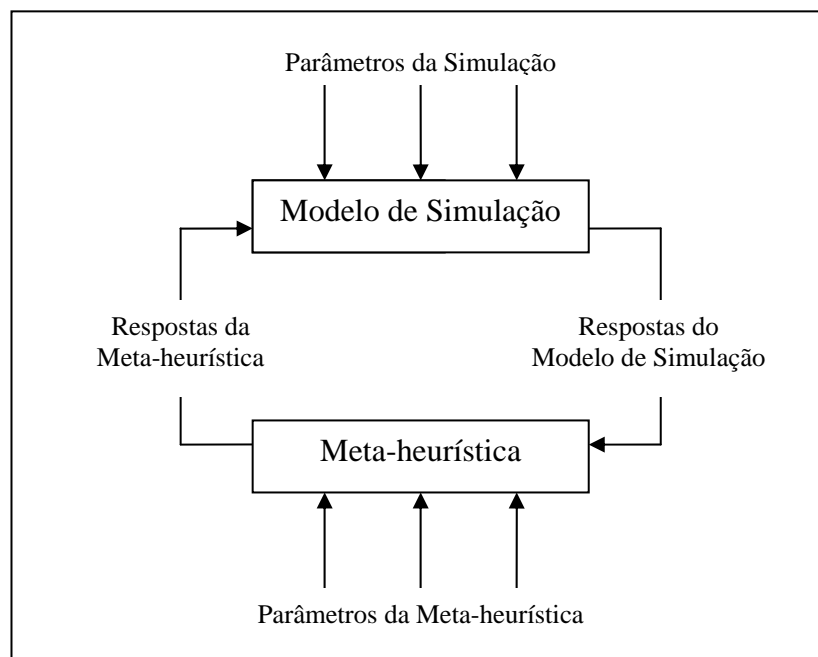


Figura 15 – Modelo híbrido envolvendo Simulação e Meta-heurística (adaptado de AL-AOMAR, 2010)

#### 4.7 Considerações Finais

O modelo proposto integra, de forma lógica, as técnicas de busca de duas metas de naturezas distintas, caracterizando um método híbrido.

A estratégia de produção, adotada pela empresa ou pelos potenciais usuários do modelo proposto, pode sugerir alterações no critério de formação do ranking de sequências de produção, ou na forma de simular as sequências do ranking.

Uma das maneiras de se implementar a proposta é apresentada, a seguir, no quinto capítulo, em uma aplicação numérica.

## CAPÍTULO 5 - APLICAÇÃO DO MODELO PROPOSTO PARA SEQUENCIAMENTO DE LINHAS MÚLTIPLAS

### 5.1 Considerações Iniciais

Neste capítulo, é apresentado um exemplo numérico, mostrando as características e a aplicação do modelo proposto, incluindo as respectivas análises.

O problema em questão apresenta um caso observado em campo e adaptado para o formato acadêmico, ou seja, com redução do volume de dados e destaque para a exploração de detalhes do modelo proposto.

O problema envolve, entre outros recursos, a aplicação dos métodos **Goal Chasing Adaptado - GCA**, *Simulated Annealing - SA* e **Simulação a Eventos Discretos**. O exemplo busca explorar o método de sequenciamento proposto visando a uma possível generalização do modelo. Neste capítulo é utilizada a notação descrita no quarto capítulo.

### 5.2 Exemplo numérico

Uma unidade produtiva que opera com três linhas de produção ( $M = 3$ ), deve produzir, em uma jornada diária de 20 horas, as demandas indicadas na Tabela 1. O consumo dos componentes críticos ( $a_j$ ) de cada produto ( $A_{im}$ ) é indicado na Tabela 2. As linhas de montagem possuem cinco postos de trabalho, onde são montados os componentes críticos. Os dados coletados, relativos aos tempos de operação, são apresentados na Tabela 3. O perfil de eficiência aceito é de, no máximo, 5% para espera e 15% para bloqueio, por jornada diária. Deseja-se determinar uma sequência de montagem eficiente, que busque o nivelamento do consumo de componentes e atenda ao perfil de eficiência. Deseja-se obter o sequenciamento pelo método *GCA* e pela meta-heurística *SA*, comparar os resultados e analisar a eficiência das melhores respostas pelo software Promodel, utilizando o modelo de sequenciamento proposto no quarto capítulo. Utilizar um ranking de seis sequências e nível de confiança  $NC=95\%$ .

Tabela 1 - Matriz de Demandas ( $Q_i$ )

Linha (m) \ Produto ( $A_j$ )	$A_{0m}$	$A_{1m}$	$A_{2m}$	$A_{3m}$	$Q = \sum Q_i$
1	2	3	5	10	20
2	4	12	2	2	20
3	5	5	4	6	20

Tabela 2 - Matriz de Componentes ( $a_j$ )

Produto ( $A_{im}$ ) \ Componente $a_j$		$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
Linha 1	$A_{01}$	1	2	1	3	0
	$A_{11}$	2	1	0	0	2
	$A_{21}$	3	2	1	0	2
	$A_{31}$	0	0	3	0	3
Linha 2	$A_{02}$	2	3	1	2	0
	$A_{12}$	2	1	2	1	2
	$A_{22}$	0	2	1	2	0
	$A_{32}$	3	2	3	2	1
Linha 3	$A_{03}$	1	0	1	0	1
	$A_{13}$	3	2	1	1	3
	$A_{23}$	2	1	2	1	0
	$A_{33}$	0	1	2	1	2

Tabela 3 – Tempos de operação por componente/quantidade (min.) e respectivas distribuições de probabilidade (LN: Distribuição Lognormal)

$n/a_j$	$a_1$ (1)	$a_1$ (2)	$a_1$ (3)	$a_2$ (1)	$a_2$ (2)	$a_2$ (3)	$a_3$ (1)	$a_3$ (2)	$a_3$ (3)	$a_4$ (1)	$a_4$ (2)	$a_4$ (3)	$a_5$ (1)	$a_5$ (2)	$a_5$ (3)
1	46,70	48,47	51,64	42,74	41,78	43,19	46,94	47,93	52,72	43,34	44,54	46,27	41,31	43,14	46,66
2	47,56	49,33	49,84	40,97	41,90	43,31	47,08	47,97	52,77	42,44	44,16	45,78	41,48	46,25	48,88
3	46,29	48,22	50,91	43,13	41,80	43,21	46,97	48,05	52,86	42,25	45,33	45,88	41,52	46,93	47,32
4	45,89	47,82	50,35	41,01	41,80	43,21	46,97	47,94	52,74	42,85	44,42	45,61	41,41	47,81	46,57
5	45,48	48,37	50,17	41,39	41,88	43,29	47,06	48,04	52,84	42,87	44,82	45,36	41,33	44,71	46,69
6	45,40	47,81	50,35	41,24	42,00	43,42	47,19	47,79	52,57	43,37	45,02	47,00	41,76	48,33	47,12
7	45,76	47,88	50,64	40,65	42,03	43,45	47,22	47,91	52,70	42,40	44,81	45,66	41,35	43,48	46,57
8	47,63	48,13	51,07	41,73	41,92	43,34	47,11	47,81	52,60	43,61	48,67	47,58	41,45	50,55	47,43
9	46,54	48,02	50,10	41,22	41,71	43,12	46,87	47,96	52,75	42,64	44,89	45,79	43,80	43,44	47,31
10	47,53	49,37	49,66	40,97	41,78	43,19	46,94	48,10	52,91	42,51	44,37	47,22	41,29	45,71	46,96
11	46,17	47,49	50,02	45,16	41,74	43,15	46,90	47,84	52,62	43,11	44,70	45,28	44,32	44,53	48,57
12	46,95	48,88	52,32	40,50	41,83	43,24	47,00	48,23	53,05	44,95	44,84	45,73	41,74	44,86	47,45
13	45,54	47,68	50,92	41,42	41,82	43,23	46,99	48,10	52,91	43,02	44,40	45,32	42,06	43,92	46,53
14	46,12	47,97	50,22	41,38	41,91	43,32	47,09	48,03	52,83	48,92	46,19	45,54	44,87	43,51	47,29
15	46,83	48,12	49,74	40,45	41,81	43,22	46,98	48,27	53,09	43,19	45,29	46,98	41,08	44,12	46,75
16	45,76	49,30	49,93	41,46	41,97	43,39	47,16	47,91	52,71	43,08	44,97	45,82	42,75	43,39	46,30
17	46,19	48,17	49,57	41,85	41,89	43,31	47,07	48,00	52,79	43,04	46,13	45,68	41,89	43,91	46,85
18	47,85	48,42	52,46	47,40	41,86	43,27	47,04	48,04	52,84	43,14	45,01	46,42	41,37	43,36	46,57
19	46,15	48,15	51,56	42,14	41,79	43,20	46,96	47,94	52,73	42,69	44,51	47,36	41,43	43,73	46,55
20	45,59	48,38	49,35	40,86	41,99	43,41	47,18	48,02	52,83	42,81	45,05	45,90	43,76	44,45	47,07
21	45,93	48,60	49,47	40,44	41,86	43,27	47,04	47,98	52,77	44,22	44,63	45,88	41,40	44,01	46,67
22	46,10	48,05	49,90	41,55	42,01	43,42	47,20	48,10	52,91	42,95	45,96	45,78	42,62	46,49	46,32
23	47,20	47,96	50,38	41,86	41,91	43,32	47,09	48,04	52,84	42,88	46,05	45,54	42,34	43,36	47,17
24	45,74	47,73	49,35	41,43	41,77	43,18	46,94	48,04	52,84	46,03	44,43	45,26	41,95	46,32	46,69
25	46,22	47,70	51,31	41,13	41,85	43,26	47,02	48,14	52,95	42,78	46,20	45,59	44,22	46,07	46,77
26	45,60	47,96	49,98	41,63	41,93	43,35	47,12	47,93	52,72	42,96	44,23	45,64	42,51	45,19	46,66
27	45,95	49,02	50,61	42,90	41,85	43,26	47,02	48,15	52,96	42,42	46,37	45,76	42,79	44,33	47,06
28	46,63	47,86	50,13	40,11	41,80	43,21	46,96	47,95	52,75	43,11	44,89	45,96	41,40	52,47	47,37
29	46,58	47,84	50,15	44,62	41,78	43,19	46,94	48,16	52,98	42,39	44,98	47,47	41,90	43,50	47,57
30	45,80	48,00	49,43	42,86	41,94	43,35	47,12	48,17	52,99	42,87	44,55	45,74	42,14	43,62	47,49
$\mu$	<b>46,32</b>	<b>48,22</b>	<b>50,39</b>	<b>41,87</b>	<b>41,86</b>	<b>43,28</b>	<b>47,04</b>	<b>48,02</b>	<b>52,82</b>	<b>43,29</b>	<b>45,15</b>	<b>46,03</b>	<b>42,18</b>	<b>45,18</b>	<b>47,04</b>
$\sigma$	<b>0,69</b>	<b>0,5</b>	<b>0,82</b>	<b>1,5</b>	<b>0,8</b>	<b>0,09</b>	<b>0,09</b>	<b>0,12</b>	<b>0,13</b>	<b>1,26</b>	<b>0,91</b>	<b>0,68</b>	<b>1,03</b>	<b>2,15</b>	<b>0,58</b>
Dist.	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN	LN
p-value	0,308	0,0786	0,1577	0,0042	0,075	0,753	0,753	0,1577	0,1577	0,004	0,0017	0,002	0,009	0,004	0,006

Solução:

Observando os procedimentos do modelo proposto, apresentados na Figura 14, deve-se determinar um ranking de seis sequências, de forma a buscar a minimização da função objetivo apresentada na equação (11) para o método *GCA* e na equação (13) para o *SA*, e adotar as melhores sequências que se enquadram no perfil de eficiência predefinido. Para solução são apresentados os passos a seguir:

1º passo: Determinar o consumo total de cada componente “ $a_j$ ”:

Para a determinação do consumo total de cada componente é utilizada a equação (8), e os dados das Tabelas 1 e 2 – Matriz de Demandas e Matriz de Componentes.

$$[N_j] = [2 \ 3 \ 5 \ 10] \cdot \begin{bmatrix} 1 & 2 & 1 & 3 & 0 \\ 2 & 1 & 0 & 0 & 2 \\ 3 & 2 & 1 & 0 & 2 \\ 0 & 0 & 3 & 0 & 3 \end{bmatrix} + [4 \ 12 \ 2 \ 2] \cdot \begin{bmatrix} 2 & 3 & 1 & 2 & 0 \\ 2 & 1 & 2 & 1 & 2 \\ 0 & 2 & 1 & 2 & 0 \\ 3 & 2 & 3 & 2 & 1 \end{bmatrix} + [5 \ 5 \ 4 \ 6] \cdot \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 3 & 2 & 1 & 1 & 3 \\ 2 & 1 & 2 & 1 & 0 \\ 0 & 1 & 2 & 1 & 2 \end{bmatrix}$$

$$[N_j] = [23 \ 17 \ 37 \ 6 \ 46] + [38 \ 32 \ 36 \ 28 \ 26] + [28 \ 20 \ 30 \ 15 \ 32]$$

$$[N_j] = [89 \ 69 \ 103 \ 49 \ 104]$$

Portanto, o consumo total por jornada diária de cada componente é  $a_1 = 89$ ,  $a_2 = 69$ ,  $a_3 = 103$ ,  $a_4 = 49$  e  $a_5 = 104$ . Esses valores são utilizados na determinação da taxa de consumo ideal de cada componente.

2º passo: Elaboração do ranking de seis sequências de produção, utilizando a heurística *GCA* e a meta-heurística *SA*.

Aplicando a heurística *Goal Chasing* Adaptado - *GCA*, conforme apresentado no quarto capítulo, as seis sequências que apresentam os melhores resultados para a função objetivo estão representados na Tabela 4:

Tabela 4 – Sequências obtidas pelo GCA.

Sequências - GCA																				
S\K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
<b>S<sub>1</sub></b>	1	1	3	3	3	2	1	3	3	3	0	2	2	3	2	0	3	3	3	2
<b>FO</b>	3	1	3	0	0	1	1	2	1	0	1	1	1	1	1	1	1	0	2	1
<b>29,636</b>	3	3	0	1	0	3	2	1	2	1	0	3	3	2	3	0	1	2	1	0
<b>S<sub>2</sub></b>	1	1	3	3	3	2	1	3	3	3	0	2	2	3	2	0	3	3	3	2
<b>FO</b>	3	1	3	0	0	1	1	2	1	0	1	1	1	1	1	1	1	0	1	2
<b>30,361</b>	3	3	0	1	0	3	2	1	2	1	0	3	3	2	3	0	1	2	1	0
<b>S<sub>3</sub></b>	1	1	3	3	3	2	1	3	3	3	0	2	2	3	2	0	3	3	3	2
<b>FO</b>	3	1	3	0	0	1	1	2	1	0	1	1	1	1	1	1	1	0	2	1
<b>30,491</b>	3	3	0	1	0	3	2	1	2	1	0	3	3	2	3	0	2	1	1	0
<b>S<sub>4</sub></b>	1	1	3	3	3	2	1	3	3	3	0	2	2	3	2	0	3	3	2	3
<b>FO</b>	3	1	3	0	0	1	1	2	1	0	1	1	1	1	1	1	1	0	2	1
<b>30,573</b>	3	3	0	1	0	3	2	1	2	1	0	3	3	2	3	0	1	2	1	0
<b>S<sub>5</sub></b>	1	1	3	3	3	2	1	3	3	3	0	2	2	3	2	0	3	3	3	2
<b>FO</b>	3	1	3	0	0	1	1	2	1	0	1	1	1	1	1	1	1	0	2	1
<b>30,816</b>	3	3	0	1	0	3	2	1	2	1	0	3	3	2	3	0	1	1	2	0
<b>S<sub>6</sub></b>	1	1	3	3	3	2	1	3	3	3	0	2	2	3	2	0	3	3	3	2
<b>FO</b>	3	1	3	0	0	1	1	2	1	0	1	1	1	1	1	1	1	2	0	1
<b>30,851</b>	3	3	0	1	0	3	2	1	2	1	0	3	3	2	3	0	1	2	1	0

Aplicando a meta-heurística SA, conforme algoritmo apresentado no anexo B, as seis sequências mais eficientes estão representadas na Tabela 5:

Tabela 5 - Sequências obtidas pelo SA

Sequências - SA																				
S\K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
<b>S<sub>1</sub></b>	3	1	3	3	1	3	2	3	0	2	3	1	3	2	2	0	3	3	2	3
<b>FO</b>	2	1	1	0	1	0	1	1	1	1	0	3	1	1	1	1	1	0	3	2
<b>27,862</b>	1	2	2	1	3	2	3	1	0	3	0	3	2	3	3	0	1	0	0	1
<b>S<sub>2</sub></b>	1	1	1	3	2	3	2	0	3	3	2	3	2	3	3	0	3	3	2	3
<b>FO</b>	3	1	3	1	1	0	1	1	1	0	1	2	1	0	1	1	1	0	1	2
<b>28,301</b>	3	3	3	2	3	0	3	0	1	2	3	1	0	2	1	0	1	2	0	1
<b>S<sub>3</sub></b>	1	1	3	1	2	3	2	0	3	3	2	3	2	3	0	3	3	3	2	3
<b>FO</b>	3	1	1	3	1	0	1	1	1	0	1	2	1	1	1	0	1	0	1	2
<b>28,355</b>	3	3	2	3	3	0	3	0	1	2	3	1	0	2	1	0	1	0	2	1
<b>S<sub>4</sub></b>	2	0	2	3	2	3	1	3	3	1	3	2	3	0	3	3	3	2	3	1
<b>FO</b>	1	1	1	1	1	2	3	1	0	1	0	1	1	1	0	0	1	1	2	3
<b>28,385</b>	3	0	3	2	3	1	0	2	1	3	2	3	0	1	0	1	2	0	1	3
<b>S<sub>5</sub></b>	1	1	3	1	2	3	2	3	3	0	2	3	2	3	3	0	3	3	2	3
<b>FO</b>	3	1	1	3	1	0	1	0	1	1	1	2	1	0	1	1	1	0	1	2
<b>28,426</b>	3	3	2	3	3	0	3	2	1	0	3	1	0	2	1	0	1	0	2	1
<b>S<sub>6</sub></b>	1	1	3	1	2	3	3	2	0	3	2	3	3	3	2	0	3	3	2	3
<b>FO</b>	3	1	1	3	1	0	0	1	1	1	1	2	0	1	1	1	1	0	1	2
<b>28,473</b>	3	3	2	3	3	0	1	3	0	2	0	1	1	2	3	0	1	0	2	1

Observa-se que as sequências obtidas pelo SA (Tabela 5) geraram melhores resultados para a FO do que os resultados obtidos pelo GCA (Tabela 4), portanto, para continuidade da solução serão considerados os resultados gerados pelo SA.

3º passo: Simular as linhas de produção utilizando o Promodel, programando as linhas de montagem conforme as melhores sequências do ranking, analisar a eficiência e verificar a adequação ao perfil aceito, conforme o procedimento descrito no quarto capítulo. A Figura 17 apresenta o modelo de simulação executado no Promodel conforme relatório *trace* apresentado no anexo A.

### Condicionamento dos dados de entrada

Para o condicionamento das distribuições de probabilidade dos tempos de execução das tarefas de cada posto de trabalho, apresentados na Tabela 3, os dados coletados foram tratados pelo software **Crystal Ball** conforme Figura 16.

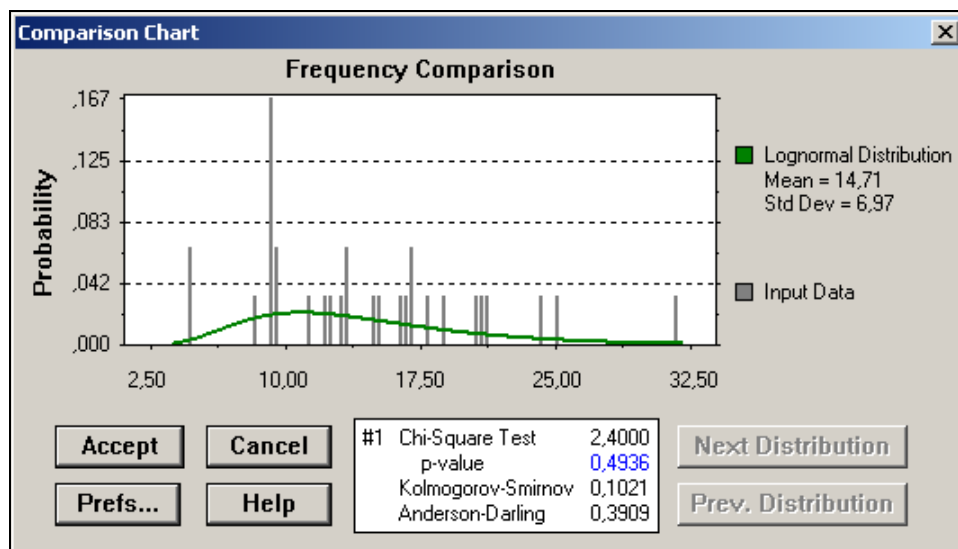


Figura 16 – Condicionamento dos dados de entrada pelo software Crystal Ball

Simulando no Promodel a melhor sequência obtida pelo SA, é possível visualizar o comportamento do modelo no decorrer do tempo, por meio das variáveis de monitoramento **Ideal** e **Real** e, de posse dos dados analíticos, estabelecer a validação do modelo.



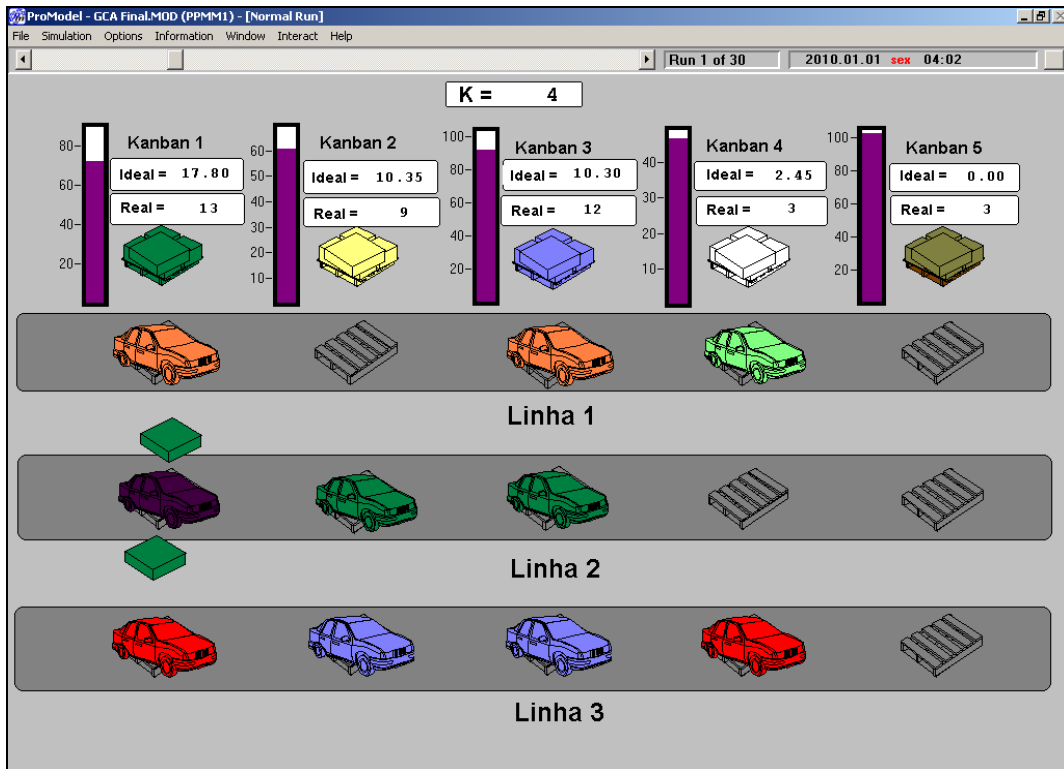


Figura 17 – Modelo de simulação no software Promodel

Com base no relatório do Promodel, é possível verificar a adequação da eficiência do modelo simulado, com relação ao perfil de eficiência aceito. A Figura 18 apresenta o relatório gráfico “*Single Capacity Location State*”, para a simulação da primeira sequência do ranking (SA).



Figura 18 – Relatório Gráfico - Promodel

Os relatórios apresentados nas Figuras 19 e 20 indicam os intervalos de confiança, em porcentagem do tempo total, para os bloqueios e esperas. Tais relatórios são emitidos a cada alteração de sequência simulada, para verificação da restrição de perfil de eficiência.

Statistic	Avg	Std Dev	Low 90% CI	High 90% CI
Linha_1_P1 - % Blocked	14.64	0.24	14.56	14.71
Linha_1_P2 - % Blocked	15.22	0.11	15.18	15.25
Linha_1_P3 - % Blocked	0.00	0.00	0.00	0.00
Linha_1_P4 - % Blocked	3.75	0.26	3.67	3.83
Linha_1_p5 - % Blocked	0.00	0.00	0.00	0.00
Linha_2_P1 - % Blocked	2.01	0.06	1.99	2.03
Linha_2_P2 - % Blocked	0.85	0.10	0.82	0.88
Linha_2_P3 - % Blocked	0.00	0.00	0.00	0.00
Linha_2_P4 - % Blocked	0.00	0.02	0.00	0.01
Linha_2_p5 - % Blocked	0.00	0.00	0.00	0.00
Linha_3_p1 - % Blocked	9.88	0.31	9.78	9.98
Linha_3_p2 - % Blocked	11.52	0.36	11.41	11.63
Linha_3_p3 - % Blocked	0.01	0.02	0.00	0.01
Linha_3_p4 - % Blocked	11.20	0.39	11.08	11.33
Linha_3_p5 - % Blocked	0.00	0.00	0.00	0.00

Figura 19 – Relatório de Bloqueios – Promodel.

Statistic	Avg	Std Dev	Low 90% CI	High 90% CI
Linha_1_P1 - % Waiting	2.05	0.01	2.05	2.06
Linha_1_P2 - % Waiting	2.05	0.01	2.05	2.06
Linha_1_P3 - % Waiting	3.49	0.01	3.49	3.49
Linha_1_P4 - % Waiting	0.41	0.01	0.41	0.41
Linha_1_p5 - % Waiting	3.70	0.01	3.70	3.70
Linha_2_P1 - % Waiting	3.70	0.01	3.70	3.70
Linha_2_P2 - % Waiting	4.11	0.01	4.11	4.11
Linha_2_P3 - % Waiting	4.11	0.01	4.11	4.11
Linha_2_P4 - % Waiting	4.11	0.01	4.11	4.11
Linha_2_p5 - % Waiting	2.88	0.01	2.87	2.88
Linha_3_p1 - % Waiting	2.88	0.01	2.88	2.88
Linha_3_p2 - % Waiting	3.08	0.01	3.08	3.08
Linha_3_p3 - % Waiting	4.11	0.01	4.11	4.11
Linha_3_p4 - % Waiting	3.08	0.01	3.08	3.08
Linha_3_p5 - % Waiting	3.29	0.01	3.28	3.29

Figura 20 – Relatório de Esperas – Promodel.

4º passo: Adotar a melhor sequência que atenda ao perfil de eficiência.

Analisando as sequências obtidas pelo SA, observa-se que nenhuma delas atende ao perfil de eficiência predefinido com relação ao bloqueio máximo de 15%, mas todas atendem ao perfil de espera de 5%. Portanto, conforme o modelo proposto, ilustrado pelo algoritmo apresentado na Figura 14, faz-se necessário o ajuste do perfil de eficiência predefinido. Sendo o bloqueio máximo aceito, reajustado de 15% para 16%, pois conforme a Figura 14, o procedimento propõe “Reavaliar o perfil de eficiência e retornar ao ranking inicial”. Com esse ajuste, a primeira sequência do ranking passa a atender os requisitos de eficiência, com um valor para a função objetivo de 27,862, conforme Tabela 5, sendo, então, a sequência adotada para a programação das linhas de montagem.

Tabela 6- Correlação “F.O. x Bloqueio Crítico”

Sequências SA	F.O.	Bloqueio Crítico	Espera Crítica	Correlação “r” F.O. x Bloqueio Crítico
Sequência 1	<b>27,862</b>	15,25	4,11	<b>0,8923</b>
Sequência 2	28,301	24,65	4,11	
Sequência 3	28,355	24,65	4,11	
Sequência 4	28,385	<b>20,83</b>	4,11	
Sequência 5	28,426	24,65	4,11	
Sequência 6	28,473	25,05	4,11	

As Tabelas 7, 8, 9 e 10 apresentam o consumo total por linha de montagem e geral, para os componentes “a<sub>j</sub>”. Estas tabelas permitem, além da checagem do consumo, a indicação do momento em que o consumo pode ser problemático, e em caso de ajustes ocasionais nas sequências, tais tabelas podem ser tomadas como referência.

Tabela 7 - Diagrama de Consumo Real – linha 1

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	15	17	18	19	20	Total
A <sub>il</sub>	3	1	3	3	1	3	2	3	0	2	3	1	3	2	2	0	3	3	2	3	
a <sub>1</sub>	0	2	0	0	2	0	3	0	1	3	0	2	0	3	3	1	0	0	3	0	<b>23</b>
a <sub>2</sub>	0	1	0	0	1	0	2	0	2	2	0	1	0	2	2	2	0	0	2	0	<b>17</b>
a <sub>3</sub>	3	0	3	3	0	3	1	3	1	1	3	0	3	1	1	1	3	3	1	3	<b>37</b>
a <sub>4</sub>	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	3	0	0	0	0	<b>6</b>
a <sub>5</sub>	3	2	3	3	2	3	2	3	0	2	3	2	3	2	2	0	3	3	2	3	<b>46</b>

Tabela 8 - Diagrama de Consumo Real – linha 2

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	15	17	18	19	20	Total
A <sub>i2</sub>	2	1	1	0	1	0	1	1	1	1	0	3	1	1	1	1	1	0	3	2	
a <sub>1</sub>	0	2	2	2	2	2	2	2	2	2	2	3	2	2	2	2	2	2	3	0	38
a <sub>2</sub>	2	1	1	3	1	3	1	1	1	1	3	2	1	1	1	1	1	3	2	2	32
a <sub>3</sub>	1	2	2	1	2	1	2	2	2	2	1	3	2	2	2	2	2	1	3	1	36
a <sub>4</sub>	2	1	1	2	1	2	1	1	1	1	2	2	1	1	1	1	1	2	2	2	28
a <sub>5</sub>	0	2	2	0	2	0	2	2	2	2	0	1	2	2	2	2	2	0	1	0	26

Tabela 9 - Diagrama de Consumo Real – linha 3

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	15	17	18	19	20	Total
A <sub>i3</sub>	1	2	2	1	3	2	3	1	0	3	0	3	2	3	3	0	1	0	0	1	
a <sub>1</sub>	3	2	2	3	0	2	0	3	1	0	1	0	2	0	0	1	3	1	1	3	28
a <sub>2</sub>	2	1	1	2	1	1	1	2	0	1	0	1	1	1	1	0	2	0	0	2	20
a <sub>3</sub>	1	2	2	1	2	2	2	1	1	2	1	2	2	2	2	1	1	1	1	1	30
a <sub>4</sub>	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	0	1	0	0	1	15
a <sub>5</sub>	3	0	0	3	2	0	2	3	1	2	1	2	0	2	2	1	3	1	1	3	32

Tabela 10 - Consumo Real Total Geral

aj\Tt	Total Linha 1	Total Linha 2	Total Linha 3	Total Geral
a <sub>1</sub>	23	38	28	89
a <sub>2</sub>	17	32	20	69
a <sub>3</sub>	37	36	30	103
a <sub>4</sub>	6	28	15	49
a <sub>5</sub>	46	26	32	104

### 5.3 Análise dos resultados

Inicialmente, convém destacar que o condicionamento dos dados, não apresentou boa aderência para todas as amostras, mas por questão de padronização todas foram aceitas como distribuições lognormais - LN.

Observa-se que o aumento no valor da função objetivo pode estar correlacionado com o aumento da variável **bloqueio crítico**, enquanto o valor da variável **espera crítica**, aparentemente, permanece constante.

Para tal análise foi calculado o coeficiente de correlação de Pearson, para as variáveis: **valor da função objetivo** e **bloqueio crítico**, obtendo um coeficiente de correlação  $r = 0,8923$ , ou seja uma correlação positiva forte (Tabela 6). Mesmo com

uma correlação forte, observa-se que é possível um **aumento** no valor da função objetivo com **redução** do bloqueio crítico, como ocorre na sequência 4.

#### 5.4 Considerações Finais

A aplicação apresentada neste capítulo mostrou a eficiência do modelo proposto para um caso relativamente abrangente. Muito ainda pode ser explorado aplicando o modelo proposto a casos distintos, ainda mais abrangentes e com diferentes enfoques determinísticos ou estocásticos.

A ausência, na literatura, de casos de sequenciamento de linhas múltiplas sujeitas a incertezas, dificulta a avaliação do método proposto, por não haver um parâmetro de comparação, porém a consistência e validade interna podem ser constatadas pela coerência e eficiência das respostas obtidas.

Para análise e reflexão mais apuradas sobre as características e informações extraídas da aplicação em questão, segue o sexto capítulo, incluindo ainda as recomendações e as sugestões para trabalhos futuros.

## CAPÍTULO 6 - CONCLUSÕES E RECOMENDAÇÕES

### 6.1 Considerações Iniciais

Primeiramente, pode-se concluir que os objetivos, tanto gerais quanto específicos, foram plenamente atingidos e, em certos aspectos, até superados, conforme é detalhado a seguir.

Devido à extensão do tema estudado, são inúmeras as conclusões e recomendações para trabalhos futuros, portanto, neste capítulo, são apresentadas as conclusões de maior destaque acadêmico e, também, as consideradas mais impactantes nas aplicações práticas.

Com relação a trabalhos futuros, as recomendações têm como objetivo ampliar o detalhamento e generalização do tema estudado. Portanto, seguem a linha de pesquisa com enfoque híbrido, considerando condições determinísticas e estocásticas quando o caso exigir, explorando técnicas matemáticas e computacionais em constante evolução.

### 6.2 Conclusões sobre o *Goal Chasing* Adaptado – *GCA*

O *Goal Chasing* Adaptado para linhas múltiplas – *GCA* (*Adapted Goal Chasing – AGC*), testado para três linhas de montagem, apresentou resultados ligeiramente inferiores aos apresentados pelo *Simulated Annealing*. Para o caso estudado, o *GCA* apresentou, como melhor resultado, uma função objetivo de “29,636”, enquanto o *Simulated Annealing* apresentou “27,862”, ou seja, uma redução de 5,98%.

Tendo em vista que o *GCA* pode ser aplicado apenas com o uso de uma planilha eletrônica, os resultados obtidos pelo *GCA* podem ser considerados expressivos.

A modelagem do *GCA* em planilha eletrônica apresenta certa complexidade, exigindo habilidade do usuário ou programador, porém, após a modelagem inicial, as sequências de programação podem ser alteradas com certa facilidade, sem necessidade

de alterações na lógica do modelo, como também ocorre com os modelos simulados pelo Promodel.

### 6.3 Conclusões sobre o *Simulated Annealing* – SA

A extensão do **espaço solução** do problema de sequenciamento apresentado, combinado com a complexidade do equacionamento dificulta a aplicação de métodos de busca exatos ou otimizantes. Nesse cenário, o SA se apresentou como uma solução relativamente eficiente, pois mesmo não garantindo a solução ótima apresentou soluções satisfatórias e, conforme citado, superiores em qualidade ao GCA.

A validação do modelo implementado em linguagem “C” para o SA pôde ser obtida pela aplicação do programa em dois casos clássicos da literatura, extraídos de Mondem (1983) e Duplaga et al. (1996), obtendo os mesmos valores, para a função objetivo. Ainda, teve sua validação reforçada por uma aplicação em um problema hipotético de linhas múltiplas solucionado pelo GCA, apresentando uma solução ligeiramente superior, ou seja, um valor menor para a função objetivo.

Portanto, os resultados obtidos indicam que é provável que o SA apresente soluções semelhantes ao GCA em problemas de menores dimensões e soluções melhores em problemas de dimensões maiores.

A qualidade das soluções apresentadas pelo SA depende muito do ajuste dos parâmetros: temperatura inicial, taxa de resfriamento e número máximo de iterações. O ajuste de tais parâmetros, assim como a implementação do SA em linguagem “C”, exige certa experiência por parte do usuário ou do programador. Portanto, neste trabalho foi necessário o suporte de um especialista na implementação do SA.

### 6.4 Conclusões sobre a Simulação a Eventos Discretos

A aplicação da simulação a eventos discretos se justifica, basicamente, por não haver uma formulação matemática completa para o problema e por não haver um

método analítico satisfatório para a solução do modelo matemático, quando consideradas as incertezas.

O software Promodel comprovou o poder da simulação para representar incertezas em situações condicionais complexas, demonstrando por meio de seus relatórios, não só os tempos percentuais de bloqueio e espera críticos, mas também o local onde tais bloqueios e esperas ocorrem, pois são informações importantes para possíveis correções e ajustes no sequenciamento e, também, para prevenção de possíveis paralisações da linha.

A validação do modelo de simulação efetivou-se pela comparação dos resultados do modelo simulado com resultados analíticos e informações lógicas esperadas. No caso, o modelo simulado apresenta, exatamente, as mesmas respostas dos métodos analíticos, dispensando a aplicação de ferramentas estatísticas nos processos de verificação e validação.

Conforme observado, os intervalos de confiança para os tempos percentuais de bloqueio e espera, apresentam pequena amplitude, sendo alguns até pontuais. Tal fato pode ser atribuído às baixas dispersões dos tempos de operação, mesmo quando consideradas as variações em quantidades de componentes.

## 6.5 Conclusões sobre o modelo de sequenciamento proposto

O problema de sequenciamento de LMMM múltiplas em condições de incerteza foi tratado como um problema híbrido, considerando a busca da primeira meta, como um problema de otimização combinatória discreta, e a busca da segunda meta como uma restrição, representada pelo limite superior de um intervalo de confiança estatístico.

No caso apresentado, a restrição relacionada à segunda meta não foi respeitada inicialmente, exigindo o ajuste no perfil de eficiência aceito, conforme indicado no modelo proposto.

Aparentemente, o modelo proposto abrange muitas possíveis situações, por mais adversas que sejam, porém exige certa habilidade e subjetividade por parte do usuário,



no sentido de relacionar os parâmetros do modelo com a realidade encontrada no ambiente de manufatura.

Também convém destacar que as restrições com relação aos limites de bloqueio e espera, não estão obrigatoriamente associadas às paralisações nas linhas de montagem, mas sim às metas de produtividade e melhoria dos processos.

Caso o objetivo seja evitar as paradas de linha, deve se ajustar o modelo para tal situação.

## 6.6 Conclusões gerais

Para maior generalização do modelo proposto é necessária a aplicação em mais casos semelhantes, explorando, inclusive, situações distintas do sequenciamento, como balanceamento, dimensionamento de *kanbans* e *setup's*.

O problema em questão pode ser modelado como um problema de rede com o objetivo de encontrar o caminho mínimo, porém é necessária a inserção de penalidades no cálculo dos caminhos devido à restrição de demanda de cada produto, limitando a região de solução.

Foi observado também que, quanto maior o nível de automação das tarefas, menor a dispersão nos tempos de operação.

A validação do modelo matemático, por meio da aplicação determinística do Promodel também merece destaque, visto que, quando as incertezas são excluídas do modelo, a resposta coincide com a resposta do modelo analítico.

As ferramentas estocásticas como simulação, a princípio, podem ser combinadas com ferramentas determinísticas, desde que a natureza do problema permita.

Em relação ao método proposto por Monden, o método proposto neste trabalho exige uma restrição adicional, que limita a quantidade programada em cada momento “k”. No **problema de Monden**, com uma linha de montagem apenas, esta restrição é automática, porque o consumo ideal é igual ao consumo real no último momento “k”. No método proposto, a média de todas as linhas é igual ao consumo ideal, mas o consumo individual real pode exceder a demanda diária, exigindo tal restrição adicional.

Geralmente, quando as incertezas são inseridas nos modelos, fica exposta a grande fragilidade de certos métodos determinísticos.

Uma conclusão de grande relevância consiste na propriedade da **inversibilidade do sequenciamento**, ou seja, foi constatado empiricamente, em todos os casos analisados no decorrer desta pesquisa, que é possível inverter a sequências sem que a função objetivo se altere. Tal conclusão é reforçada por um recente trabalho apresentado por Flidner et al. (2010), onde é comprovada analiticamente a propriedade da inversibilidade, denominada pelo autor de **simetria do sequenciamento**.

A inversibilidade pode ser de grande utilidade prática, quando determinado produto a ser inserido na linha no início da jornada diária, não se encontra com todos os componentes disponíveis, sendo possível, então, o início da montagem pelos últimos produtos da sequência (inversão), sem prejuízo do nivelamento do consumo.

### 6.7 Recomendações para trabalhos futuros

Para continuidade deste trabalho são apresentadas algumas sugestões e recomendações:

a) Elaboração de uma regra ou fator de ponderação para as metas do sequenciamento, ou seja, a importância de cada meta pode ser diferenciada, cada qual ponderada conforme critério do usuário;

b) Inserção de maiores detalhes de ordem prática que podem ser encontrados em ambiente real como, por exemplo: tempo de abastecimento, movimentação humana e, principalmente, *setups*, gerando novas metas e, conseqüentemente, uma nova função objetivo a ser otimizada, aprimorando o modelo de simulação computacional;

c) A verificação da segunda meta pode ser explorada estatisticamente por meio de um teste de hipóteses, como por exemplo, o teste ‘t’ (não pareado). Considerando-se que o perfil de eficiência aceito seja definido por um tempo improdutivo máximo, dentro de um intervalo hipotético de aceitação, com nível de confiança predeterminado, o que seria uma aplicação, matematicamente mais elegante, da verificação de pertinência a um intervalo de confiança estatístico;

d) Analisar o modelo visando à minimização das paralisações das linhas utilizando os recursos específicos do Promodel, como **Conveyor / Queue**, desde que as condições do problema em estudo sejam compatíveis;

e) Implementar uma meta-heurística de busca local, selecionando como ponto de troca de **bit**, o produto com maior “ $D_k$ ” e não a troca aleatória como foi utilizado neste trabalho. Tal procedimento pode, ocasionalmente, acelerar o processo de busca e melhorar a qualidade da resposta;

f) Buscar possíveis generalizações do modelo, aplicando o mesmo a casos de maior extensão e complexidade. Assim, aprimorar a validade externa do modelo e, conseqüentemente, ampliar as delimitações do trabalho para situações ainda mais gerais.

## 6.8 Considerações Finais

A expectativa, com a finalização deste trabalho, está na possibilidade de aplicação do método proposto, em algumas das diversas empresas que se enquadram no perfil do problema estudado, e também, na divulgação em periódicos técnicos para aprimoramento do método e contribuição com a comunidade científica.

Na certeza de que muitos trabalhos mais aprofundados se sucederão nessa área, espera-se ter apresentado alguma contribuição significativa para as ciências aplicadas à gestão, em constante evolução. Pois, apesar da acelerada evolução dos recursos técnicos e computacionais, a capacidade de processamento e solução de problemas, dificilmente, algum dia, será maior do que a extensão e complexidade dos problemas que ainda estão por surgir.

## REFERÊNCIAS

- AL-AOMAR, R.; Handling multi-lean measures with simulation and simulated annealing. **Journal of the Franklin Institute**. Article in Press. Franklin Inst. DOI:10.1016/j.jfranklin.2010.05.002, 2010.
- BALCI, O.; Verification, validation and accreditation of simulation models. **Proceedings of the Winter Simulation Conference**, Atlanta, USA. 1997.
- BAUTISTA, J.; COMPANYS, R.; COROMINAS, A.; Heuristics and exact algorithms for solving the Monden problem. **European Journal of Operational Research**, v.88, p. 101-113, 1996.
- BANKS, J.; **Handbook of simulation: Principles, Methodology, Advances, Applications, and Practice**. Ed. John Wiley & Sons, Inc., p. 864, 1998.
- BANKS, J. J. S.; CARSON, B. L.; NELSON, D. M.; **Discrete-event simulation**. 4. ed. Upper Saddle River, NJ: Prentice-Hall, 2005.
- BERTRAND, J. W. M.; FRANSOO, J. C.; Modeling and simulation - operations management research methodologies using quantitative modeling. **International Journal of Operations & Production Management**, v.22, n.3, p. 241-264, 2002.
- BLUM, C.; ROLI, A.; Metaheuristics in combinatorial optimization overview and conceptual comparison. **ACM Computing Surveys**. v.35, n.3, p. 268-308, 2003.
- BOYSEN, N.; FLIEDNE, M.; SCHOLL, A.; Sequencing mixed-model assembly lines: Survey, classification and model critique. **European Journal of Operational Research**, v.192, p. 349–373, 2009.
- CELANO, G., COSTA, A., FICHERA, S.; A comparative analysis of sequencing heuristics for solving the Toyota Goal Chasing problem. **Robotics and Computer Integrated Manufacturing**. v.20, p. 573–581, 2004.
- CHAVES, A. A.; **Uma meta-heurística híbrida com busca por agrupamentos aplicada problemas de otimização combinatória**. 2009. 197 f. (Tese de Doutorado). INPE - São José dos Campos. 2009.
- CHWIF, L.; **Redução de modelos de simulação de eventos discretos na sua concepção: uma abordagem causal**. 1999. 151 f. Tese de Doutorado em Engenharia Mecânica - Escola Politécnica, Universidade de São Paulo, São Paulo, 1999.
- CHWIF, L.; MEDINA, A.C.; **Modelagem e Simulação de Eventos Discretos: Teoria e Aplicações**. São Paulo: Ed. dos Autores, p. 254, 2006.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; **Algoritmos - Teoria e Prática**. Editora Campus. Rio de Janeiro. 2ª edição. 2002.

DERRINGER, G., SUICH, R.; Simultaneous optimization of several response variables. **Journal of Quality Technology**. v.12, n.4, p. 214-219, 1980.

DREXL, A., KIMMS, A.; Sequencing JIT mixed-model assembly lines under station-load and part-usage constraints. **Management Science**, v.47, p. 480–491, 2001.

DUPLAGA, E. A.; HAHN, C. K.; HUR, D.; Mixed-model assembly line sequencing at Hyundai Motor Company. **Production and Inventory Management Journal**. v.37, n.3, p. 20, 1996.

FISHER, M. L.; ITTNER, C. D.; The impact of product variety on automobile assembly operations: Empirical evidence and simulation analysis. **Management Science**. v.45, n.6, 1999.

FLIEDNER, M.; BOYSEN, N.; SCHOLL, A.; Solving symmetric mixed-model multi-level just-in-time scheduling problems. **Discrete Applied Mathematics**. v.158, p. 222-231, 2010.

GLOVER, F.; Future paths for integer programming and links to artificial intelligence. **Computers and Operations Research**. v.5, p. 553-549, 1986.

HALACHMI, I., DZIDIC, A.; METZ, J. H. M.; SPEELMAN, L.; DIJKHUIZEN, A. A.; KLEIJNEN, J. P. C.; Validation of a simulation model for robotic milking barn design. **European Journal of Operational Research**. v.134, p. 677–688, 2001.

HANCOCK, L.; KRIEGER, M.; **Manual de Linguagem C**, Rio de Janeiro, Campus, p. 182, 1985.

HARRELL, C.; GHOSH, B. K.; BOWDEN, R.; **Simulation Using Promodel**. 3.ed. Boston: McGraw-Hill. p. 603, 2000.

HELD, M.; KARP, R. M.; A dynamic programming approach to sequencing problems. **Journal of Society Industrial Applied Math**. v.10, n.1, 1962.

HILLIER, F. S.; LIEBERMAN, G. J.; **Introdução à Pesquisa Operacional**. São Paulo. McGraw-Hill, 2006.

HOLWEG, M., The genealogy of lean production. **Journal of Operations Management**. v.25, p. 420-437, 2007.

INMAN, R. R.; BULFIN, R. L.; Sequencing JIT mixed-model assembly lines. **Management Science**. v.37, n.7, 1991.

JAHANGIRIAN, M.; ELDABI, T.; NASEER, A.; STERGIOULAS, L. K.; YOUNG, T.; Simulation in manufacturing and business: A review. **European Journal of Operational Research**. v.203, p. 1–13, 2010.

KELTON, W. D.; Designing simulation experiments. In: **Proceedings of the Winter Simulation Conference**, Phoenix, AZ, USA. 1999.

KIRKPATRICK, S.; GELLAT, C. D.; VECCHI, M. P.; Optimization by Simulated Annealing. **Science**. v.220, n.459813, 1983.

KLEIJNEN, J.P.C.; Theory and Methodology: Verification and validation of simulation models. **European Journal of Operational Research**, v.82, p. 145-162, 1995.

KLEIJNEN, J. P. C.; Validation of models: Statistical techniques and data availability. **Proceeding of the Winter Simulation Conference**. 1999.

KORKMAZEL, T.; MERAL, S.; Bicriteria sequencing methods for the mixed-model assembly line in just-in-time production systems. **European Journal of Operational Research**, v.131, p. 188-207, 2001.

KUBIAK, W.; SETHI, S.; A Note on "Level schedules for mixed-model assembly lines in Just-in-Time production systems". **Management Science**. v.37, n.1, 1991.

KURASHIGE, K.; YOSHINARI, Y.; MIYAZAKI, S.; KAMEYAMA, Y.; Sequencing method for products in consideration of assembly time. **Int. J. Production Economics**. v.60–61, p. 565–573, 1999.

LAW, A.; KELTON, D.; **Simulation modeling and analysis**. New York, McGraw-Hill, 2000.

LEAL, F.; **Análise do efeito interativo de falhas em processos de manufatura através de projeto de experimentos simulados**. 2008. 237 f. (Tese de Doutorado). Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2008.

LEU, Y. Y.; MATHESON, L. A.; REES, L. P.; Sequencing mixed-model assembly lines with Genetic Algorithms. **Computers Industrial Engineering**. v.30, n.4, p. 1027-1036, 1996.

LEU, Y. Y.; HUANG, P. Y.; RUSSEL, R. S.; Using bean search techniques for sequencing mixed-model assembly lines. **Annals of Operational Research**. v.70, p. 379-397, 1997.

LUMUS, R. R.; A simulation analysis of sequencing alternatives for JIT lines using kanbans. **Journal of Operations Management**. v.13, p. 183-191, 1995.

MARCONE, J. F. S.; MARTINS, A. X.; ARAÚJO, C. R.; Experiências com *simulated annealing* e busca tabu na resolução do problema de alocação de salas. **Simpósio Brasileiro de Pesquisa Operacional - SBPO**. Rio de Janeiro/RJ. 2002.

MARCONI, M. A.; LAKATOS, M. E.; **Técnicas de Pesquisa**. 6ª edição. São Paulo: Atlas, 2006.

MATZKA, J.; MASCOLO, M. D.; FURMANS, K.; Buffer sizing of a Heijunka Kanban system. **Journal of Intelligent Manufacturing**. DOI 10.1007/s10845-009-0317-3, 2009.

MCMULLEN, P. R.; JIT Sequencing for mixed-model assembly lines with setups using Tabu Search. **Production Planning & Control**. v.9, n.5, p. 504-510, 1998.

MCMULLEN, P. R.; FRAZIER G. V.; A simulated annealing approach to mixed-model sequencing with multiple objectives on a just in time line. **IIE Transactions**. v.32. p. 679-686, 2000.

MCMULLEN, P. R.; An ant colony optimization approach to addressing a just in time sequencing problem with multiple objectives. **Artificial Intelligence in Engineering**. v.15, p. 309-317, 2001.

METROPOLIS, N.; ROSENBLUTH, A. W.; ROSENBLUTH, M. N.; TELLER, E.; Equation of state calculations by fast computing machines. **The Journal of Chemical Physics**. v.21, p. 1087-1092, 1953.

MILTENBURG, J.; Level schedules for mixed-model assembly lines in Just-in-Time production systems. **Management Science**. v.35, n.2, 1989.

MITROFF, I. I.; BETZ, F.; PONDY, L.R.; SAGASTI, F.; On managing science in the systems Age: Two schemas for the study of science as a whole systems phenomenon. **Interfaces**. v.4, n.3, p. 46-58, 1974.

MONDEN, Y.; **Toyota Production System – Practical approach to production management**. Industrial Engineering and Management Press – Institute of Industrial Engineers. p. 181-192, 1983.

OHNO, T.; **O Sistema Toyota de Produção: Além da produção em larga escala**. Porto Alegre. Bookman, 1997.

OKAMURA, K.; YAMASHINA, H.; A heuristic algorithm for the assembly line model-mix sequencing problem to minimize the risk of stopping the conveyor. **International Journal of Production Research**. v.17, p. 233, 1979.

PROMODEL, User's guide. **Promodel Corporation**, 2002.

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; Dimensionamento de kanban estatístico por DOE simulado. **Simpósio de Excelência em Gestão Tecnológica – SEGeT**. Resende – RJ. 2007.

SARGENT, R.G.; Validation and verification of simulation models. In: **Proceedings of the Winter Simulations Conference**, Washington, DC, USA. 2005.

SCHOLL, A.; KLEIN, R.; DOMSCHKE, W.; Pattern based vocabulary building for effectively sequencing mixed-model assembly lines. **Journal of Heuristics**. v.4, p. 359–381, 1998.

SCHOMBERGER, R. J.; Plant layout becomes product-oriented with cellular, just in-time production concepts. **Industrial Engineering**. v.15, n.11, p. 66-71, 1983.

SCHOMBERGER, R. J.; Kanban at the Nexus - **Production and Inventory Management Journal**. v.43, n.3/4, p. 1, 2002.

SHANNON, P. W.; KRUMWIEDE, K. R.; STREET, J. N.; Using Simulation to Explore Lean Manufacturing Implementation Strategies. **Journal of Management Education**. v.34, n.2, p. 280 –302, 2010.

SHAO, X.; WANG, B.; RAO, Y; GAO, L.; XU, C.; Metaheuristic approaches to sequencing mixed-model fabrication/assembly systems with two objectives. **International Journal of Advanced Manufacturing Technology**. v.48, p. 1159–1171, 2010.

STEINER, G.; YEOMANS, S.; Level scheduling for mixed-model, Just in Time process. *Management Science*. v.39, n.6, 1993.

STRACK, J.; **Modelagem e simulação de sistemas**. Rio de Janeiro. LTC, 1984.

SUMICHRAST, R. T.; OXENRIDER, K. A.; CLAYTON, E. R.; An evolutionary algorithm for sequencing production on a paced assembly line. **Decision Sciences**. v.31, n.1, 2000.

TAMURA, T.; LONG, H.; OHNO. K.; A sequencing problem to level part usage rates and work loads for a mixed-model assembly line with a bypass subline. **International Journal of Production Economics**. v.60–61, p. 557–564, 1999.

THOMOPOULOS, N. T.; Line balancing-sequencing for mixed-model assembly. **Management Science**. v.14, n.2, 1967.

TRIOLA, M. F.; **Introdução à Estatística**. 9ª edição. Rio de Janeiro. Editora LTC, 2005.



TSAI, L. H.; Mixed-model sequencing to minimize utility work and the risk of conveyor stoppage. **Management Science**. v.41, n.3, 1995.

XIAOBO, Z.; OHNO, K.; Algorithms for sequencing mixed models on an assembly line in a JIT production system. **Computers Ind. Engng**. v.32. n.1. p.47-56. 1997.

XIAOBO, Z.; ZHOU, Z.; Algorithms for Toyota's goal of sequencing mixed models on an assembly line with multiple workstations. **Journal of the Operational Research Society**. v.50, p. 704-710, 1999.

XIAOBO, Z.; OHNO, K.; Properties of a sequencing problem for a mixed model assembly line with conveyor stoppages. **European Journal of Operational Research**. v.124, p. 560-570, 2000.

YANO, C. A.; RACHAMADUGU, R.; Sequencing to minimize work overload in assembly lines with product options. **Management Science**. v.37, n.5, may, 1991.

## **BIBLIOGRAFIA CONSULTADA**

**ALMEIDA, D. A.; Critical path analysis type scheduling in a finite capacity environment.** 1992. 186 f. (Ph.D. Thesis). School of industrial and manufacturing science. Cranfield Institute of technology. Cranfield, 2000.

**PINHO, A.; Metodologia para utilização de algoritmos genéticos em modelos de simulação computacional em ambientes de manufatura.** 2008. 189 f. (Tese de Doutorado). Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2008.

## GLOSSÁRIO

**Análise combinatória.** Técnica Matemática utilizada para obtenção do número de possíveis combinações, arranjos ou permutações.

**Balanceamento.** Distribuição de tarefas em uma unidade produtiva com o objetivo de nivelar os tempos de operação.

**Componentes críticos.** Peças ou subconjuntos, fornecidos à linha de montagem, que exigem controle com relação à constância do consumo.

**Corrida.** Unidade amostral resultante de uma experimentação ou simulação.

**Cromossomos pais.** Respostas utilizadas para recombinação e geração de outras respostas aprimoradas.

**Elitismo.** Método que seleciona as melhores respostas para formar uma nova população.

**Espaço de soluções.** Conjunto de todas as possíveis soluções de um problema de pesquisa operacional.

**Família de produtos.** Conjunto de produtos com características de processos semelhantes.

**Função Densidade de Probabilidade.** Mapeamento da distribuição de probabilidades para variáveis aleatórias contínuas.

**Função objetivo.** Equação matemática utilizada como referência para verificação da qualidade de uma resposta.

**Gargalo.** Posto de trabalho que possui demanda maior que a sua capacidade.

**Layout celular.** Arranjo físico de máquinas e equipamentos em uma unidade industrial, com o objetivo de produzir itens com similaridade de processo.

**Lead time.** Tempo contínuo de produção de determinado item, desde a chegada dos componentes na unidade produtiva até a efetiva transferência do item acabado.

**Método determinístico.** Procedimento para solução de um problema no qual as variáveis envolvidas são consideradas não aleatórias, não considerando riscos e incertezas envolvidos.

**Método estocástico.** Procedimento para solução de um problema que envolve variáveis aleatórias, utilizando técnicas estatísticas e probabilidades.

**Método míope.** Procedimento que busca solução parcial ou pontual de um problema, não alcançando soluções gerais ou globais.

**Mínimo/Máximo local.** Ponto ótimo de um intervalo que não abrange todo o espaço de soluções.

**Mix.** Combinação de itens com o objetivo de nivelar determinada variável.

**Otimização combinatória.** Procedimento que busca a combinação de um conjunto de variáveis para maximização ou minimização de determinada resposta.

**Problema de pesquisa.** Questão a ser solucionada em trabalho acadêmico, devidamente definida e delimitada.

**Problema de rede.** Problema que pode ser representado graficamente por um conjunto de nós e arcos.

**Problema intratável.** Problema, de grande extensão e complexidade, que inviabiliza a solução pelos métodos analíticos ou computacionais atuais.

**Programa-mestre.** Composição de pedidos efetivos e previsão de demanda, com o objetivo de orientar a programação da produção. Geralmente mensal.

**Replicações.** Repetições de uma corrida nas mesmas condições experimentais.

**Restrições.** Condições, representadas por expressões matemáticas, que devem ser respeitadas no processo de otimização.

**Solução analítica.** Solução obtida por meio de equações e cálculos matemáticos.

**Solução ótima.** Melhor solução possível pertencente ao espaço de soluções, podendo ser múltipla.

**Troca rápida.** Operação acelerada de mudança de ferramental e regulagem de equipamento visando à redução do tempo ocioso numa unidade produtiva (SMED).

**Validade externa.** Característica de um método que permite que ele seja generalizado e se aplique a casos além do estudado.

**Validade interna.** Coerência de um método com relação às características e os objetivos do problema em estudo.

**Variável contínua.** Variável que pode assumir um número infinito de valores dentro de um intervalo qualquer.

**Variável discreta.** Variável que só pode assumir um número finito de valores dentro de um intervalo qualquer.

**Work in process.** Material localizado nas etapas intermediárias do processo produtivo entre matéria-prima e produto acabado. Estoque intermediário.

## APÊNDICE A – PUBLICAÇÕES GERADAS

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; Dimensionamento de kanban estatístico por DOE simulado. **Simpósio de Excelência em Gestão Tecnológica – SEGeT**. Resende – RJ. 2007.

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; Dimensionamento de Kanban Estatístico por Simulação de Monte Carlo Utilizando o Software Crystal Ball. **Simpósio de Excelência em Gestão Tecnológica – SEGeT**. Resende – RJ. 2007.

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; A compatibilidade entre técnicas de otimização e simulação de eventos discretos. **Encontro Nacional de Engenharia de Produção – ENEGEP**. Rio de Janeiro – RJ. 2008.

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; Aplicação integrada de projeto de experimentos, simulação de eventos discretos e otimização na manufatura. **Encontro Nacional de Engenharia de Produção – ENEGEP**. Rio de Janeiro – RJ. 2008.

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; Proposta para modelagem do sequenciamento de linhas de montagem múltiplas em ambiente de produção enxuta utilizando simulação. **Simpósio Brasileiro de Pesquisa Operacional – SBPO**. Bento Gonçalves – RS. 2010.

SANCHES, A. L.; ISHIBASHI, L. E.; ALMEIDA, C. F.; CHAVES, A. A.; MARINS, F. A. S.; Sequenciamento de linhas de montagem múltiplas em ambiente de produção enxuta utilizando Simulated Annealing. **Encontro Regional de Pesquisa Operacional – ERPO**. Rio de Janeiro – RJ. 2010.

SANCHES, A. L.; MARINS, F. A. S.; MONTEVECHI, J. A. B.; Sequencing mixed-model in multiple assembly lines with common suppliers: a simulated study to control the risk of stoppages. Submetido à **Interfaces**. 2010.

## ANEXO A – Relatório Trace – Modelo de Simulação - Promodel

```
*****
*
*                               Formatted Listing of Model :
*       C:\Documents and Settings\Alexandre\Desktop\Tese 2010\GCA Final.MOD
*
*****
```

```
Time Units:           Hours
Distance Units:      Meters
```

```
*****
*                               Locations
*
*****
```

Name	Cap	Units	Stats	Rules	Cost
Kanban_1	89	1	Time Series	Oldest,	,
Kanban_2	69	1	Time Series	Oldest,	,
Kanban_3	103	1	Time Series	Oldest,	,
Kanban_4	49	1	Time Series	Oldest,	,
Kanban_5	104	1	Time Series	Oldest,	,
Linha_1_P1	1	1	Time Series	Oldest,	,
Linha_1_P2	1	1	Time Series	Oldest,	,
Linha_1_P3	1	1	Time Series	Oldest,	,
Linha_1_P4	1	1	Time Series	Oldest,	,
Linha_1_p5	1	1	Time Series	Oldest,	,
Linha_2_P1	1	1	Time Series	Oldest,	,
Linha_2_P2	1	1	Time Series	Oldest,	,
Linha_2_P3	1	1	Time Series	Oldest,	,
Linha_2_P4	1	1	Time Series	Oldest,	,
Linha_2_p5	1	1	Time Series	Oldest,	,
Linha_3_p1	1	1	Time Series	Oldest,	,
Linha_3_p2	1	1	Time Series	Oldest,	,
Linha_3_p3	1	1	Time Series	Oldest,	,
Linha_3_p4	1	1	Time Series	Oldest,	,
Linha_3_p5	1	1	Time Series	Oldest,	,

```
*****
*                               Entities
*
*****
```

Name	Speed (mpm)	Stats	Cost
a1	50	Time Series	
a2	50	Time Series	
a3	50	Time Series	
a4	50	Time Series	
a5	50	Time Series	
AOL1	50	Time Series	
AOL2	50	Time Series	
AOL3	50	Time Series	
A1L1	50	Time Series	
A1L2	50	Time Series	
A1L3	50	Time Series	
A2L1	50	Time Series	
A2L2	50	Time Series	
A2L3	50	Time Series	
A3L1	50	Time Series	
A3L2	50	Time Series	
A3L3	50	Time Series	

```
*****
*                               Processing
*
*****
```

Entity	Location	Process		Routing			
		Operation	Blk	Output	Destination	Rule	Move Logic
a1	Kanban_1		1	a1	Linha_1_P1	JOIN 1	MOVE FOR .05 HR
				a1	Linha_2_P1	JOIN	MOVE FOR .05 HR
a2	Kanban_2		1	a1	Linha_3_p1	JOIN	MOVE FOR .05 HR
				a2	Linha_1_P2	JOIN 1	MOVE FOR .05 HR
a3	Kanban_3		1	a2	Linha_2_P2	JOIN	MOVE FOR .05 HR
				a2	Linha_3_p2	JOIN	MOVE FOR .05 HR
				a3	Linha_1_P3	JOIN 1	MOVE FOR .05 HR
				a3	Linha_2_P3	JOIN	MOVE FOR .05 HR

a4	Kanban_4		1	a3	Li nha_3_p3	JOIN	MOVE FOR	.05 HR
				a4	Li nha_1_p4	JOIN 1	MOVE FOR	.05 HR
				a4	Li nha_2_p4	JOIN	MOVE FOR	.05 HR
a5	Kanban_5		1	a4	Li nha_3_p4	JOIN	MOVE FOR	.05 HR
				a5	Li nha_1_p5	JOIN 1	MOVE FOR	.05 HR
				a5	Li nha_2_p5	JOIN	MOVE FOR	.05 HR
				a5	Li nha_3_p5	JOIN	MOVE FOR	.05 HR
AOL1	Li nha_1_P1	JOIN 1 a1 INC R1 WAIT L(46.32, 0.69)	MIN					
			1	AOL1	Li nha_1_P2	FIRST 1	MOVE FOR	.05 HR
AOL1	Li nha_1_P2	JOIN 2 a2 INC R2 INC R2 WAIT L(41.86, 0.08)	MIN					
			1	AOL1	Li nha_1_P3	FIRST 1	MOVE FOR	.05 HR
AOL1	Li nha_1_P3	JOIN 1 a3 INC R3 WAIT L(47.04, 0.09)	MIN					
			1	AOL1	Li nha_1_P4	FIRST 1	MOVE FOR	.05 HR
AOL1	Li nha_1_P4	JOIN 3 a4 INC R4 INC R4 INC R4 WAIT L(46.03, 0.68)	MIN					
			1	AOL1	Li nha_1_p5	FIRST 1	MOVE FOR	.05 HR
AOL1	Li nha_1_p5	JOIN 0 a5 WAIT 0.2 HR						
			1	AOL1	EXIT	FIRST 1	MOVE FOR	.05 HR
AOL2	Li nha_2_P1	JOIN 2 a1 INC R1 INC R1 WAIT L(48.22, 0.50)	MIN					
			1	AOL2	Li nha_2_P2	FIRST 1	MOVE FOR	.05 HR
AOL2	Li nha_2_P2	JOIN 3 a2 INC R2 INC R2 INC R2 WAIT L(43.28, 0.09)	MIN					
			1	AOL2	Li nha_2_P3	FIRST 1	MOVE FOR	.05 HR
AOL2	Li nha_2_P3	JOIN 1 a3 INC R3 WAIT L(47.04, 0.09)	MIN					
			1	AOL2	Li nha_2_P4	FIRST 1	MOVE FOR	.05 HR
AOL2	Li nha_2_P4	JOIN 2 a4 INC R4 INC R4 WAIT L(45.15, 0.91)	MIN					
			1	AOL2	Li nha_2_p5	FIRST 1	MOVE FOR	.05 HR
AOL2	Li nha_2_p5	JOIN 0 a5 WAIT 0.2 HR						
			1	AOL2	EXIT	FIRST 1	MOVE FOR	.05 HR
AOL3	Li nha_3_p1	JOIN 1 a1 INC R1 WAIT L(46.32, 0.69)	MIN					
			1	AOL3	Li nha_3_p2	FIRST 1	MOVE FOR	.05 HR
AOL3	Li nha_3_p2	JOIN 0 a2 WAIT 0.2 HR						
			1	AOL3	Li nha_3_p3	FIRST 1	MOVE FOR	.05 HR
AOL3	Li nha_3_p3	JOIN 1 a3 INC R3 WAIT L(47.04, 0.09)	MIN					
			1	AOL3	Li nha_3_p4	FIRST 1	MOVE FOR	.05 HR
AOL3	Li nha_3_p4	JOIN 0 a4 WAIT 0.2 HR						
			1	AOL3	Li nha_3_p5	FIRST 1	MOVE FOR	.05 HR
AOL3	Li nha_3_p5	JOIN 1 a5 INC R5						



			WAIT L(42. 18, 1. 03) MIN					
A1L1	Li nha_1_P1	JOIN 2 a1 INC R1 INC R1 WAIT L(48. 22, 0. 50) MIN	1	A0L3	EXIT	FIRST 1	MOVE FOR .05 HR	
A1L1	Li nha_1_P2	JOIN 1 a2 INC R2 WAIT L(41. 87, 1. 50) MIN	1	A1L1	Li nha_1_P2	FIRST 1	MOVE FOR .05 HR	
A1L1	Li nha_1_P3	JOIN 0 a3 WAIT 0. 2 HR	1	A1L1	Li nha_1_P3	FIRST 1	MOVE FOR .05 HR	
A1L1	Li nha_1_P4	JOIN 0 a4 WAIT 0. 2 HR	1	A1L1	Li nha_1_P4	FIRST 1	MOVE FOR .05 HR	
A1L1	Li nha_1_p5	JOIN 2 a5 INC R5 INC R5 WAIT L(45. 18, 2. 15) MIN	1	A1L1	Li nha_1_p5	FIRST 1	MOVE FOR .05 HR	
A1L2	Li nha_2_P1	JOIN 2 a1 INC R1 INC R1 WAIT L(48. 22, 0. 50) MIN	1	A1L2	EXIT	FIRST 1	MOVE FOR .05 HR	
A1L2	Li nha_2_P2	JOIN 1 a2 INC R2 WAIT L(41. 87, 1. 50) MIN	1	A1L2	Li nha_2_P2	FIRST 1	MOVE FOR .05 HR	
A1L2	Li nha_2_P3	JOIN 2 a3 INC R3 INC R3 WAIT L(48. 02, 0. 12) MIN	1	A1L2	Li nha_2_P3	FIRST 1	MOVE FOR .05 HR	
A1L2	Li nha_2_P4	JOIN 1 a4 INC R4 WAIT L(43. 29, 1. 26) MIN	1	A1L2	Li nha_2_P4	FIRST 1	MOVE FOR .05 HR	
A1L2	Li nha_2_p5	JOIN 2 a5 INC R5 INC R5 WAIT L(45. 18, 2. 15) MIN	1	A1L2	Li nha_2_p5	FIRST 1	MOVE FOR .05 HR	
A1L3	Li nha_3_p1	JOIN 3 a1 INC R1 INC R1 INC R1 WAIT L(50. 39, 0. 82) MIN	1	A1L2	EXIT	FIRST 1	MOVE FOR .05 HR	
A1L3	Li nha_3_p2	JOIN 2 a2 INC R2 INC R2 WAIT L(41. 86, 0. 08) MIN	1	A1L3	Li nha_3_p2	FIRST 1	MOVE FOR .05 HR	
A1L3	Li nha_3_p3	JOIN 1 a3 INC R3 WAIT L(50. 39, 0. 82) MIN	1	A1L3	Li nha_3_p3	FIRST 1	MOVE FOR .05 HR	
A1L3	Li nha_3_p4	JOIN 1 a4 INC R4 WAIT L(43. 29, 1. 26) MIN	1	A1L3	Li nha_3_p4	FIRST 1	MOVE FOR .05 HR	
			1	A1L3	Li nha_3_p5	FIRST 1	MOVE FOR .05 HR	

A1L3	Li nha_3_p5	JOIN 3 a5 INC R5 INC R5 INC R5 WAIT L(47.04, 0.58) MIN	1	A1L3	EXIT	FIRST 1	MOVE FOR .05 HR
A2L1	Li nha_1_P1	JOIN 3 a1 INC R1 INC R1 INC R1 WAIT L(50.39, 0.82) MIN	1	A2L1	Li nha_1_P2	FIRST 1	MOVE FOR .05 HR
A2L1	Li nha_1_P2	JOIN 2 a2 INC R2 INC R2 WAIT L(41.86, 0.08) MIN	1	A2L1	Li nha_1_P3	FIRST 1	MOVE FOR .05 HR
A2L1	Li nha_1_P3	JOIN 1 a3 INC R3 WAIT L(47.04, 0.09) MIN	1	A2L1	Li nha_1_P4	FIRST 1	MOVE FOR .05 HR
A2L1	Li nha_1_P4	JOIN 0 a4 WAIT 0.2 HR	1	A2L1	Li nha_1_p5	FIRST 1	MOVE FOR .05 HR
A2L1	Li nha_1_p5	JOIN 2 a5 INC R5 INC R5 WAIT L(45.18, 2.15) MIN	1	A2L1	EXIT	FIRST 1	MOVE FOR .05 HR
A2L2	Li nha_2_P1	JOIN 0 a1 WAIT 0.2 HR	1	A2L2	Li nha_2_P2	FIRST 1	MOVE FOR .05 HR
A2L2	Li nha_2_P2	JOIN 2 a2 INC R2 INC R2 WAIT L(41.86, 0.08) MIN	1	A2L2	Li nha_2_P3	FIRST 1	MOVE FOR .05 HR
A2L2	Li nha_2_P3	JOIN 1 a3 INC R3 WAIT L(47.04, 0.09) MIN	1	A2L2	Li nha_2_P4	FIRST 1	MOVE FOR .05 HR
A2L2	Li nha_2_P4	JOIN 2 a4 INC R4 INC R4 WAIT L(45.15, 0.91) MIN	1	A2L2	Li nha_2_p5	FIRST 1	MOVE FOR .05 HR
A2L2	Li nha_2_p5	JOIN 0 a5 WAIT 0.2 HR	1	A2L2	EXIT	FIRST 1	MOVE FOR .05 HR
A2L3	Li nha_3_p1	JOIN 2 a1 INC R1 INC R1 WAIT L(48.22, 0.50) MIN	1	A2L3	Li nha_3_p2	FIRST 1	MOVE FOR .05 HR
A2L3	Li nha_3_p2	JOIN 1 a2 INC R2 WAIT L(41.87, 1.50) MIN	1	A2L3	Li nha_3_p3	FIRST 1	MOVE FOR .05 HR
A2L3	Li nha_3_p3	JOIN 2 a3 INC R3 INC R3 WAIT L(48.02, 0.12) MIN	1	A2L3	Li nha_3_p4	FIRST 1	MOVE FOR .05 HR
A2L3	Li nha_3_p4	JOIN 1 a4 INC R4 WAIT L(43.29, 1.26) MIN	1				

A2L3	Li nha_3_p5	JOIN 0 a5 WAIT 0.2 HR	1	A2L3	Li nha_3_p5	FIRST 1	MOVE FOR .05 HR
A3L1	Li nha_1_P1	JOIN 0 a1 WAIT 0.2 HR	1	A2L3	EXIT	FIRST 1	MOVE FOR .05 HR
A3L1	Li nha_1_P2	JOIN 0 a2 WAIT 0.2 HR	1	A3L1	Li nha_1_P2	FIRST 1	MOVE FOR .05 HR
A3L1	Li nha_1_P3	JOIN 3 a3 INC R3 INC R3 INC R3 WAIT L(52.82, 0.13) MIN	1	A3L1	Li nha_1_P3	FIRST 1	MOVE FOR .05 HR
A3L1	Li nha_1_P4	JOIN 0 a4 WAIT 0.2 HR	1	A3L1	Li nha_1_P4	FIRST 1	MOVE FOR .05 HR
A3L1	Li nha_1_p5	JOIN 3 a5 INC R5 INC R5 INC R5 WAIT L(47.04, 0.58) MIN	1	A3L1	Li nha_1_p5	FIRST 1	MOVE FOR .05 HR
A3L2	Li nha_2_P1	JOIN 3 a1 INC R1 INC R1 INC R1 WAIT L(50.39, 0.82) MIN	1	A3L1	EXIT	FIRST 1	MOVE FOR .05 HR
A3L2	Li nha_2_P2	JOIN 2 a2 INC R2 INC R2 WAIT L(41.86, 0.08) MIN	1	A3L2	Li nha_2_P2	FIRST 1	MOVE FOR .05 HR
A3L2	Li nha_2_P3	JOIN 3 a3 INC R3 INC R3 INC R3 WAIT L(52.82, 0.13) MIN	1	A3L2	Li nha_2_P3	FIRST 1	MOVE FOR .05 HR
A3L2	Li nha_2_P4	JOIN 2 a4 INC R4 INC R4 WAIT L(45.15, 0.91) MIN	1	A3L2	Li nha_2_P4	FIRST 1	MOVE FOR .05 HR
A3L2	Li nha_2_p5	JOIN 1 a5 INC R5 WAIT L(42.18, 1.03) MIN	1	A3L2	Li nha_2_p5	FIRST 1	MOVE FOR .05 HR
A3L3	Li nha_3_p1	JOIN 0 a1 WAIT 0.2 HR	1	A3L2	EXIT	FIRST 1	MOVE FOR .05 HR
A3L3	Li nha_3_p2	JOIN 1 a2 INC R2 WAIT L(41.87, 1.50) MIN	1	A3L3	Li nha_3_p2	FIRST 1	MOVE FOR .05 HR
A3L3	Li nha_3_p3	JOIN 2 a3 INC R3 INC R3 WAIT L(48.02, 0.12) MIN	1	A3L3	Li nha_3_p3	FIRST 1	MOVE FOR .05 HR
A3L3	Li nha_3_p4	JOIN 1 a4 INC R4 WAIT L(43.29, 1.26) MIN	1	A3L3	Li nha_3_p4	FIRST 1	MOVE FOR .05 HR
A3L3	Li nha_3_p5	JOIN 2 a5 INC R5 INC R5 WAIT L(45.18, 2.15) MIN	1	A3L3	Li nha_3_p5	FIRST 1	MOVE FOR .05 HR
			1	A3L3	EXIT	FIRST 1	MOVE FOR .05 HR

```

*****
*
*                               Arrivals
*
*****

```

Entity	Location	Qty each	First Time	Occurrences	Frequency	Logi c
a1	Kanban_1	89	Fri, Jan 01 2010 @ 12:00 AM	1		
a2	Kanban_2	69	Fri, Jan 01 2010 @ 12:00 AM	1		
a3	Kanban_3	103	Fri, Jan 01 2010 @ 12:00 AM	1		
a4	Kanban_4	49	Fri, Jan 01 2010 @ 12:00 AM	1		
a5	Kanban_5	104	Fri, Jan 01 2010 @ 12:00 AM	1		
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 01:00 AM	1		INC K I deal _1 =
K * 89/20						
(K-1) * 69/20						
(0) * 103/20						
(0) * 49/20						
(0) * 104/20						
A2L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 01:00 AM	1		
A1L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 01:00 AM	1		
A1L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 02:00 AM	1		INC K I deal _1 =
K * 89/20						
(K-1) * 69/20						
(K-2) * 103/20						
(0) * 49/20						
(0) * 104/20						
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 02:00 AM	1		
A2L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 02:00 AM	1		
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 03:00 AM	1		INC K I deal _1 =
K * 89/20						
(K-1) * 69/20						
(K-2) * 103/20						
(K-3) * 49/20						
(0) * 104/20						
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 03:00 AM	1		
A2L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 03:00 AM	1		
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 04:00 AM	1		INC K I deal _1 =
K * 89/20						
(K-1) * 69/20						
(K-2) * 103/20						
(K-3) * 49/20						
(K-4) * 104/20						
AOL2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 04:00 AM	1		
A1L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 04:00 AM	1		
A1L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 05:00 AM	1		INC K I deal _1 =
K * 89/20						
(K-1) * 69/20						
(K-2) * 103/20						
(K-3) * 49/20						
(K-4) * 104/20						
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 05:00 AM	1		
A3L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 05:00 AM	1		
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 06:00 AM	1		INC K I deal _1 =
K * 89/20						
(K-1) * 69/20						
(K-2) * 103/20						

(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
AOL2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 06:00 AM	1	
A2L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 06:00 AM	1	
A2L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 07:00 AM	1	INC K
K * 89/20					Ideal_1 =
(K-1) * 69/20					Ideal_2 =
(K-2) * 103/20					Ideal_3 =
(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 07:00 AM	1	
A3L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 07:00 AM	1	
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 08:00 AM	1	INC K
K * 89/20					Ideal_1 =
(K-1) * 69/20					Ideal_2 =
(K-2) * 103/20					Ideal_3 =
(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 08:00 AM	1	
A1L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 08:00 AM	1	
AOL1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 09:00 AM	1	INC K
K * 89/20					Ideal_1 =
(K-1) * 69/20					Ideal_2 =
(K-2) * 103/20					Ideal_3 =
(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 09:00 AM	1	
AOL3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 09:00 AM	1	
A2L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 10:00 AM	1	INC K
K * 89/20					Ideal_1 =
(K-1) * 69/20					Ideal_2 =
(K-2) * 103/20					Ideal_3 =
(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
A1L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 10:00 AM	1	
A3L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 10:00 AM	1	
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 11:00 AM	1	INC K
K * 89/20					Ideal_1 =
(K-1) * 69/20					Ideal_2 =
(K-2) * 103/20					Ideal_3 =
(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
AOL2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 11:00 AM	1	
AOL3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 11:00 AM	1	
A1L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 12:00 PM	1	INC K
K * 89/20					Ideal_1 =
(K-1) * 69/20					Ideal_2 =
(K-2) * 103/20					Ideal_3 =
(K-3) * 49/20					Ideal_4 =
(K-4) * 104/20					Ideal_5 =
A3L2	Li nha_2_P1	1	Fri, Jan 01 2010 @ 12:00 PM	1	
A3L3	Li nha_3_p1	1	Fri, Jan 01 2010 @ 12:00 PM	1	
A3L1	Li nha_1_P1	1	Fri, Jan 01 2010 @ 01:00 PM	1	INC K

K * 89/20					I deal _1 =
(K-1) * 69/20					I deal _2 =
(K-2) * 103/20					I deal _3 =
(K-3) * 49/20					I deal _4 =
(K-4) * 104/20					I deal _5 =
A1L2	Li nha_2_P1	1	Fri , Jan 01 2010 @ 01:00 PM	1	
A2L3	Li nha_3_p1	1	Fri , Jan 01 2010 @ 01:00 PM	1	
A2L1	Li nha_1_P1	1	Fri , Jan 01 2010 @ 02:00 PM	1	INC K
K * 89/20					I deal _1 =
(K-1) * 69/20					I deal _2 =
(K-2) * 103/20					I deal _3 =
(K-3) * 49/20					I deal _4 =
(K-4) * 104/20					I deal _5 =
A1L2	Li nha_2_P1	1	Fri , Jan 01 2010 @ 02:00 PM	1	
A3L3	Li nha_3_p1	1	Fri , Jan 01 2010 @ 02:00 PM	1	
A2L1	Li nha_1_P1	1	Fri , Jan 01 2010 @ 03:00 PM	1	INC K
K * 89/20					I deal _1 =
(K-1) * 69/20					I deal _2 =
(K-2) * 103/20					I deal _3 =
(K-3) * 49/20					I deal _4 =
(K-4) * 104/20					I deal _5 =
A1L2	Li nha_2_P1	1	Fri , Jan 01 2010 @ 03:00 PM	1	
A3L3	Li nha_3_p1	1	Fri , Jan 01 2010 @ 03:00 PM	1	
A0L1	Li nha_1_P1	1	Fri , Jan 01 2010 @ 04:00 PM	1	INC K
K * 89/20					I deal _1 =
(K-1) * 69/20					I deal _2 =
(K-2) * 103/20					I deal _3 =
(K-3) * 49/20					I deal _4 =
(K-4) * 104/20					I deal _5 =
A1L2	Li nha_2_P1	1	Fri , Jan 01 2010 @ 04:00 PM	1	
A0L3	Li nha_3_p1	1	Fri , Jan 01 2010 @ 04:00 PM	1	
A3L1	Li nha_1_P1	1	Fri , Jan 01 2010 @ 05:00 PM	1	INC K
K * 89/20					I deal _1 =
(K) * 69/20					I deal _2 =
(K) * 103/20					I deal _3 =
(K) * 49/20					I deal _4 =
(K) * 104/20					I deal _5 =
A1L2	Li nha_2_P1	1	Fri , Jan 01 2010 @ 05:00 PM	1	
A1L3	Li nha_3_p1	1	Fri , Jan 01 2010 @ 05:00 PM	1	
A3L1	Li nha_1_P1	1	Fri , Jan 01 2010 @ 06:00 PM	1	INC K
K * 89/20					I deal _1 =
(K) * 69/20					I deal _2 =
(K) * 103/20					I deal _3 =
(K) * 49/20					I deal _4 =
(K) * 104/20					I deal _5 =
A0L2	Li nha_2_P1	1	Fri , Jan 01 2010 @ 06:00 PM	1	
A0L3	Li nha_3_p1	1	Fri , Jan 01 2010 @ 06:00 PM	1	
A2L1	Li nha_1_P1	1	Fri , Jan 01 2010 @ 07:00 PM	1	INC K
K * 89/20					I deal _1 =
(K) * 69/20					I deal _2 =
(K) * 103/20					I deal _3 =
(K) * 49/20					I deal _4 =

```

(K) * 104/20
A3L2   Li nha_2_P1 1          Fri, Jan 01 2010 @ 07:00 PM 1
AOL3   Li nha_3_p1 1          Fri, Jan 01 2010 @ 07:00 PM 1
A3L1   Li nha_1_P1 1          Fri, Jan 01 2010 @ 08:00 PM 1
K * 89/20
(K) * 69/20
(K) * 103/20
(K) * 49/20
(K) * 104/20

A2L2   Li nha_2_P1 1          Fri, Jan 01 2010 @ 08:00 PM 1
A1L3   Li nha_3_p1 1          Fri, Jan 01 2010 @ 08:00 PM 1
    
```

Ideal\_5 =  
INC K  
Ideal\_1 =  
Ideal\_2 =  
Ideal\_3 =  
Ideal\_4 =  
Ideal\_5 =

\*\*\*\*\*  
\* Attributes  
\*\*\*\*\*

ID	Type	Classification
Att1	Integer	Entity

\*\*\*\*\*  
\* Variables (global)  
\*\*\*\*\*

ID	Type	Initial value	Stats
K	Integer	0	Time Series
R1	Integer	0	Time Series
R2	Integer	0	Time Series
R3	Integer	0	Time Series
R4	Integer	0	Time Series
R5	Integer	0	Time Series
Ideal_1	Real	0	Time Series
Ideal_2	Real	0	Time Series
Ideal_3	Real	0	Time Series
Ideal_4	Real	0	Time Series
Ideal_5	Real	0	Time Series

## ANEXO B – Código fonte em linguagem C - sequenciamento pelo SA

```

/*****
** PROGRAMA EM C PARA RESOLVER O PROBLEMA DE SEQUENCIAMENTO **
** DE LINHAS DE MONTAGEM MÚLTIPAS EM AMBIENTE DE PRODUÇÃO **
** ENXUTA POR MEIO DA META-HEURÍSTICAS SIMULATED ANNEALING **
** AUTORES: ANTONIO AUGUSTO CHAVES (FEG/UNESP) **
** : ALEXANDRE LEME SANCHES (FEG/UNESP - UNI FEI) **
** : CRISTIANO FERREIRA DE ALMEIDA (FEG/UNESP) **
** : LAÍS EMY ISHIBASHI (FEG/UNESP) **
*****/

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>
//#include <vcl.h>

#pragma hdrstop

#define MAX(x,y) ((x)<(y) ? (y) : (x))

//Constantes
#define m 3 //numero de linhas de montagem
#define p 4 //numero de produtos
#define c 5 //numero de componentes
#define Q 20 //quantidade de demanda diaria de todos os produtos

#define k1 400 //numero de solucoes geradas

//Tipo Solução
struct TSol
{
    int seq[m][Q];
    double fo;
};

TSol OrdBest[k1];
TSol AUX;

//Variáveis Globais
int A[m][p], //matriz com a quantidade de produtos a serem programados nas linhas
    B[m*p][c], //matriz com a quantidade necessária de componentes para produzir
uma unidade de um produto em cada linha
    N[c], num1; //vetor com a quantidade total necessaria
dos componentes

clock_t CPUinicio, CPUmelhor, CPUfim;

float tempoMelhor = 0.0, tempoTotal = 0.0;

//Declaração de Funções
void LerArquivos();
double CalcularFO(TSol s);
TSol GerarSolucaoInicial();
void ImprimirSolucao(TSol s);

void SA();

double randomico(double min, double max);
int irandomico(int min, int max);

int n=0;

#pragma argsused
/*****
*** Método: Main() ***
*** Função: Programa principal ***
*****/
int main()
{
    TSol s;
    double media=0;

```





```

for (int i=0; i<m; i++)
    for (int j=0; j<p; j++)
        fscanf(arquivo, "%d", &A[i][j]);

//Ler a quantidade necessaria de cada componente
for (int i=0; i<m*p; i++)
    for (int j=0; j<c; j++)
        fscanf(arquivo, "%d", &B[i][j]);

//Ler a quantidade total necessaria de cada componente
for (int i=0; i<c; i++)
    fscanf(arquivo, "%d", &N[i]);

fclose(arquivo);

//imprimir as tabelas - teste
printf("A[m][p]\n");
for (int i=0; i<m; i++)
{
    for (int j=0; j<p; j++)
        printf("%d ", A[i][j]);

    printf("\n");
}

printf("\nB[mp][c]\n");
for (int i=0; i<m*p; i++)
{
    for (int j=0; j<c; j++)
        printf("%d ", B[i][j]);

    printf("\n");
}

printf("\nN[c]\n");
for (int i=0; i<c; i++)
    printf("%d ", N[i]);
}

/*****
*** Método: CalcularF0(TSol s) ***
*** Função: Calcula o valor da função objetivo da solução s ***
*****/
double CalcularF0(TSol s)
{
    double DTotal = 0.0, Dk, d;
    int X[c], b, K, aux1;

    //inicializar X
    for (int i=0; i<c; i++)
        X[i] = 0;

    //calcular cada valor de Dk
    for (int i=0; i<Q; i++)
    {
        K = i+1;
        //calcular a diferenca entre o ideal e o real
        Dk = 0;
        for (int j=0; j<c; j++)
        {
            aux1 = 0;
            for(int k=0; k<m; k++)
            {
                aux1 += B[s.seq[k][i] + (k*p)][j];
            }
            double aux = (double)K*N[j];
            d = (aux/Q) - X[j] - aux1;
            Dk = Dk + d*d;

            //ajustar o consumo X[j]
            X[j] = X[j] + aux1;
        }

        Dk = sqrt(Dk);
        DTotal += Dk;
    }

    return DTotal;
}

/*****
*** Método: ImprimirMelhorSol () ***
*** Função: Imprimir a melhor solucao encontrada ***
*****/

```

```

*****/
void ImprimirSolucao(TSolucoes s)
{
    printf("\n\n\nSOLUCAO ENCONTRADA");
    printf("\n*****");
    printf("\n\nFO: %.3f", s.fo);
    printf("\n\nSequenciamento = \n");
    for (int i=0; i<m; i++)
    {
        for (int j=0; j<q; j++)
            printf("%d ", s.seq[i][j]);
        printf("\n");
    }
    printf("\n\nTempo total: %.3f", (float)(CPUfim - CPUinicial)/CLOCKS_PER_SEC);
    printf("\n\nTempo mel hor: %.3f\n", (float)(CPUmelhor - CPUinicial)/CLOCKS_PER_SEC);
}

/*****
*** Método: GerarSolucaoInicial() ***
*** Função: GerarSolucaoInicialAleatoriamente ***
*****/
TSolucoes GerarSolucaoInicial()
{
    TSolucoes s;

    //numero de produtos sequenciados
    int prod[p];

    for(int i=0; i<m; i++)
    {
        //inicializar o número de produtos sequenciados
        for (int k=0; k<p; k++)
            prod[k] = 0;

        //alocar os 0 produtos para a linha i
        for (int j=0; j<q; j++)
        {
            int x = irandomico(0, p-1);

            //encontrar um produto que nao tenha sido totalmente sequenciado
            while (prod[x] >= A[i][x])
                x = irandomico(0, p-1);

            s.seq[i][j] = x;
            prod[x]++;
        }
    }
    s.fo = CalcularFO(s);

    //imprimir solucao
    ImprimirSolucao(s);

    return s;
}

/*****
*** Método: SA() ***
*** Função: Aplica a metaheurística Simulated Annealing no MMAL ***
*****/
void SA()
{
    TSolucoes s, //solução corrente do SA
    sMelhor, //melhor solução do SA
    sViz; //solução vizinha

    double delta;
    int aquecimento = 0;

    //Ler os parametros
    float T0 = 1000000, //temperatura inicial
    T, //temperatura corrente
    alfa = 0.98; //taxa de resfriamento

    int SAmáx = 1000, //numero máximo de iteracoes
    IterT; //iteracao corrente

    //inicializar o tempo computacional do método
    CPUinicial = CPUfim = CPUmelhor = clock();

    //Gerar uma solucao inicial aleatoriamente
    s = GerarSolucaoInicial();

    sMelhor = s;
    IterT = 0;
    T = T0;

```

```

while (T > 0.0001)
{
    num1=0;
    while (IterT < SMax)
    {
        IterT++;
        //gerar vizinho aleatoriamente
        sViz = s;
        int k = irandomico(1,2);

        switch (k)
        {
            case 1://trocar dois produtos de posicao
            {
                int pos1 = irandomico(0, Q-1); //escolher duas posicoes
                int pos2 = irandomico(0, Q-1);
                while (pos1 == pos2)
                {
                    pos1 = irandomico(0, Q-1);
                    pos2 = irandomico(0, Q-1);
                }
                int linha = irandomico(0,m-1);
                sViz.seq[linha][pos1] = s.seq[linha][pos2];
                sViz.seq[linha][pos2] = s.seq[linha][pos1];
                break;
            }
            case 2:
            {
                int pos3=irandomico(0, Q-1);
                int pos4=irandomico(0, Q-1);
                while (pos3==pos4)
                {
                    pos4=irandomico(0, Q-1);
                    pos3=irandomico(0, Q-1);
                }
                int linha = irandomico(0,m-1);
                if(pos3<pos4)
                {
                    sViz.seq[linha][pos3]=s.seq[linha][pos4];
                    for(int a=pos3; a<pos4; a++)
                    sViz.seq[linha][a+1]=s.seq[linha][a];
                }
                else
                {
                    sViz.seq[linha][pos4]=s.seq[linha][pos3];
                    for(int a=pos4; a<pos3; a++)
                    sViz.seq[linha][a+1]=s.seq[linha][a];
                }
                break;
                num1++;
            }
        }

        //calcular a FO
        sViz.fo = CalcularFO(sViz);
        del ta = sViz.fo - s.fo;
        //definir a partir de qual solucao continuar a busca
        if (del ta < 0)
        {
            s = sViz;
            num1++;
            if (s.fo < sMelhor.fo)
            {
                sMelhor = s;
                CPUmelhor = clock();
            }
        }
        else
        {
            double x = randomico(0, 1);
            if ( x < (exp(-del ta/T)) )
            {
                s = sViz;
            }
        }
    }
}

```

```

        num1++;
    }
} //Fim-SAmax

printf("\nT = %.4f \t FO = %.3lf \t melhorFO = %.3lf", T, s.fo, sMelhor.fo);

//modificando os parametros de acordo com a temperatura
if(T>1000)
{
    alfa=0.98;
    SAmax=1000;
}
else
{
    if(T>100)
    {
        alfa=0.95;
        SAmax=1500;
    }
    else
    {
        alfa=0.80;
        SAmax=2000;
    }
}

T = T * alfa;
IterT = 0;

//aquecer a temperatura do SA
if (T < 0.01 && aquecimento == 0)
{
    T = T0 * 0.2;
    aquecimento = 1;
}

} //Fim-T

//finalizar o tempo computacional do método
CPUfim = clock();
tempoMelhor += (float)(CPUmelhor - CPUinicio)/CLOCKS_PER_SEC;
tempoTotal += (float)(CPUfim - CPUinicio)/CLOCKS_PER_SEC;

ImprimirSolucao(sMelhor);

//Gravando os resultados para ordenar
OrdBest[n]=sMelhor;
n++;
}

/*****
*** Método: randomico(double min, double max) ***
*** Função: Gera um numero aleatorio entre min e max ***
*****/
double randomico(double min, double max)
{
    return ((double)(rand()%10000)/10000.0)*(max-min)+min;
}

/*****
*** Método: irandomico(int min, int max) ***
*** Função: Gera um numero inteiro aleatorio entre min e max ***
*****/
int irandomico(int min, int max)
{
    return (int)randomico(0,max-min+1.0) + min;
}

```

SOLUCAO ENCONTRADA 1  
\*\*\*\*\*

F0: 27.862

Sequenci amento =  
3 1 3 3 1 3 2 3 0 2 3 1 3 2 2 0 3 3 2 3  
2 1 1 0 1 0 1 1 1 1 0 3 1 1 1 1 1 0 3 2  
1 2 2 1 3 2 3 1 0 3 0 3 2 3 3 0 1 0 0 1

SOLUCAO ENCONTRADA 2  
\*\*\*\*\*

F0: 28.301

Sequenci amento =  
1 1 1 3 2 3 2 0 3 3 2 3 2 3 3 0 3 3 2 3  
3 1 3 1 1 0 1 1 1 0 1 2 1 0 1 1 1 0 1 2  
3 3 3 2 3 0 3 0 1 2 3 1 0 2 1 0 1 2 0 1

SOLUCAO ENCONTRADA 3  
\*\*\*\*\*

F0: 28.355

Sequenci amento =  
1 1 3 1 2 3 2 0 3 3 2 3 2 3 0 3 3 3 2 3  
3 1 1 3 1 0 1 1 1 0 1 2 1 1 1 0 1 0 1 2  
3 3 2 3 3 0 3 0 1 2 3 1 0 2 1 0 1 0 2 1

SOLUCAO ENCONTRADA 4  
\*\*\*\*\*

F0: 28.385

Sequenci amento =  
2 0 2 3 2 3 1 3 3 1 3 2 3 0 3 3 3 2 3 1  
1 1 1 1 1 2 3 1 0 1 0 1 1 1 0 0 1 1 2 3  
3 0 3 2 3 1 0 2 1 3 2 3 0 1 0 1 2 0 1 3

SOLUCAO ENCONTRADA 5  
\*\*\*\*\*

F0: 28.426

Sequenci amento =  
1 1 3 1 2 3 2 3 3 0 2 3 2 3 3 0 3 3 2 3  
3 1 1 3 1 0 1 0 1 1 1 2 1 0 1 1 1 0 1 2  
3 3 2 3 3 0 3 2 1 0 3 1 0 2 1 0 1 0 2 1

SOLUCAO ENCONTRADA 6  
\*\*\*\*\*

F0: 28.473

Sequenci amento =  
1 1 3 1 2 3 3 2 0 3 2 3 3 3 2 0 3 3 2 3  
3 1 1 3 1 0 0 1 1 1 1 2 0 1 1 1 1 0 1 2  
3 3 2 3 3 0 1 3 0 2 0 1 1 2 3 0 1 0 2 1