

UNIVERSIDADE FEDERAL DE ITAJUBÁ

BRUNO LOPES MENDES TORGA

**Modelagem, Simulação e Otimização em Sistemas
Puxados de Manufatura**

Orientador: Prof. José Arnaldo Barra Montevechi, Dr.

Co-orientador: Prof. Alexandre Ferreira de Pinho, MSc.

Itajubá, 12 de fevereiro de 2007

2007	Bruno Lopes Mendes Torga	Dissertação de Mestrado
------	--------------------------	-------------------------

UNIVERSIDADE FEDERAL DE ITAJUBÁ

BRUNO LOPES MENDES TORGA

**Modelagem, Simulação e Otimização em Sistemas
Puxados de Manufatura**

Dissertação aprovada por banca examinadora em 12 de fevereiro de 2007, conferindo ao autor o título de *Mestre em Engenharia de Produção*

Banca Examinadora:

Prof. Paulo Eigi Miyagi, Dr. (USP)

Prof. João Roberto Ferreira, Dr. (UNIFEI)

Prof. Alexandre Ferreira de Pinho, MSc. (Co-orientador)

Prof. José Arnaldo Barra Montevechi, Dr. (Orientador)

Itajubá, 12 de fevereiro de 2007

Agradecimentos

Agradeço a DEUS por me possibilitar mais essa conquista. Por iluminar meus caminhos e por me guiar nos momentos de dificuldade. Agradeço também a meu pai, que nos deixou no meio do caminho, por todos seus ensinamentos, pelo exemplo e pelo incentivo, e dedico a ele não só esta conquista, mas tudo que eu possa alcançar ao longo de minha vida.

Agradeço a minha mãe e ao meu irmão pelo incentivo e por acreditarem em mim. Agradeço também a Adriana, minha namorada, pela dedicação, apoio, incentivo e compreensão. Agradeço também ao meu orientador, José Arnaldo, pelo exemplo, orientação, rigor e incentivo; Ao meu co-orientador, Alexandre Pinho, pela atenção, orientação e incentivo; A todos os professores do curso de Administração de Empresas e do Programa de Pós Graduação da UNIFEI que me acompanharam ao longo desses anos e pelo profissionalismo e incentivo.

Agradeço ainda a todos os meus amigos e ao Professor Wander, pela amizade, pelos conselhos e incentivo ao longo dessa caminhada.

Por fim agradeço a CAPES, pelo apoio financeiro e incentivo a pesquisa brasileira.

Sumário

Agradecimentos.....	iv
Sumário	v
Resumo	ix
Abstract.....	x
Lista de figuras	xi
Lista de tabelas	xiv
1.INTRODUÇÃO	1
1.1. Objetivos do trabalho	2
1.2. Relevância do Trabalho.....	3
1.3. Metodologia	3
1.4. Estrutura do trabalho	6
2.SISTEMAS DE PRODUÇÃO	8
2.1. Considerações iniciais	8
2.2. Sistemas de puxar e empurrar a produção.....	8
2.2.1. Definições	8
2.3. O Sistema Just in Time.....	11
2.3.1. Introdução	11
2.3.2. Definições	12
2.3.3. Princípios	12
2.4. O Sistema <i>Kanban</i>	17
2.4.1. Introdução	17
2.4.2. Definições	18
2.4.3. Funcionamento	19
2.4.4. Tipos de <i>kanban</i>	20
2.4.5. O Quadro <i>kanban</i>	23
2.4.6. O Sistema <i>Kanban</i> com um cartão	27
2.4.7. O Sistema <i>Kanban</i> com dois cartões.....	27

2.4.8. Cálculo do número de <i>kanbans</i>	30
2.4.9. O papel do <i>kanban</i> na redução do inventário	31
2.5. Considerações finais	32
3.SIMULAÇÃO COMPUTACIONAL	34
3.1. Considerações iniciais	34
3.2. Introdução	34
3.3. Vantagens e desvantagens.....	37
3.4. Definições	39
3.4.1. Sistemas	39
3.4.2. Modelos.....	40
3.4.3. Simulação.....	43
3.5. Metodologias de simulação.....	47
3.6. Verificação e validação de um modelo computacional.....	51
3.6. A simulação computacional em ambientes de manufatura	54
3.7. Considerações finais	55
4.OTIMIZAÇÃO	57
4.1. Considerações iniciais	57
4.2. Definições	57
4.3. Algoritmos Evolutivos	59
4.4. Algoritmos Genéticos.....	61
4.4.1. Funcionamento e nomenclatura.....	62
4.4.2. Operadores Genéticos	65
4.5. Considerações finais	71
5.OTIMIZAÇÃO via SIMULAÇÃO.....	72
5.1 Considerações iniciais	72
5.2 Introdução	72
5.3 Funcionamento.....	73
5.4. Metodologia para otimização via simulação.....	75
5.5. Casos na literatura	76
5.6. Considerações finais	78
6.APLICAÇÃO	79

6.1 Considerações iniciais	79
6.2. Descrição do objeto de estudo	79
6.2.1. Aspectos gerais	79
6.2.2. Funcionamento	80
6.3. Definição do problema	84
6.4. Modelagem do sistema	85
6.4.1. Formulação do problema	85
6.4.2. Definição dos objetivos e planejamento do projeto	86
6.4.3. Conceitualização do modelo	86
6.4.4. Coleta de dados	87
6.4.5. Construção do modelo	88
6.4.6. Execução do modelo	94
6.4.7. Verificação	94
6.4.8. Validação:	94
6.4.9. Planejamento dos experimentos	95
6.4.10. Realização e análise dos experimentos	96
6.4.11. Replicações extras	97
6.4.12. Documentação e relato	97
6.5. Otimização	97
6.5.1. Definição das variáveis	98
6.5.2. Definição dos tipos de variáveis	98
6.5.3. Definição da função objetivo	99
6.5.4. Seleção do tamanho da população do Algoritmo Evolutivo	100
6.5.5. Análise dos resultados	101
6.6. Considerações finais	103
7. CONCLUSÃO	104
7.1 Considerações iniciais	104
7.2. Conclusões e contribuições do trabalho	104
7.3 Sugestões para trabalhos futuros	105
7.4. Considerações finais	106
REFERÊNCIAS BIBLIOGRÁFICAS	107

ANEXO 1 116
ANEXO 2 139

Resumo

O presente trabalho aborda a aplicação conjunta de simulação e otimização em um sistema puxado de manufatura. Primeiramente é feita uma revisão da literatura abordando as metodologias descritas nesse trabalho como *just in time*, Sistema *kanban*, simulação computacional, otimização e otimização via simulação computacional. Tal revisão procura dar respaldo para a fase de aplicação, que ocorre em um sistema puxado de produção de uma empresa do setor automobilístico. Com o uso das ferramentas estudadas, o sistema produtivo foi modelado, validado e otimizado, verificando-se as mudanças no dimensionamento de cada peça modelada de acordo com a política seguida pela empresa e após a otimização. A otimização e a simulação foram executadas utilizando-se o pacote de simulação ProModel®, que inclui um *software* de otimização baseado em algoritmos genéticos, o SimRunner®. Este trabalho verifica a real potencialidade do uso conjunto da simulação e otimização na redução do número de *kanbans* de produção de um sistema puxado, se mostrando uma alternativa para análise no dimensionamento dos estoques.

Abstract

The present study approached the joint application of simulation and optimization in a pull system. First, a revision of the literature was elaborated to verify the researches about methodologies used in this work such as *just in time*, *Kanban system*, simulation, optimization and optimization via simulation. The propose of the revision is support the application phase, which was done in a pull production system of a company of the automobilist branch. With studied tools this cell was modeled, validated and optimized. The changes in the number of *kanbans* were verified comparing its values according to the numbers used in the study object and its values after optimization. The optimization and the simulation were executed using the simulation package ProModel® that includes optimization software based on genetic algorithms, SimRunner®. Finally, this work verified the real potentiality of the joint use of optimization and simulation in the reduction of *kanbans* in a pull system and its use as a alternative for analysis of the stock problems.

Lista de figuras

Figura 1.1. Classificações da pesquisa.....	5
Figura 2.1. Empurrar e puxar a produção.	9
Figura 2.2. Quebra cabeças <i>Just in Time</i>	16
Figura 2.3. Subdivisões dos <i>kanbans</i>	20
Figura 2.4. Exemplo de <i>kanban</i> de produção.	21
Figura 2.5. Exemplo de <i>kanban</i> de requisição.	22
Figura 2.6. Exemplo de <i>kanban</i> de fornecedor.	23
Figura 2.7. Quadro <i>kanban</i>	24
Figura 2.8. Quadro <i>kanban 2</i>	25
Figura 2.9. Cores indicativas do quadro <i>kanban</i>	26
Figura 2.10 <i>kanban</i> com um cartão.....	27
Figura 2.11. <i>kanban</i> com dois cartões.....	29
Figura 3.1. Simulação determinística X simulação estocástica.....	45
Figura 3.2. Simulação discreta X simulação contínua.....	46
Figura 3.3. Metodologia de condução de um estudo de simulação.....	51
Figura 3.4. Equilíbrio entre esforço e custo de validação.....	54
Figura 3.5. Realidade X modelo.....	55
Figura 4.1. Representação de um cromossomo.....	63

Figura 4.2. Ciclo de operações dos algoritmos genéticos.....	64
Figura 4.3. Pseudocódigo de um algoritmo genético.....	65
Figura 4.4. Exemplo de aplicação do operador de cruzamento de um ponto.....	68
Figura 4.5. Exemplo de aplicação do operador de cruzamento de múltiplos pontos.....	68
Figura 4.6. Exemplo de aplicação do operador de cruzamento uniforme.....	69
Figura 4.7. Exemplo de aplicação do operador de mutação.	69
Figura 5.1. Questionamentos na modelagem, simulação e otimização.....	74
Figura 5.2. Otimização em simulação.	75
Figura 6.1. Estrutura ABC de estoques.	79
Figura 6.2. Mapeamento de processos.	81
Figura 6.3. Fluxograma do objeto de estudo.....	85
Figura 6.4. Tela inicial do ProModel.....	87
Figura 6.5. Configuração dos locais.....	88
Figura 6.6. Configuração das entidades.....	88
Figura 6.7. Configuração das variáveis globais.....	89
Figura 6.8. Configuração das variáveis locais.....	89
Figura 6.9. Configuração dos atributos.....	90
Figura 6.10. Configuração das chegadas.....	90
Figura 6.11. Configuração do processamento.....	91
Figura 6.12. Modelo computacional do objeto de estudo.....	94

Figura 6.13. Tela de acesso do SimRunner a partir do ProModel.....	95
Figura 6.14. Definição das <i>macros</i>	96
Figura 6.15. Definição dos fatores de entrada e seus limites.....	97
Figura 6.16. Definição da função objetivo.....	98
Figura 6.17. Configuração do <i>setup</i>	99

Lista de tabelas

Tabela 2.1. Empurrar versus puxar.....	10
Tabela 3.1. Histórico da utilização da simulação computacional.	35
Tabela 3.2. Classificação de Sistema, Modelo e Simulação.....	47
Tabela 5.1. Softwares de otimização.....	73
Tabela 6.1. Quantidades de cartões nos meses de abril, maio, junho e julho.	83
Tabela 6.2. Comparação Real X Virtual.	93
Tabela 6.3. Comparação Produção real X Produção virtual.	93
Tabela 6.4. Resultados da otimização.....	100

Capítulo 1

INTRODUÇÃO

Este capítulo tem a finalidade de discutir aspectos importantes desta dissertação. Nesta seção são realizadas algumas considerações iniciais, apresentado o objetivo do trabalho, a relevância e a estrutura do trabalho.

Devido ao crescente interesse das empresas pelo Sistema Toyota de Produção, inúmeros trabalhos têm abordado o sistema *just in time* e a quantidade mínima de *kanbans* necessária para seu funcionamento (LABAS et al, 2000, MARTIM, et al, 1998, KOCHER E NIELANDER, 2000, CO E JACOBSON, 1994). Sistemas *just in time*, se implementados corretamente resultam em aumento de produtividade, redução do estoque em processo e aumento da qualidade dos produtos. Nestes tipos de sistemas, estoque em processo e *lead times* são considerados medidas de desempenho e o controle dos estoques ocorre através dos *kanbans*, que são cartões utilizados para requisitar a produção de determinados itens, movimentação de mercadorias dentro das fábricas e abastecimento de matéria prima. Além disso, possuem informações referentes ao que produzir, onde e para onde direcionar as mercadorias.

Uma das maiores aplicações da simulação computacional está na manufatura (LAW, 1999). Dentre os benefícios que a simulação pode trazer pode-se destacar a necessidade de se definir a quantidade de maquinário ou funcionários extras, avaliação de desempenho e avaliação dos procedimentos operacionais. As medidas de desempenho mais utilizadas são peças produzidas, tempo de espera das peças para serem processadas, porcentagem de utilização dos funcionários e das máquinas (LAW, 1999).

A simulação computacional é uma poderosa ferramenta na análise de processos e sistemas complexos. Tornando possível o estudo, a análise e a avaliação de situações que não seriam possíveis na vida real. Em um mundo em crescente competição, tem se tornado uma metodologia indispensável para os tomadores de decisão nas mais diversas áreas (SHANNON, 1998). JOHANSSON (2002) reforça esta diversidade de áreas de aplicação

mostrando como a simulação tem sido utilizada desde a análise de operações militares, operações logísticas, linhas de manufatura até operações na área de saúde.

Segundo HARREL *et al.* (2000), a otimização é o processo de tentar diferentes combinações de valores para variáveis que podem ser controladas (variáveis independentes), buscando uma combinação de valores que provê a saída mais desejada de um modelo. Alguns exemplos de programas que executam a otimização a partir da simulação são: AutoStat, OptQuest, OPTIMIZ, SimRunner, e o WITNESS Optimizer; que já estão incorporados a alguns pacotes comerciais de simulação. Esta incorporação fez com que a simulação computacional oferecesse respostas mais eficientes, possibilitando uma melhor tomada de decisão.

A associação da simulação com a otimização possibilita a busca dos melhores valores das variáveis de decisão de um determinado sistema onde o desempenho é avaliado através dos resultados da simulação. Inúmeros trabalhos têm abordado esta associação (CARSON E MARIA, 1997, BOWDEN E HALL, 1998, AZADIVAR, 1999, LEE e KIM, 2000, PINTO, 2001, DING *et al.*, 2003, SILVA, 2005), e este trabalho busca através da associação de tais conceitos otimizar o número de *kanbans* de um sistema puxado de manufatura.

1.1. Objetivos do trabalho

Esta dissertação faz parte de uma linha de estudo que dá continuação a três outros trabalhos, (OLIVEIRA, 2002, DUARTE, 2003 e SILVA, 2005). O que diferencia esta das duas primeiras é a associação dos conceitos de simulação e otimização. Já a última também utiliza esta associação de conceitos, porém sua maior diferença está na maneira como são aplicados. Pode-se afirmar que a contribuição deste trabalho perante os demais, tratando-se de uma evolução científica, é a modelagem e simulação de um sistema puxado de manufatura e a posterior otimização da quantidade de *kanbans* do sistema através da otimização via simulação. Além disso, planeja-se evidenciar a importância da técnica de mapeamento de processos na construção do modelo conceitual do sistema a ser simulado. Os fundamentos teóricos necessários para o entendimento desta aplicação são tratados ao longo deste trabalho.

1.2. Relevância do Trabalho

O Sistema *Kanban* é responsável pela produção dos itens certos, nas quantidades certas e na hora certa nos ambientes enxutos de manufatura. A quantidade de cartões no sistema relaciona-se diretamente com as quantidades armazenadas sob a forma de estoques de cada item. Existem inúmeras metodologias de cálculo das quantidades de *kanbans* e na mesma proporção, indícios de que os resultados obtidos são simplesmente as quantidades necessárias ao funcionamento do sistema, e não as quantidades mínimas, capazes de atender o mesmo nível de demanda minimizando as quantidades em estoques simultaneamente (TUBINO,1999).

Justifica-se desta forma a aplicação da simulação como ferramenta de análise de sistemas complexos, em especial sistemas puxados de manufatura, e da otimização como ferramenta de busca pelas melhores soluções, minimizando as quantidades de *kanbans* ao mesmo tempo em que se reduz os estoques.

Para TAVEIRA (1997), a simulação é uma ferramenta muito útil, uma vez que prevê o comportamento de sistemas complexos, calculando os movimentos e interações dos componentes do sistema. Juntamente com a simulação, a otimização vem sendo cada vez mais utilizada nas organizações, isso porque esta última ferramenta traz as melhores respostas ao modelo (PRICE e HARREL, 1999), economizando tempo e dinheiro em manipulações do modelo.

1.3. Metodologia

Segundo SILVA e MENEZES (2005), pesquisa é um conjunto de ações, propostas para encontrar a solução para um problema, que têm por base procedimentos racionais e sistemáticos. A pesquisa é realizada quando se tem um problema e não se têm informações para solucioná-lo.

Segundo os mesmos autores, uma pesquisa pode ser classificada quanto a sua natureza, abordagem, objetivos e procedimentos técnicos.

Quanto à natureza uma pesquisa pode ser *básica*, se objetivar gerar conhecimentos novos úteis para o avanço da ciência sem aplicação prática prevista. *Aplicada*, se objetivar gerar conhecimentos para aplicação prática e dirigida à solução de problemas específicos.

Quanto à abordagem pode ser *quantitativa*, considerando que tudo pode ser quantificável, o que significa traduzir em números opiniões e informações para classificá-las e analisá-las. Requer o uso de recursos e de técnicas estatísticas (percentagem, média, moda, mediana, desvio-padrão, coeficiente de correlação, análise de regressão, etc.). Pode também ser *qualitativa*, considerando que há uma relação dinâmica entre o mundo real e o objeto, isto é, um vínculo indissociável entre o mundo objetivo e a subjetividade do sujeito que não pode ser traduzido em números. A interpretação dos fenômenos e a atribuição de significados são básicas no processo de pesquisa qualitativa. Não requer o uso de métodos e técnicas estatísticas. O ambiente natural é a fonte direta para coleta de dados e o pesquisador é o instrumento-chave. É descritiva. Os pesquisadores tendem a analisar seus dados indutivamente. O processo e seu significado são os focos principais de abordagem.

Quanto aos objetivos pode ser *exploratória*, visando proporcionar maior familiaridade com o problema com vistas a torná-lo explícito ou a construir hipóteses. Envolve levantamento bibliográfico; entrevistas com pessoas que tiveram experiências práticas com o problema pesquisado; análise de exemplos que estimulem a compreensão. Assume, em geral, as formas de pesquisas bibliográficas e estudos de caso.

Pode também se *descritiva*, visando descrever as características de determinada população ou fenômeno ou o estabelecimento de relações entre variáveis. Envolve o uso de técnicas padronizadas de coleta de dados: questionário e observação sistemática. Assume, em geral, a forma de Levantamento.

Podendo também, além de exploratória e descritiva, ser *explicativa*, visando identificar os fatores que determinam ou contribuem para a ocorrência dos fenômenos. Aprofunda o conhecimento da realidade porque explica a razão, o “porquê” das coisas. Quando realizada nas ciências naturais, requer o uso do método experimental, e nas ciências sociais requer o uso do método observacional.

Finalmente, quanto aos procedimentos técnicos uma pesquisa pode ser classificada como *bibliográfica*, quando elaborada a partir de material já publicado, constituído principalmente de livros, artigos de periódicos e atualmente com material disponibilizado na Internet. *Documental*, quando elaborada a partir de materiais que não receberam tratamento analítico. *Experimental*, quando se determina um objeto de estudo, selecionam-se as variáveis que seriam capazes de influenciá-lo, definem-se as formas de controle e de observação dos efeitos que a variável produz no objeto. *Levantamento*, quando envolve a interrogação direta das pessoas cujo comportamento se deseja conhecer. *Estudo de caso*, quando envolve o estudo profundo e exaustivo de um ou poucos objetos de maneira que se permita o seu amplo e detalhado conhecimento. *Pesquisa Expost-Facto*, quando o “experimento” se realiza depois dos fatos. *Pesquisa-Ação*, quando concebida e realizada em estreita associação com uma ação ou com a resolução de um problema coletivo. E por último, *participante*, quando é desenvolvida a partir da interação entre pesquisadores e membros das situações investigadas.

De acordo com tal classificação este trabalho se enquadra dentro dos seguintes aspectos: Quanto a natureza, trata-se de uma *pesquisa aplicada*; Quanto a forma de abordagem, *quantitativa*; Quanto aos objetivos, *explicativa*; Quanto aos procedimentos técnicos, *estudo de caso*.

Neste trabalho também serão usadas ferramentas de modelagem e simulação, o que também a caracteriza como *simulação* (BERTRAND E FRANSOO, 2002).

A **Figura 1.1** representa de forma sintetizada a classificação da pesquisa quanto as classes e a metodologia seguida por este trabalho.

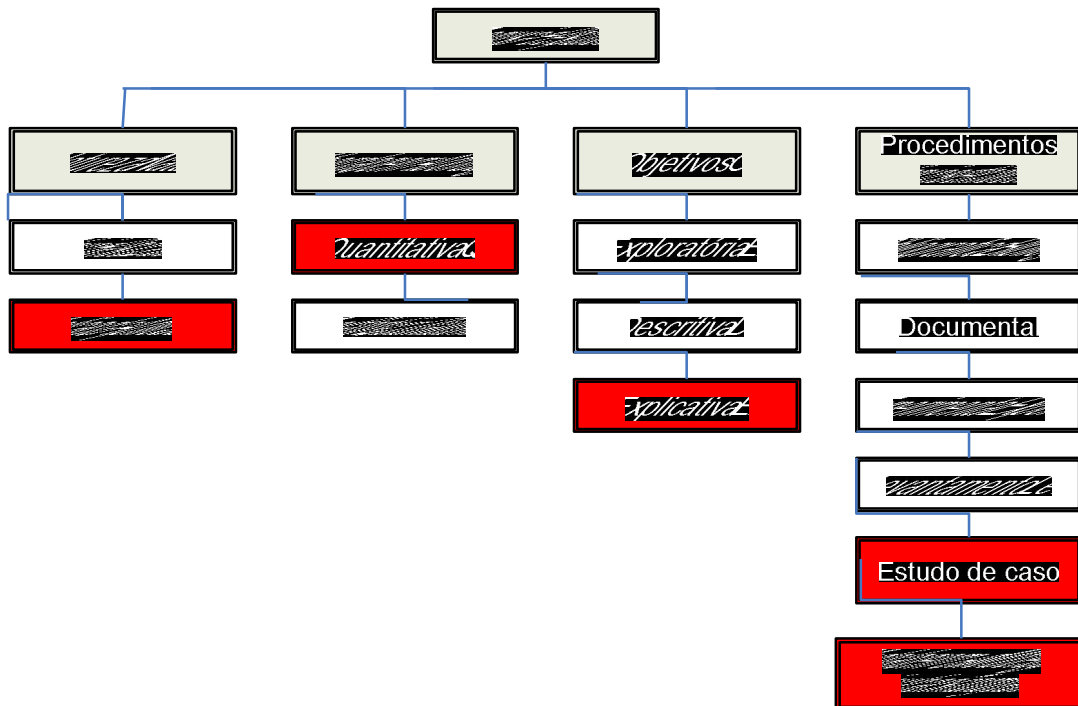


Figura 1.1: Classificações da Pesquisa
 Fonte: Adaptado de SILVA e MENEZES, 2005 e BERTRAND E FRANSOO, 2002.

A elaboração de hipóteses, ou simplesmente de afirmações iniciais que se buscará afirmar no decorrer da pesquisa, é algo essencial na elaboração e execução de um trabalho científico. Sendo assim, o presente trabalho busca afirmar a seguinte hipótese:

“A combinação entre simulação e otimização é de grande valia no apoio a decisão, trazendo vantagens significativas em relação às metodologias de cálculo de kanbans existentes na literatura e na redução de estoques?”

1.4. Estrutura do trabalho

Este trabalho está dividido em sete capítulos. O primeiro capítulo é destinado à introdução, fornecendo as primeiras impressões do trabalho, a relevância e o objetivo da dissertação. Nos capítulos dois, três, quarto e cinco são realizados os embasamentos teóricos, ou seja, a revisão bibliográfica das pesquisas realizadas acerca do Sistema *just in time*, Sistema *kanban*, simulação computacional, otimização e otimização via simulação, respectivamente. O capítulos seguintes

descrevem a aplicação da metodologia na solução da hipótese estabelecida anteriormente através da sistematização da aplicação da simulação e otimização. O capítulo sete traz as conclusões e recomendações para futuros trabalhos.

Capítulo 2

SISTEMAS DE PRODUÇÃO

2.1. Considerações iniciais

O objetivo deste capítulo é diferenciar os sistemas de produção puxados e empurrados comparando-os sob diversos aspectos para em seguida descrever o sistema *just in time*, suas principais características e fundamentos proporcionando uma base teórica para que o sistema *kanban*, que será explicado em seguida, possa ser melhor compreendido.

2.2. Sistemas de puxar e empurrar a produção

2.2.1. Definições

Segundo MOURA (1989), os processos de produção com inúmeras etapas podem ser classificadas em dois tipos: sistemas de empurrar e sistemas de puxar. A maioria dos tradicionais sistemas de produção utiliza os sistemas de empurrar, no entanto, o sistema *kanban*, utiliza o sistema de puxar a produção e o fluxo de materiais. A diferença básica é que o sistema *kanban* só reivindica e retira as unidades em processo da etapa anterior caso sua etapa seguinte também esteja consumindo itens.

No primeiro método, as peças estocadas em cada estágio são previstas, considerando o tempo e o fluxo total para a finalização do processo no estágio final. No segundo, existe um sistema onde uma certa quantidade de estoque é mantida em cada fase e cuja reposição é ordenada pelo processo seguinte, na proporção em que é consumida.

Segundo TUBINO (1997) “empurrar a produção” significa que o plano mestre de produção elabora periodicamente um programa de produção emitindo ordens de compra e ordens de fabricação. O próximo período de programação considera os estoques remanescentes ou por vir,

incorporando-os de forma natural. Como mencionado anteriormente, é como se os estoques empurrassem a produção. No sistema *kanban*, os estoques de material só entram na empresa ou são produzidos por um processo interno anterior de acordo com o que as linhas de produção subsequentes podem absorver. É como se a produção puxasse os estoques. A **Figura 2.1** ilustra a relação entre o PCP e os sistema de empurrar e puxar a produção

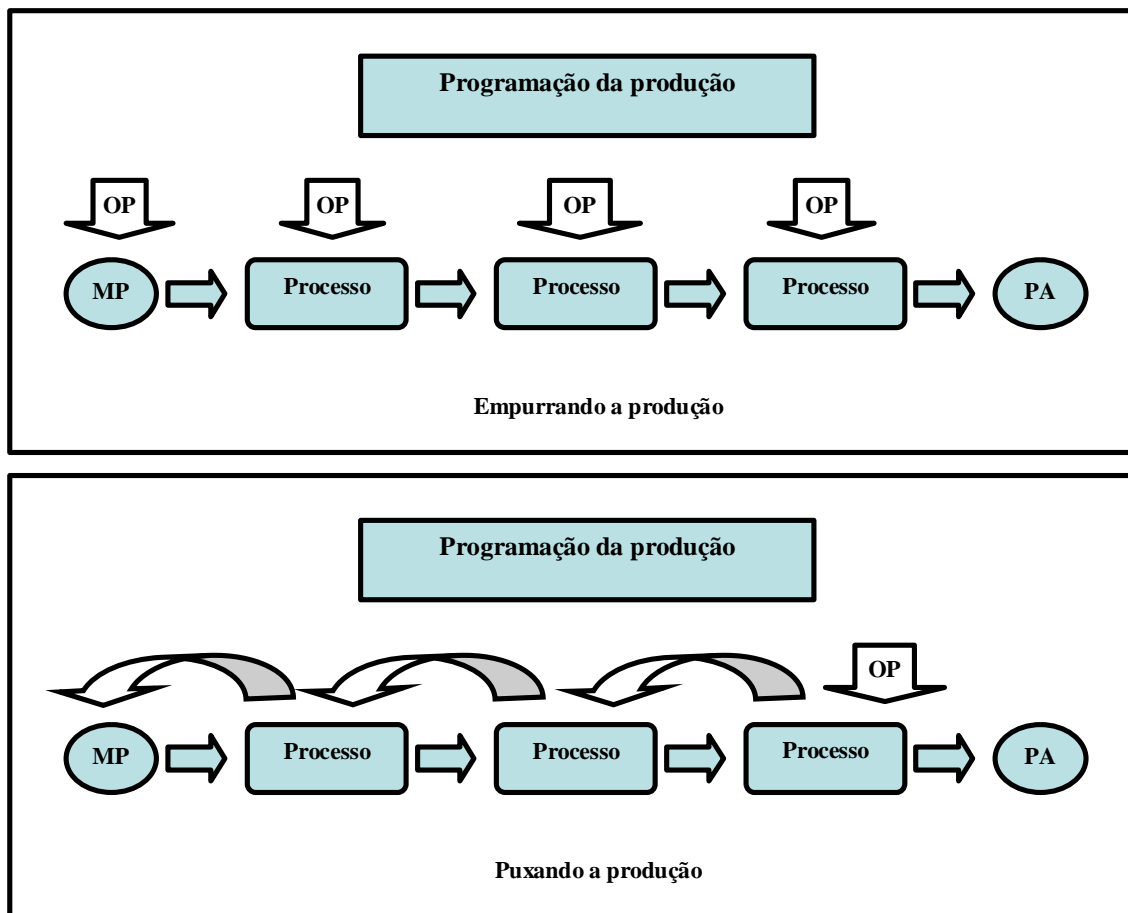


Figura: 2.1: Empurrar e puxar a produção
Fonte: TUBINO, 1997.

De uma forma mais direta pode-se dizer que no sistema empurrado o estoque comanda a produção, enquanto que no sistema puxado a produção comanda o estoque.

De acordo com HUTCHINS (1992), a principal vantagem do sistema “empurrado” é a previsibilidade da programação e carga das máquinas. Assim, torna-se possível fazer planos de

mix de produção variados, utilizando a análise computacional e dados advindos da área de marketing, das previsões realizadas e de pesquisas de mercado. No entanto, as coisas são diferentes para o sistema “puxado”, que depende menos de análises computacionais e mais da habilidade e o sistema responder a demandas repentinas e inesperadas. Esta é a principal vantagem do sistema *kanban*. A principal desvantagem é o risco de não atender a demanda inesperada.

Ambos os sistemas possuem vantagens e desvantagens, mas é preciso confrontá-los para decidir qual a melhor metodologia a ser utilizada. A **Tabela 2.1** compara as metodologias empurrar/puxar proporcionando um melhor entendimento em relação ao assunto:

Tabela 2.1: Empurrar versus Puxar

Sistemas	<p>Empurrar: Prevêem a demanda de peças estocadas ou material em processo em cada estágio, considerando o tempo de fluxo até o estágio final. De posse do valor previsto, todos os múltiplos estágios são controlados, justificando os estoques de produtos finais e peça em cada processa.</p> <p>Puxar: Possui certa quantidade de estoque em cada estágio. Uma operação posterior pede e retira peças da operação anterior somente na proporção e na hora em que consumi tais itens.</p>
Problemas	<p>A maioria dos sistemas convencionais corresponde ao de empurrar. E quanto maior se torna o sistema, mais aparecem os seguintes problemas:</p> <ol style="list-style-type: none"> 1. Quando ocorrem mudanças drásticas de demanda ou problemas na produção, torna-se impossível renovar os planos para cada processo. Além disso, é provável que tais dificuldades causem estoque em excesso ou mesmo inventário morto. 2. Torna-se impossível para os funcionários do controle de produção examinar todas as situações relativas ao índice de produção e ao nível de estoque. Assim, um plano de produção deve ter um estoque com excesso de segurança. 3. Melhorias relativas ao tamanho de lote e tempo de processamento podem não progredir, porque torna-se muito incômodo computar em detalhes os planos ótimos de produção.
Soluções	<p>O sistema de puxar tem sido idealizado como um meio de resolver tais problemas. Deve-se obter melhoramentos constantes, de maneira simples e confiável, e repor os itens à medida em que a área posterior os consome.</p>

Adaptado de Moura (2000)

A maneira de puxar a produção faz do sistema *kanban* um sistema relativamente simples de autocontrole no nível de fábrica e independente de gestões paralelas e controles computacionais (MOURA, 2000).

O Sistema *kanban* é um fundamento de extrema importância do sistema *just in time*, que associado a outros fundamentos de melhoria como ISO 9000, manutenção preventiva total, qualidade total, sistema multifuncional, *kaizen* e sistema 5 s podem reduzir o nível de estoque da empresa, eliminando desperdícios e aumentando a produtividade. Cada um desses componentes podem ser encarados como os alicerces principais do sistema *just in time*, o *kanban* pode ser considerado o coração do *JIT* (PEINADO, 2000).

2.3. O Sistema Just in Time

2.3.1. Introdução

A filosofia *just in time* de produção surgiu no Japão, na década de 60 na *Toyota Motor Company*, que buscava um modelo de produção simples e eficiente, capaz de otimizar a utilização dos seus recursos de capital, equipamento e mão-de-obra. Surgindo como uma alternativa à falta de escala de produção da indústria japonesa para competir com a indústria automobilística americana, que adotava o sistema de produção em massa desde o início do século XX e que se caracterizava pela produção de produtos altamente padronizados em grande escala de produção (GOMES, 2003).

Em meio a essa carência em termos de sistemas de produção, no Japão a empresa *Toyota Motor Company* buscava uma forma alternativa à produção em massa para gerenciar seu sistema de produção já que o antigo sistema já não se ajustava a difícil situação econômica do país naquela época. Surge, então, a “produção enxuta”, com princípios diferentes dos da produção em massa, particularmente em relação à gestão dos materiais (acabados, não acabados e em processo) e ao trabalho humano nas fábricas. Alguns fatores desse novo conceito, o *just in time*, a polivalência dos trabalhadores, o defeito zero, o *kaizen*, a produção em pequenos lotes, entre outros, passaram a ser os elementos principais do novo paradigma que se firmava. Por volta de 1970 esse sistema alcançou o seu auge e na década de 80, o país, com a quase total adoção do novo sistema,

alcançou índices de crescimento fantásticos em vários setores econômicos, lançando o país como potência reconhecida e consagrada mundialmente (CLETO, 2002).

2.3.2. Definições

A filosofia *just in time* objetiva a otimização dos recursos de capital, equipamento e mão de obra eliminando todo e qualquer desperdício que possa trazer custos indiretos, despesas desnecessárias e que não acrescente valor para a empresa. Resultando em um sistema de produção capaz de atender às exigências de qualidade e de entrega dos clientes ao menor custo (LUBBER, 1989).

Segundo o mesmo autor, o *just in time* pode ser descrito como uma filosofia de administração que está constantemente enfocando a eficiência e integração do sistema de manufatura, buscando uma simplificação cada vez maior de cada um de seus processos minimizando os elementos do sistema de manufatura que possam restringir a produtividade.

2.3.3. Princípios

Os cinco princípios básicos que orientam uma organização e seus colaboradores no desenvolvimento de um sistema *just in time* são:

1. Cada funcionário ou posto de trabalho é visto como cliente e fornecedor dentro do sistema;
2. Clientes e fornecedores são uma extensão do processo de manufatura;
3. Procurar continuamente simplificar cada processo;
4. É mais importante prevenir problemas do que resolvê-los;
5. Obter ou produzir algo somente quando for necessário (*just in time*).

Para OHNO (1997), um dos mentores da filosofia JIT, "a base do sistema Toyota de produção é a absoluta eliminação dos desperdícios. Os dois pilares necessários à sustentação do sistema são: *just in time* e automação". Mas o que vem a ser o *just in time* e a automação?

Segundo o mesmo autor, *just in time* significa que, em um processo de fluxo, as partes corretas necessárias à montagem alcançam a linha de montagem no momento em que são necessárias e

somente na quantidade necessária. Uma empresa que estabelece este fluxo integralmente pode chegar ao estoque zero. Se o *just in time* é realizado em toda a empresa, os estoques desnecessários ao longo do processo produtivo são eliminados, reduzindo, desta maneira, os investimentos e os custos relacionados aos estoques. Outro sustentáculo do sistema é a automação, que, segundo OHNO (1997), é conhecida também como "automação com o toque humano". A automação refere-se a automatizar um processo incluindo-se aí a inspeção: a inteligência humana complementa o automatismo na condução e operação de uma máquina. A atenção humana é necessária somente quando um defeito é detectado; neste caso pára-se a máquina e o fluxo da produção é interrompido, continuando somente após o problema ter sido resolvido OHNO (1997).

Para MOURA (2000), o *just in time* é uma abordagem disciplinada para melhorar a produtividade e a qualidade total, por meio do respeito pelas pessoas e através da eliminação das perdas. Durante a fabricação ou montagem de um produto, o *just in time* proporciona a produção no custo efetivo e a entrega apenas das peças necessárias com qualidade, na quantidade certa, no tempo e no lugar certos, e ao mesmo tempo, utiliza o mínimo de instalações, equipamentos, materiais e recursos humanos. Segundo o mesmo autor, o conceito envolve:

1. Fluxo de produção de uma peça: Um arranjo de máquinas agrupadas umas próximas das outras na seqüência necessária a produção das peças faz com que cada item seja processado um de cada vez do começo ao fim. Os benefícios são: *lead-times* menores, pouco inventário e material, nenhum transporte entre os processos e flexibilidade de produção.
2. Utilização de células de produção: Diz respeito a eliminação dos departamentos e a disposição das máquinas de forma a produzirem determinada peça ou *mix* de produtos. As peças com dimensão, forma, material e seqüências de processamento similares são distribuídas entre as linhas por um sistema conhecido como tecnologia de grupo.
3. *Set-up* rápido: A capacidade de trocar de ferramentas para passar a produção de outro tipo de produtos em menos pouco tempo é vital para atingir a meta *just in time*. As atividades do *set-up* rápido são divididas em duas atividades: A primeira relaciona-se com as atividades que podem ser realizadas enquanto as máquinas estiverem em funcionamento, e a segunda

relacionada com as atividades que só podem ser efetuadas a partir do momento em que a máquina estiver parada.

4. Relacionamento com fornecedores: Necessário para que os fornecedores externos possam atingir os níveis necessários de entrega (frequência) e qualidade.
5. Controle de qualidade: Sendo idealizado como uma responsabilidade de todos os departamentos, não apenas dos departamentos de manufatura, concentrando esforços para contribuir com a satisfação do consumidor, que é a principal avaliação do sucesso da empresa. Só quando o cliente está 100% satisfeito com um produto é que ele voltará a comprá-lo novamente e o recomendará para seus amigos fazerem o mesmo. O conhecimento das necessidades dos clientes torna-se imprescindível, assim como a capacidade da empresa de responder com rapidez e flexibilidade a qualquer mudança que possa vir a ocorrer.
6. O elemento humano: Envolve o esclarecimento para o funcionário dos objetivos da empresa e de como ela planeja atingi-los. Fazendo com que eles se sintam importantes nessa caminhada e valorizem seu trabalho.
7. Controles visuais e andon: No sistema JIT a produção seqüenciada produz lotes pequenos com variabilidade em turnos diferentes sem interrupção, assim é vital que a produção seja reiniciada o mais rápido possível caso haja algum problema. Para isso, são utilizadas uma série de lâmpadas de informação, ou sinais visuais conhecidos como “lâmpadas andon”. Como o *kanban* é um sistema visual para controle de produção do piso de fábrica, as lâmpadas andon são um sistema visual para parar a produção no chão de fábrica e identificar problemas. Nesse processo, o funcionário tem autoridade para parar a produção e em alguns casos, toda a linha, se a qualidade ou algum requisito não estiver dentro dos padrões.
8. Programação da Produção e Controle do Inventário: A programação e o controle do inventário utilizam o sistema *kanban*. Os cartões *kanban* são como dinheiro no caixa de uma loja. Os itens a serem vendidos são as peças que serão produzidas. E no lugar de produzir um número certo de produtos até o lote estar completo e empurrá-los para a próxima etapa, o próximo operário vai até a operação anterior e “compra” somente o que precisa e seu fornecedor produz somente o necessário para repor o que acabou de ser vendido para a

operação posterior. Dessa forma, o inventário flui de acordo com sua necessidade, contrastando com o sistema até então utilizado, onde o inventário flui de acordo com a produção.

9. **Kaizen:** Abordagem que visa assegurar que os processos da empresa satisfaçam as necessidades e ótimas expectativas dos clientes, não apenas no momento atual, mas de forma continuamente melhor ao longo do tempo. Significa desenvolver sistemas que sejam capazes de aprender, adaptar e inovar, ao contrário de simplesmente representar soluções. A implantação de tal metodologia envolve seis estágios: Identificação e seleção das oportunidades de melhoria; Estabelecimento de metas de melhoria; Análise do estágio atual, Geração e seleção de alternativas de aperfeiçoamento; Implementação das melhorias e Avaliação Contínua do Processo (HONDA e VIVEIRO, 1993).
10. **Sistema 5 S's:** Para que seja possível caminhar rumo ao sistema *just in time* é ponto básico que sejam observadas as condições mínimas de limpeza, identificação e organização das “coisas” na empresa. O sistema *kanban* trabalha com o critério de controle visual de estoque assim sendo, estes três aspectos são fundamentais para seu perfeito funcionamento. O sistema *kanban* vai exigir que os contentores estejam devidamente identificados e armazenados em seu local definido e os quadros para os cartões *kanban*, quando houver, também deverão ser manuseados de forma apropriada. Para se trabalhar com o sistema *kanban* é necessário ter disciplina e boa organização física dos estoques. É mais fácil que o sistema funcione em empresas que tiveram um programa nos moldes do 5S's. É preciso que exista uma cultura de preocupação com os aspectos de limpeza, organização e disciplina para que um programa *kanban* tenha êxito. Convém ressaltar que estes atributos são necessários para a implementação de qualquer espécie de programa de melhoria numa empresa (PEINADO, 2000).
11. **Multifunção:** Um sistema de multifunção numa empresa pode proporcionar várias vantagens rumo ao ambiente *just in time*. Um sistema de multifunção permite que o funcionário possa trabalhar em diversas áreas e atividades na empresa. Isto inegavelmente é bom para a empresa e também para o próprio funcionário que vai se sentir mais disposto e valorizado tendo seu trabalho menos monótono. A mão de obra nos dias de hoje deve ser participativa e polivalente (PEINADO, 2000).

12. **Qualidade assegurada:** Quando um item de material não possui qualidade assegurada de seu fornecedor este deverá passar por um sistema de inspeção de recebimento para poder ser utilizado na linha de produção. Estes ensaios podem ser demorados ou então também pode acontecer do material não ser aprovado, portanto sempre será necessário um estoque de segurança para prevenir uma parada de linha. Uma condição básica e imprescindível para que um item seja colocado em sistema de abastecimento *kanban* é que o mesmo seja recebido do fornecedor em qualidade assegurada. A lógica é simples: não é possível se trabalhar e calcular os estoques necessários para o sistema de abastecimento *kanban* se não se tem confiança total de que todas as peças ou material recebido poderá ser utilizado. Além disto o sistema *kanban* determina, na maioria dos casos, que o material seja entregue diretamente do fornecedor ao lado do montador da linha de produção (PEINADO, 2000).
13. **Manutenção Produtiva Total:** A necessidade de manutenções corretivas constantes, ou longo período de manutenção preventiva faz com que seja necessário um estoque de segurança para se evitar a falta de material nas linhas de produção durante estas paradas para manutenção. Assim sendo, para se atingir um ambiente *just in time* é necessário que a empresa implante um bom sistema de manutenção de suas máquinas e equipamentos, como por exemplo, a MPT (manutenção preventiva total). Um dos pontos fortes da manutenção produtiva total (MPT) é aquele em que se aproveita a “intimidade” que o operador adquire com a máquina que trabalha. Como analogia podemos comparar com o dono de um mesmo automóvel há bastante tempo. Caso este automóvel apresente qualquer ruído ou comportamento diferente do conhecido, o dono já prevê a necessidade de manutenção iminente. Empresas que implantaram um sistema MPT tiveram a necessidade de manutenções corretivas sensivelmente reduzidas (PEINADO, 2000).
14. **ISO 9000:** As empresas que possuem um sistema da qualidade no modelo requerido pela série ISO 9000 possuem uma cultura mais disciplinada para o cumprimento de normas. Como o *kanban* exige o cumprimento fiel a certas rotinas torna-se fácil implantar um sistema *kanban* em uma empresa que seja certificada pela ISO 9000. A certificação ISO 9000 é um dos requisitos para se obter um ambiente de *just in time* ou de qualidade total, porém vale ressaltar que a obtenção do certificado é apenas um dos passos rumo à qualidade total (SOARES, 1999).

A **Figura 2.2** abaixo ilustra como esses conceitos estão relacionados com o sistema *just in time*.

Sistema *Just In Time*

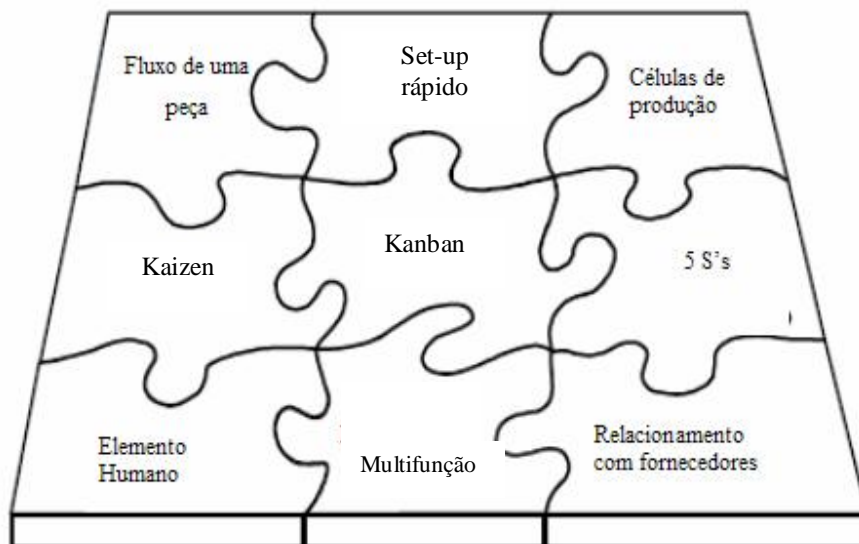


Figura 2.2: Quebra cabeças *Just In Time*
Adaptado de PEINADO (1999)

A razão de o sistema *Kanban* estar no centro relaciona-se com sua importância rumo a redução dos estoques. Sendo seu maior atrativo em relação aos sistemas tradicionais, a não formação dos estoques onde sua existência não é necessária e sua ausência onde é necessário. O Sistema *kanban* será melhor explicado no tópico seguinte.

2.4. O Sistema *Kanban*

2.4.1. Introdução

O mecanismo de controle da produção ou de movimentação de materiais no sistema de manufatura *just in time* é o *kanban*. Esta palavra de origem japonesa significa um sinalizador visual tal como uma etiqueta ou cartão. No contexto do sistema de manufatura *just in time*, o *kanban* representa um cartão contendo um código alfanumérico 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 possuir ou não um código de barras que faz a interface com o sistema de informação *ERP* (COELHO, 2003).

Trata-se de um cartão ou etiqueta de pedido de trabalho, sujeito a circulação repetitiva na área. Diferente das ordens tradicionais de trabalho, o *kanban* sempre acompanha os produtos facilitando o controle de estoques. É um método que reduz o tempo de espera, diminuindo o estoque, melhorando a produtividade e interligando todas as atividades da empresa em um fluxo constante e ininterrupto. O principal objetivo: transformação da matéria prima em produtos acabados, com tempos de espera iguais aos tempos de processamento, eliminando todo o tempo em fila do material e todo estoque ocioso (MOURA, 2000).

Existem duas interpretações para o termo *kanban* (MOURA, 2000):

1. Um sistema de controle de fluxo de material dentro da fábrica (*kanban* interno) e que pode se estender ao controle de material distribuído ou recebido por fornecedores (*kanban* externo).
2. Um sistema para melhorar a produtividade, visando mudanças nos equipamentos, nos métodos de trabalho e nas práticas de movimentação de material, que usa o sistema de controle de material por cartões (*kanbans*) para identificar as áreas com problemas e avaliar os resultados das alterações.

2.4.2. Definições

O *kanban* é um dos instrumentos essenciais para a implantação do *just in time* (MOURA, 2000, HUTCHINS, 1989, SHINGO,1996). É visto como uma forma de controlar o estoque em processo, a produção e as entregas dos fornecedores (ESPARRAGO, 1988).

O sistema *Kanban* utiliza cartões para autorizar a produção e movimentar material (matéria prima ou mercadorias semi processadas) entre centros de trabalho. Entretanto, um *kanban* não precisa necessariamente ser um cartão, pode vir a ser um comando verbal, uma luz, uma bandeira ou qualquer outro tipo de sinal (ESPARRAGO, 1988).

Para SHINGO (1996) *kanban* significa “etiqueta” ou “cartão” e é utilizado como meio de controle e coordenação e de satisfazer as seguintes funções:

1. O *kanban* indica o que produzir, quanto produzir, quando produzir e para onde levar os produtos.
2. A quantidade dos itens que atravessam a produção é controlada pelo número de *kanban*;
3. O *kanban* tornou possível uma resposta mais flexível a variações de demanda através da simplificação das instruções, já que implica em ausência de controles computacionais;
4. O número de *kanban* emitidos é controlado para responder a mudanças na carga e resolver problemas.

2.4.3. Funcionamento

Nos sistemas empurrados de produção, elabora-se periodicamente, para atender ao programa mestre de produção (PMP), um programa de produção completo, da compra de matéria prima à montagem do produto acabado, transmitindo-o aos setores responsáveis através da emissão de ordens de compra, fabricação e montagem, não sem antes passá-lo por uma etapa de seqüenciamento, para adequá-lo as restrições de capacidade física do processo produtivo. No próximo período de programação, em função dos estoques remanescentes, programam-se novas ordens para atender a um novo PMP (TUBINO, 1999).

No sistema *kanban* de puxar a produção não se produz nada até que o cliente(interno ou externo) de seu processo solicite a produção de determinado item. Assim, a programação da produção usa as informações do PMP para emitir ordens apenas para o último estágio do processo produtivo, assim como para dimensionar as quantidades de *kanbans* dos estoques em processo para os demais setores. Quando o cliente de um processo necessita de itens, ele recorre aos *kanbans* em estoque nesse processo, acionando diretamente o processo para que os *kanbans* dos itens consumidos sejam fabricados e repostos aos estoques. Desse modo, o sistema *kanban* de puxar a produção distribui por todas as subfábricas quantidades previamente calculadas de estoque para fazer a conexão entre dois pontos de trabalho relacionados. Seja entre células, entre células e a linha de montagem, ou entre fornecedores externos e os usuários internos (TUBINO, 1999).

O sistema *kanban* se propõe a eliminar os almoxarifados centralizadores. Cabe ressaltar que internamente nas células e nas linhas de montagem não se empregam *kanbans*, visto que nesses casos busca-se fluxo de produção unitário (TUBINO, 1999).

O sistema *kanban*, na sua forma de agir, simplifica em muito as atividades de curto prazo desempenhadas pelo planejamento e controle da produção dos sistemas de produção *just in time*, delegando-as aos próprios funcionários do chão de fábrica (TUBINO, 1999).

2.4.4. Tipos de *kanban*

O sistema *kanban* funciona baseado no uso de sinalizações para ativar a produção e a movimentação dos itens pela fábrica. Os cartões *kanban* tradicionais são confeccionados de material durável para suportar o manuseio constante em função do giro constante entre os estoques do cliente e do fornecedor do item. Cada empresa confecciona seus próprios cartões de acordo com suas necessidades de informações (TUBINO, 1999).

Conforme suas funções, os cartões dividem-se em dois grupos: os *kanbans* de produção e os *kanbans* de requisição ou movimentação. A **Figura 2.3** esquematiza essa subdivisão dos *kanbans*.

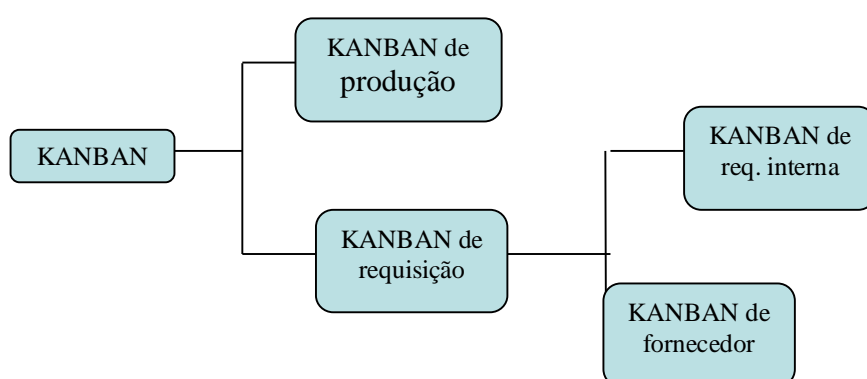


Figura 2.3: Subdivisões dos *kanbans*
Fonte: TUBINO (1999)

Os *kanbans* de produção autorizam a fabricação ou montagem de determinado lote de itens em um determinado centro de trabalho ou célula que executa a atividade produtiva nos itens. Quando

um lote de peças é retirado de um ponto de saída, o cartão de produção é enviado a produção para que outro lote seja produzido e substitua o retirado anteriormente (MOURA, 2000).

Dentre as informações que um *kanban* de produção deve conter destacam-se (TUBINO, 1999):

1. Especificação do processo e do centro de trabalho ou célula onde esse item é produzido;
2. Descrição do item, com o código e especificação do mesmo;
3. Local onde o lote deve ser armazenado após a produção;
4. Capacidade do contêiner ou tamanho do lote que será fabricado;
5. Tipo de contêiner para esse item;
6. Número de emissão desse cartão em relação ao número total de cartões de produção para esse item;
7. Relação dos materiais necessários para a produção desse item e local onde se deve buscá-los.

A **Figura 2.4** representa um exemplo de *kanban* de produção:


Processo		Centro de		
N° do item		Local de estocagem		
Nome do item				
Materiais necessários		Tamanho do lote	N° de emissão	Tipo de contêiner
Código	Localização			
				

Figura 2.4: Exemplo de *kanban* de produção
Fonte: TUBINO (1999)

Os *kanbans* de requisição ou movimentação autorizam a movimentação de lotes entre as estações de alimentação e o local de utilização. Podendo esta requisição ser interna ou externa à

organização. Podendo neste ultimo caso ser denominado também *kanban* de fornecedor (MOURA, 2000).

Dentre as informações que um *kanban* de requisição deve conter destacam-se (TUBINO, 1999):

1. Descrição do item, com o código e especificação do mesmo;
2. Especificação do centro de trabalho ou célula onde o item é produzido também chamado de centro de trabalho precedente, e local onde encontra-se armazenado o lote;
3. Especificação do centro de trabalho ou célula onde o item é consumido, também chamado de centro de trabalho subsequente, e local onde deve-se depositar o lote requisitado;
4. Capacidade do contêiner ou tamanho do lote que será movimentado;
5. Tipo de contêiner para esse item;
6. Número de emissão desse cartão em relação ao numero total de cartões de requisição para esse item.

A **Figura 2.5** representa um exemplo de cartão *kanban* de requisição:


N° de item			Centro de trabalho precedente
Nome de item			
			Local de estocagem
Tamanho do lote	N° de emissão	Tipo de contêiner	Centro de trabalho subsequente
			Local de estocagem

Figura 2.5: Exemplo de *kanban* de requisição

Fonte: TUBINO (1999)

Dentre as informações que um *kanban* de fornecedor deve conter destacam-se (TUBINO, 1999):

1. Nome e código do fornecedor autorizado a fazer entrega;
2. Descrição do item a ser entregue, com o código e especificação do mesmo;
3. Especificação do centro de trabalho ou célula onde o lote do item deve ser entregue, e local onde deve-se depositar o lote requisitado;
4. Lista de horários em que se deve fazer as entregas dos lotes e ciclo em número de vezes por período, normalmente diário;
5. Capacidade do contêiner ou tamanho do lote que será entregue;
6. Tipo de contêiner para esse item;
7. Número de emissão desse cartão em relação ao número total de cartões de fornecedor para esse item.

A **Figura 2.6** representa um exemplo de cartão *kanban* de fornecedor:



Nome e código do fornecedor <input type="text"/>	Local de		Local de estocagem
	Nº do item		
Horário de entregas ____	Nome do item		
Ciclo de entregas <input type="text"/>	Tamanho do lote	Nº de emissão	Tipo de contêiner
			

Figura 2.6: Exemplo de *kanban* de fornecedor
Fonte: TUBINO (1999)

2.4.5. O Quadro *kanban*

O sistema *kanban* tradicional emprega painéis ou quadros de sinalização, junto aos pontos de armazenagem espalhados pela produção, trata-se de uma representação visual do estoque, pois

através dele é possível saber como estão os níveis de estoque. Se os quadros *kanban* forem corretamente monitorados, muitos benefícios poderão ser obtidos em relação a otimização do estoque (PEINADO, 2000).

Para entender o funcionamento do quadro, serão formuladas algumas questões (PEINADO, 2000):

1. Como o quadro é desenhado? A **Figura 2.7** representa um modelo de quadro *kanban*. Neste exemplo, o quadro serve para controlar seis itens de estoque. O nome dos itens está descrito na primeira linha sob a forma das letras de “A” a “F”. A coluna do item “A” possui cinco linhas formando cinco lugares representados pelo fundo cinza, um lugar para cada um dos cinco cartões que formam o estoque deste item; os dois lugares no final da coluna não são utilizados neste caso. Cada cartão representa um lote do item “A”. A coluna do item “B” possui sete locais, portanto contém sete cartões e sete lotes do item no circuito. Assim sucessivamente o item “C” tem seis cartões, o item “D” tem quatro cartões, o item “E” apenas três e o item “F” tem cinco cartões.

Como o dimensionamento do número de cartões *kanban* no sistema é um processo dinâmico, os espaços vazios sob as colunas dos itens a, c, d, e, f não são utilizados.

Peça A	Peça B	Peça C	Peça D	Peça E	Peça F
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>			
	<input type="checkbox"/>				

Figura 2.7: Quadro *kanban*
Fonte: Adaptado de MOURA (2000)

3. Qual item deve ser produzido primeiro? Supondo que um funcionário pretende iniciar a produção de um destes itens. Ele observa no quadro, conforme **Figura 2.8** a seguinte situação: Item “A” com dois cartões no quadro; item “B” com quatro cartões no quadro; item

“C” sem nenhum cartão no quadro; item “D” com um cartão no quadro; item “E” com dois cartões no quadro e item “F” sem cartão no quadro.

Peça A	Peça B	Peça C	Peça D	Peça E	Peça F
<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>			<input type="checkbox"/>	
	<input type="checkbox"/>				
	<input type="checkbox"/>				

Figura 2.8: Quadro *kanban* II
 Fonte: Adaptado de MOURA (2000)

Qual desses itens apresenta a maior prioridade de produção? Em uma primeira observação pode-se pensar que deverá ser o item que possui o maior número de cartões no quadro, ou seja, o item “B” que apresenta quatro cartões. Por outro lado, o número total de contêineres do item “B” é sete e o número total de contêineres do item “E” é três, isto o torna mais crítico. Sendo assim, o item “E” possui prioridade de produção em relação aos outros itens.

3. Como saber qual é o item mais crítico? Conforme **Figura 2.9** cada linha das colunas desses painéis é pintada com uma cor para facilitar a visualização da urgência em se requisitar ou produzir este item. Normalmente, emprega-se a cor verde para indicar condições normais de requisição ou produção, a cor amarela para indicar “atenção” com este item, e a cor vermelha para sinalizar urgência na requisição ou produção deste item (TUBINO, 1999).

A regra é sempre fazer o que se encontra mais crítico, porém, se existir mais de um item com a mesma prioridade então se deve optar em produzir o item que for mais fácil e conveniente de se fabricar.

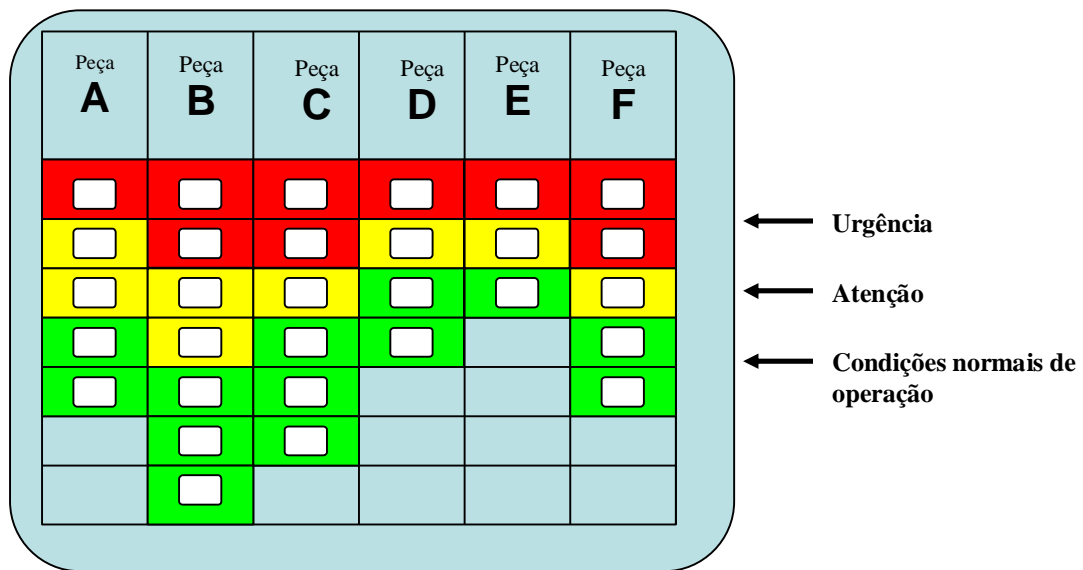


Figura 2.9. Cores indicativas do quadro *kanban*
Fonte: Adaptado de MOURA (2000) e TUBINO (1999)

4. O que fazer se o quadro estiver vazio? Vamos supor que em um dia qualquer o quadro esteja vazio. A indagação que surge é que peça deverá ser produzida em primeiro lugar? A resposta a esta pergunta deveria soar de forma veemente: “Não se deve produzir nada se o quadro estiver vazio”, porém a tradição pede para produzir. Isto parece estranho para alguns supervisores, gerentes e até mesmo para os próprios funcionários: É desperdício produzir estoques sem necessidade, mas porque então existe a tendência de se produzir? A resposta é simples: A cultura da empresa neste caso é tradicional, voltada à produção, o funcionário deve estar produzindo. O prejuízo de se fazer estoques desnecessários dispensa maiores comentários.

5. Como saber qual item está em produção? O quadro também deve indicar quais peças estão sendo produzidas em determinado momento, pois além de ser uma informação importante para indicação da posição de estoques, também serve para impedir a eventual possibilidade de duas pessoas produzirem o mesmo item.

Uma vez descritos as características e funcionamento do sistema *kanban* torna-se necessário discutir suas alternativas operacionais mais encontradas, que são: o sistema *kanban* com um cartão e o sistema *kanban* com dois cartões.

2.4.6. O Sistema *Kanban* com um cartão

A **Figura 2.10** ilustra o funcionamento de um sistema *kanban* de um cartão, onde a situação 1 reflete um cenário onde o quadro *kanban* encontra-se vazio, ou seja, não existe a necessidade de se produzir. Já na situação 2, o consumidor requisita as peças e então o cartão é alocado no quadro *kanban*, indicando a necessidade de se produzir o item. Na situação 3 as peças são produzidas e o cartão é retirado do quadro *kanban* e a situação 1 é retomada.

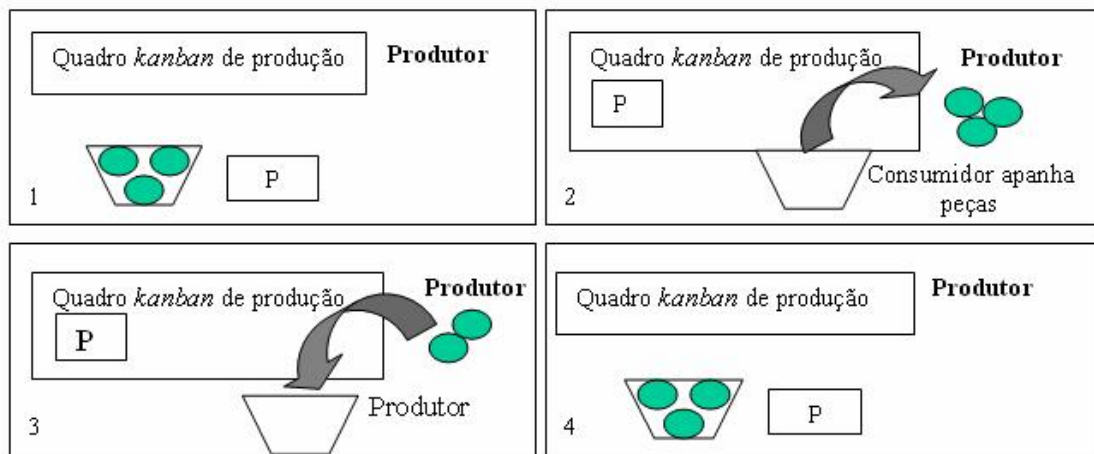


Figura 2.10: Kanban com um cartão
Fonte: Adaptado de PEINADO (2000)

2.4.7. O Sistema *Kanban* com dois cartões

A **Figura 2.11** ilustra o funcionamento de um sistema *kanban* de dois cartões, onde a situação 1 reflete um cenário onde os quadros *kanban* encontram-se vazios, e os contêineres, tanto do consumidor como do fornecedor se encontram abastecidos. Nesse exemplo são utilizados dois tipos de cartões, o cartão da área fornecedora, denominado cartão de produção e o cartão da área consumidora, denominado cartão de movimentação.

Nas situações 2 e 3, a área consumidora do item ao necessitar do mesmo, retira o cartão de movimentação do contêiner e coloca-o no quadro, passando a consumir as peças que estavam naquele contêiner.

Logo em seguida, o transportador verifica que existe um cartão de movimentação no quadro da área consumidora e leva o cartão de movimentação para a área fornecedora.

Na situação 4 o transportador retira o cartão de produção do contêiner cheio e o coloca no quadro *kanban* do fornecedor. Logo após a produção, o cartão de movimentação é colocado no contêiner cheio e enviado para a área consumidora.

Na situação 5, a área produtora do item verifica a existência de um cartão no quadro e produz mais um contêiner da peça, quando o contêiner estiver completo a área produtora retira o cartão do quadro e o coloca no contêiner novamente.

Na situação 6 temos os quadros *kanban* vazios e os contêineres cheios com os respectivos cartões fixados neles.

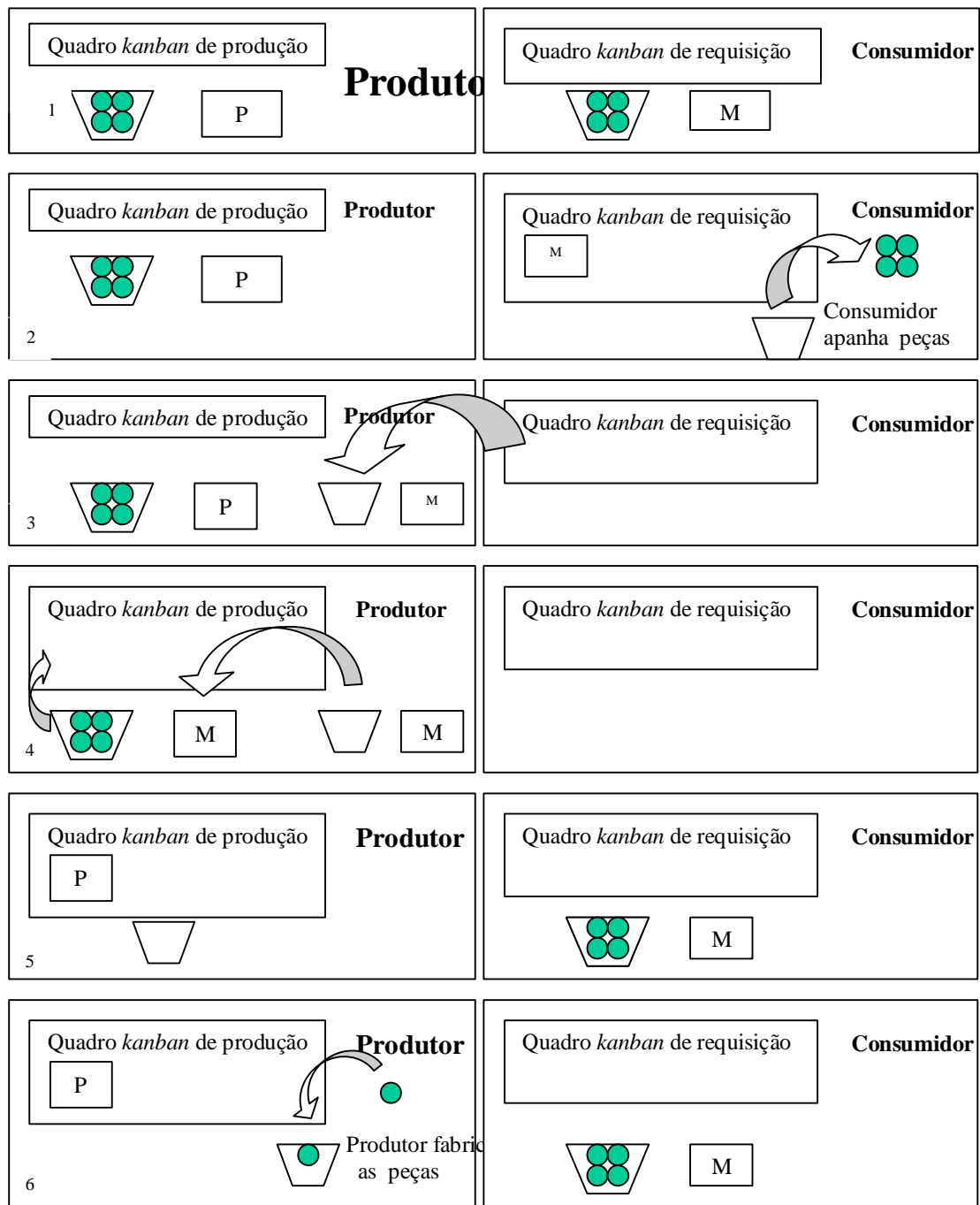


Figura 2.11: *Kanban* com dois cartões
 Fonte: Adaptado de PEINADO (2000)

2.4.8. Cálculo do número de *kanbans*

A quantidade em estoque de cada peça é igual ao número de *kanbans* distribuídos para aquela peça vezes o número de peças em cada contêiner padrão (o número de contêineres padrão é igual ao número de cartões *kanban* para aquela peça) (MOURA, 2000). Sendo assim, quantos cartões são necessários?

SHINGO (1984) afirma que no sistema Toyota de produção, a determinação do número de *kanbans* está muito longe de ser tão importante quanto o aperfeiçoamento do sistema de produção para minimizar o número de *kanbans*.

Apesar disso foram desenvolvidas formulas para a determinação do número de *kanbans* que consideram fatores como previsão de demanda, tempo de processamento dos produtos e o tempo de espera entre processos. Entretanto, com o intuito de facilitar o cálculo do número de *kanbans* e agilizar o início da implantação do sistema, SHINGO (1984) apresenta uma formula simples, ilustrada na equação 2.1, que pode ser utilizada para determinar o número de cartões necessários.

$$\text{Número de Kanbans (N)} = \frac{\text{Estoque máximo (Q + á)}}{\text{Capacidade de 1 pallet (n)}} \quad (2.1)$$

Onde:

N = Número de *kanbans*

Q = Quantidade de produtos no lote de produção

á = Estoque mínimo de segurança

n = Quantidade de produtos transportados em um pallet

Além dessa, existem outras formulas capazes de calcular o número de *kanbans* (MOURA, 2000, TUBINO, 1999, MONDEN, 1998). Entretanto, a determinação do número de *kanbans* está longe de ser tão importante quanto o aperfeiçoamento do sistema produtivo para minimizar sua quantidade. Se o processo é melhorado seja pela redução do tamanho do lote de produção ou

pela redução do tempo de processamento, o número de *kanbans* pode ser reduzido. Além disso, Shingo (1984) afirma que a experiência nos ensina que flutuações na ordem de 10 a 30% podem ser administradas sem alterar o número de *kanbans* em circulação. A implementação real é o guia mais confiável, e esses valores irão variar de acordo com a natureza da fábrica.

Para evidenciar esta afirmação basta examinar os componentes que constituem as fórmulas, que uma vez reduzidos ou otimizados, impactam diretamente no número de cartões, que são: Média de produção diária; Tempo de processamento necessário para um cartão de produção completar o ciclo completo de produção da peça; Tamanho do lote, que deve ser no máximo 10% das necessidades diárias (MOURA, 2000). Coeficiente de segurança, condizente com um coeficiente de variação determinado de acordo com a eficiência do sistema produtivo.

Outros exemplos de fórmulas encontrados em MONDEN (1998), TUBINO (1999) e MOURA (2000), são representados nas equações 2.2, 2.3 e 2.4:

$$\text{Kanbans} = \frac{\left[\frac{\text{Demanda_Mensal}}{\text{Setups_Mensais}} \right] + [\text{demanda_diaria} * \text{coeficiente_seguran\c{a}}]}{\text{Capacidade do container}} \quad (2.2)$$

$$\text{Kanbans} = \left[\frac{\text{Demanda_diária}}{\text{Tamanho do lote}} \right] * \text{Tempo de ciclo(\%)} * (1 + \text{coeficiente_seguran\c{a}}) \quad (2.3)$$

$$\text{Kanbans} = \frac{\text{Prod. Diária média} * (\text{Tempo_espera} + \text{Tempo_processamento}) * (1 + \text{coeficiente_seguran\c{a}})}{\text{Tamanho do lote}} \quad (2.4)$$

2.4.9. O papel do *kanban* na redução do inventário

Segundo PEINADO (2000), o sistema *just in time* não tem como objetivo principal atingir o estoque zero, mas sim eliminar desperdícios. A redução dos estoques deve ser encarada como uma consequência da eliminação de desperdícios. Mas qual o papel do *kanban* dentro deste contexto? Responsáveis pela administração de áreas relacionadas à logística e produção, podem ter em mente a idéia do *kanban* como sendo apenas uma metodologia de abastecimento de

estoque. Entretanto, se o *kanban* for considerado como apenas uma forma de controle de estoques, sua implantação terá grande chance de ser tratada de maneira isolada, deixando de considerar a existência ou até a necessidade de outros projetos atuando em paralelo e em conjunto.

Segundo o mesmo autor o funcionamento de um sistema *kanban* depende de outros fatores que devem ser encarados como alicerces de sua sustentação. Um projeto para a implementação de um sistema *kanban* deve levar em consideração vários outros componentes, como, por exemplo, sistemas de limpeza e organização tal como o conhecido 5 S's, sistemas de multifunção de funcionários, sistemas da qualidade tais como a ISO-9000, sistemas de desenvolvimento de fornecedores de materiais com qualidade assegurada, sistemas de manutenção das máquinas a exemplo da MPT - manutenção produtiva total, e assim por diante.

O *kanban* apenas limita o nível máximo dos estoques e se este for o único projeto sendo trabalhado na empresa, então os estoques não poderão ser reduzidos porque todos os sistemas de melhorias agem de forma conjunta. Todos esses componentes podem ser encarados como os alicerces principais do sistema *just in time*, o *kanban* pode ser considerado o coração do *JIT* (PEINADO, 2000).

Se fosse possível colocar o trabalho de implantação de um sistema *kanban* dentro de uma escala percentual de 0% a 100% pode-se afirmar de que o trabalho técnico, ou seja, os cálculos das quantidades e tipo de contêineres, definição da forma dos cartões, confecção dos quadros e demais atividades desta natureza, ocuparão dedicação inferior a 20% do total do tempo e energia que deverão ser consumidos para a implantação efetiva do sistema. Os demais 80% serão dedicados a mudar a forma de pensar das pessoas da organização (PEINADO, 2000).

2.5. Considerações finais

A fundamentação teórica realizada neste capítulo é fundamental para o entendimento dos objetivos deste trabalho e para a utilização da simulação computacional na modelagem de um

sistema puxado, cujo funcionamento se baseia no sistema kanban. A definição de simulação será realizada no capítulo seguinte.

Capítulo 3

SIMULAÇÃO COMPUTACIONAL

3.1. Considerações iniciais

O objetivo deste capítulo é apresentar uma revisão bibliográfica a respeito da simulação computacional. Inicialmente foi levantado o histórico e a evolução da simulação computacional desde a década de 50 e as vantagens e desvantagens de sua utilização. Conceitos como sistema, modelo e simulação serão descritos, assim como suas classificações. No fim do capítulo será dada atenção especial a simulação de ambientes de manufatura, considerada uma das técnicas mais populares para se analisar problemas complexos de ambientes de produção (O’KANE *et al.*, 2000 e TAVEIRA, 1997).

3.2. Introdução

A simulação computacional vem a ser a representação funcional de um sistema real através de um modelo de grande precisão através do computador, trazendo a vantagem de se poder visualizar a dinâmica desse sistema, implementar mudanças, respondendo a questões tipo: “o que aconteceria se” (*what-if*), desta maneira economizando recursos econômicos e tempo (PEREIRA, 2000).

A aplicação desta ferramenta teve início na década de 60 juntamente com a introdução dos primeiros computadores no mercado, já que devido ao grande número de cálculos matemáticos necessários, sua utilização sem o auxílio de um computador seria inviável. As primeiras aplicações desta ferramenta foram na área militar, tendo como objetivo principal o planejamento da distribuição de suprimentos nas frentes de batalha e alocação de recursos.

A simulação computacional foi desenvolvida primeiramente utilizando-se linguagens de programação de propósito geral, principalmente o FORTRAN e PASCAL (GAVIRA, 2003). No entanto, sistemas de maior complexidade apresentavam limitações tanto na modelagem como na execução da simulação tornando a utilização da ferramenta em alguns casos inviável.

A **Tabela 3.1** mostra o histórico de utilização da simulação desde seus primórdios até os dias atuais, onde se presencia o maior desenvolvimento dos computadores e como conseqüência uma maior possibilidade de aplicação da ferramenta em função da facilidade de uso em decorrência do surgimento de interfaces mais amigáveis a partir da década de 90.

Tabela 3.1: Histórico da utilização da simulação computacional

Décadas	Ferramenta	Características do estudo da simulação	Exemplos
50 e 60	Linguagens de propósito geral	Aplicações em grandes corporações; Grupos de desenvolvimento de modelos com 6 a 12 pessoas; Geram programas a serem executados em grandes computadores.	FORTRAN, PASCAL E C.
70 e início dos anos 80	Linguagens de simulação	Utilização em uma maior numero de corporações; Desenvolvimento e uso de pacotes de linguagens; Surgem linguagens de simulação baseadas em <i>System Dynamics</i> ; Comandos projetados para tratar lógica de filas e demais fenômenos comuns; Mais amigáveis, mas ainda requerem programador especializado.	SIMSCRIPT, GPSS, GASP IV, DYNAMO, SIMAN E SLAM
80 e início dos anos 90	Linguagens de simulação	Introdução do PC e da animação; Presença de guias, menus e caixas de diálogos; Simulação realizada antes do início da produção; Facilidade de uso' Menos flexível que as linguagens de propósito geral e de simulação; Projetados para permitir modelagem rápida; Dispõe de elementos específicos para representar filas, transportadores e etc.; Restringem-se a sistemas de certos tipos.	Simfactory e Xcell
Após 90	Pacotes flexíveis de programas de simulação	Melhor animação e facilidade de uso; Fácil integração com outras linguagens de programação; Usada na fase de projeto; Grande uso em serviços; Uso para controle de sistemas reais; Grande integração com outros pacotes (base de dados e processadores de texto) Aprimoramento dos simuladores, o que permite modelagem rápida; Integram a flexibilidade das linguagens de simulação, com a facilidade de uso dos pacotes de simulação.	Witness, Extend, tella, ProModel for Windows

Fonte: Adaptado de GAVIRA (2003)

Essa evolução a partir da década de 60 se deu em função da necessidade de desenvolvimento da ferramenta como conseqüência de uma demanda maior em termos de resultados mais confiáveis e de uma maior visualização de seus benefícios por parte dos usuários. O surgimento de softwares específicos acoplados a simuladores capazes de reproduzir graficamente inúmeros sistemas

ilustra essa idéia. Dentre as vantagens da utilização dos softwares pode-se citar a facilidade de compreensão, o treinamento de pessoal e uma melhor visualização do sistema produtivo das indústrias (SILVA, 2005).

Mesmo após o reconhecimento relativo a eficiência da simulação computacional, seu uso ainda demandava muito tempo de treinamento, caso os candidatos usuários não tivessem um conhecimento prévio. A construção dos modelos e animações era demorada, e os analistas precisavam ter conhecimento do sistema que estavam simulando. A partir daí, tornou-se necessário que os próprios usuários dos modelos de simulação fossem os analistas e implementadores do modelo computacional. Surgiu então uma nova tecnologia, conhecida como VIS - *Visual Interactive Simulation* (GAVIRA, 2003). Esta nova tecnologia, base dos atuais softwares de simulação, realiza a modelagem através de ícones agrupando comandos das linguagens tradicionais de simulação, e tornando o trabalho de desenvolvimento mais fácil, através de uma interface semelhante à do *Windows*, mas muito mais amigável. Reduzindo drasticamente o tempo de treinamento.

Embora as linguagens de programação utilizadas até então ainda poderem ser utilizadas, isso raramente ocorre. Existem pacotes de simulação disponíveis no mercado atualmente que apresentam inúmeras vantagens em termos de facilidade de uso, eficiência e eficácia dos resultados obtidos. Dentre as vantagens em se usar os pacotes de simulação podemos ressaltar (SHANNON, 1998) a redução na tarefa de programar, o aumento da flexibilidade de realizar mudanças nos modelos, menos erros de programação e a coleta automática de dados estatísticos;

O objetivo dos pacotes de simulação é diminuir o espaço entre a conceituação que o usuário tem do modelo e sua forma executável. Os pacotes de simulação são divididos em duas categorias: uma de propósito geral e outra de propósito específico. Na primeira categoria estão os pacotes que podem resolver praticamente todos os problemas de simulação de eventos discretos, como ARENA®, AweSim®, GPSS/H™, Simscript II.5®, Extend™ etc. Na segunda categoria estão os pacotes utilizados na simulação de sistemas de manufatura e problemas de manuseio de material, tais como SimFactory, ProModel®, AutoMod™, Taylor II® e Witness® se encaixam nesta categoria. Assim como os pacotes designados para a condução de estudos de reengenharia de processos, como BPSimulator™, processModel™, SIMPROCESS® e Extend+BPR. Outros,

possuem o objetivo de simular áreas como a da saúde, como o MedModel®, ou redes de comunicação, como o COMNET II.5 (SHANNON, 1998).

Este trabalho utiliza o ProModel® para realizar a modelagem de um sistema de manufatura puxado. A razão de seu uso esta no fato de ser o software disponível na Universidade Federal de Itajubá para a realização de estudos de simulação.

3.3. Vantagens e desvantagens

Os modelos de simulação apresentam inúmeras vantagens em relação aos modelos matemáticos e analíticos. Sua conceituação de fácil compreensão torna fácil sua justificativa seja para a gerência da organização ou para os clientes. Além disso, possui mais créditos, já que é capaz de comparar o modelo virtual com a situação real e assim proporcionar grandes contribuições para o objeto de estudo.

STRACK (1984) em seu livro recomenda a utilização da simulação quando:

1. Não há uma formulação matemática completa para o problema;
2. Não há método analítico para a resolução do modelo matemático;
3. Resultados são mais fáceis de serem obtidos por simulação que por qualquer outro analítico;
4. Não existe habilidade técnica para a resolução do modelo matemático por técnica analítica ou numérica;
5. Torna-se necessário observar o desenvolvimento do processo desde o início até o seu término;
6. Quando são necessários detalhes específicos do sistema;
7. Quando a experimentação na situação real apresenta inúmeros obstáculos ou não é possível.

SHANNON (1998) ressalta as seguintes vantagens de utilização da simulação:

1. Possibilidade de se testar novos *designs* e *layouts* sem a implementação real dos recursos necessários;
2. Pode ser utilizada para explorar novas políticas de alocação de funcionários, procedimentos operacionais, tomadas de decisão, estruturas organizacionais, fluxos de informação e etc. Sem causar nenhuma ruptura no sistema real da organização;
3. A simulação permite a identificação de gargalos nos fluxos de informação, material e produto e realiza testes com o objetivo de aumentar cada taxa;
4. Permite a realização de testes de hipótese em relação a como e por que certos fenômenos ocorrem no sistema;
5. A simulação permite o controle do tempo. Assim, sistemas podem ser simulados por meses ou anos em questão de segundos e fornecer resultados visualizados em longo prazo. Além disso, pode-se diminuir a velocidade da simulação para a realização de estudos;
6. Permite adquirir conhecimento em relação a como o sistema funciona e a identificação de quais são as variáveis que mais afetam a performance do modelo;
7. A simulação significa em uma maior possibilidade de realização de experimentos com situações não familiares e responder a questões “e se”.

Apesar das inúmeras vantagens de sua utilização, a simulação apresenta algumas desvantagens, que são as que se seguem (LAW e KELTON, 2000):

1. Os modelos de simulação são caros e consomem tempo para serem desenvolvidos;
2. A execução de um modelo de simulação estocástico só estima as características do mesmo para valores específicos dos parâmetros de entrada. Assim, serão necessárias várias execuções independentes do modelo para os conjuntos de parâmetros a serem estudados. Por essa razão é que os modelos de simulação geralmente não são tão eficientes em relação a otimização.
3. A grande quantidade dos 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 dado sistema, não importa o tipo de informação que será obtida, a real utilidade será mínima.

Além destas desvantagens, SHANNON (1998) identifica:

1. A modelagem de um dado sistema é uma “arte” que requer um treinamento especializado e as habilidades dos modeladores tendem a variar amplamente. A utilidade do estudo dependerá diretamente da qualidade do modelo desenvolvido e das habilidades do modelador;
2. A coleta de dados de entrada confiáveis pode consumir grande quantidade de tempo e mesmo assim os resultados podem ser questionáveis. A simulação não pode compensar dados inadequados;
3. Modelos de simulação envolvem a execução de dados de entrada, produzindo as saídas propriamente ditas, surgidas em função das rodadas realizadas. Os modelos não geram por si só uma solução ótima, servindo apenas como uma ferramenta de análise a partir de condições pré-estabelecidas pelo modelador.

3.4. Definições

Para um entendimento satisfatório sobre simulação, é preciso conhecer as definições de sistema e modelo.

3.4.1. Sistemas

Segundo SEILA (1995), um sistema é um conjunto de componentes ou entidades interagindo entre si, trabalham juntos para atingir algum objetivo.

Para LAW e KELTON (2000), um sistema pode ser definido como uma coleção de entidades, pessoas ou máquinas, que agem e interagem juntas através de suas habilidades com algum objetivo específico. Segundo os mesmos autores, um estado do sistema pode ser definido como o conjunto de variáveis necessárias para descrever um sistema em um intervalo de tempo.

Os sistemas podem ser classificados em dois tipos: discretos e contínuos. Um sistema discreto apresenta mudanças em um dado intervalo de tempo, por exemplo, peças que chegam a uma

máquina. Em um sistema contínuo as variáveis mudam continuamente de acordo com o tempo, como por exemplo, os quilômetros rodados pelos caminhões na simulação de um sistema logístico (HARREL *et al*, 2000).

São exemplos de sistema: sistema de tráfego, sistema policial, sistema econômico, sistema bancário, entre outros. Um sistema também pode ser uma parte ou conjunto no qual o estudo será realizado, que por sua vez está inserido em universo ainda maior, como a área responsável pelo saque e depósito de uma agência bancária (SILVA, 2005).

3.4.2. Modelos

O homem, ao estudar sistemas, objetos ou fenômenos, muitas vezes depara-se com dificuldades de analisá-los na sua forma natural de existência, por dificuldade de acesso, medição ou mesmo altos riscos e custos envolvidos. Por isso são utilizadas formas de representação que permitam manipular e compreender as entidades estudadas, tanto em seus aspectos qualitativos como nos quantitativos. Esta representação é feita por meio de modelos (STRACK, 1984).

Em casos de otimização, o uso de modelos traz muitos benefícios, e em muitos caso, é inevitável, quando o sistema real não existe, porque trata-se de projeto, ou porque não está disponível para experimentos, devido a diversos motivos, como riscos de grandes prejuízos ou de vidas. A essência da otimização concentra-se na construção e uso de modelos (STAMM, 1998).

Um modelo pode ser definido como uma representação de um objeto, sistema, ou idéia em alguma outra forma que não a da entidade em si. Em um sentido amplo, um modelo é uma certa quantidade de informações e atributos sobre aquilo que é representado, conforme os objetivos e necessidades da análise. Pode variar desde uma representação simplificada, croqui, descrição, equação matemática ou até uma réplica acrescida de sensores para medida e experimentação. Dados, parâmetros, relações e vínculos devem ter representação adequada ao problema sob investigação (STRACK, 1984).

Para LAW e KELTON (2000), um modelo pode ser definido como uma representação de um sistema desenvolvido com objetivo de estudo. O modelo pode ser suficientemente detalhado ou “válido” para permitir ao analista ou tomador de decisão a sua utilização para tomar as mesmas decisões que estes tomam utilizando o sistema real.

Para BRATLEY, BENNET e SCHRAGE (1987), um modelo é uma descrição de algum sistema com o objetivo de se prever o que acontece quando determinada atitude é tomada. Para isso o mesmo deve ser válido. No entanto, as fronteiras entre o sistema e o modelo devem estar bem definidas, já que muitas forças que afetam o sistema precisam ser omitidas no modelo para que o mesmo continue sendo tratável, mesmo quando não há provas de que tais fatores realmente afetam o sistema. Sendo assim, inevitavelmente, o modelo acaba sendo mais bem definido do que o sistema. Para um modelo ser útil, é essencial que após sua realização, qualquer comportamento relevante ou propriedade seja determinada de maneira prática. Seja analiticamente, numericamente ou manipulando o modelo com dados de entrada (aleatórios ou não), que sejam capazes de produzir determinadas saídas. Este último processo é definido como simulação.

Os modelos podem ser classificados de diferentes maneiras. Alguns são concretos, têm representação física, outros são abstratos, já que são formulados por meio de símbolos, dados ou descrições. STRACK (1984) classifica os modelos em quatro tipos, que são os que se seguem:

1. Modelos icônicos ou físicos: Enquadram-se neste tipo os modelos que são representados por atributos físicos semelhantes aos sistemas em estudo. Podem ser bi ou tridimensionais e são utilizados para demonstração ou experimentação indireta. A este grupo pertencem os protótipos, modelos pilotos e modelos em escala. Os primeiros podem ser descritos como uma cópia do sistema real, contendo quase todos seus níveis de detalhes e atributos. São utilizados para minimizar os riscos e incertezas, associados com a realização de experiências em sistemas já definidos. Os modelos pilotos representam uma versão do sistema ou processo em questão contendo os atributos essenciais predominantes na entidade modelada. São normalmente utilizados em laboratórios ou locais bem determinados, para produzirem as mesmas operações e resultados do sistema modelado, mas não em escala industrial. Os modelos em escala dizem respeito às reduções das dimensões do modelo em relação ao sistema real, como túneis, estradas entre outros.
2. Modelos analógicos: São aqueles nos quais as propriedades em estudo são representadas por conjuntos de dispositivos análogos, ou seja, que se comportam de maneira similar. Os estudos são realizados com um tipo de variável e os resultados transferidos para outras variáveis. A substituição de variáveis pode acontecer em dois casos. O primeiro se refere à semelhança de variáveis, como no caso em que os fluxos de água representam corrente elétrica. No segundo

caso, o segundo caso é o de computador eletrônico analógico que simula com tensão elétrica outros tipos de variáveis, soluciona equações diferenciais e permite o acoplamento do computador simulador aos sistemas em estudo. Gráficos são tipos de modelos analógicos. Nestes a distância pode representar o valor das variáveis modeladas, a forma das linhas mostrar as inter-relações e as dimensões podem representar custos, tempo, produção entre outros.

3. Modelos matemáticos: São aqueles que fazem uma abstração da realidade utilizando conceitos complexos envolvendo linguagens formais, sentenças e expressões cujas sintaxes e semânticas matemáticas guardam uma semelhança básica com o conceito de modelos para simulação, que é a relação de satisfação, ou seja, a condição de semelhança entre a estrutura e a teoria (STAMM, 1998); Para STRACK (1984), são os modelos onde são buscadas interpretações matemáticas para os atributos físicos, constituindo o tipo mais abstrato, mais geral e com grande uso em estudos de sistemas. O mesmo autor os divide em dois grupos: analíticos e numéricos. Os primeiros são aqueles que resultam em uma expressão matemática bem definida entre as variáveis dependentes e independentes. E a solução é alcançada através da resolução de tal expressão. Nos modelos numéricos não é necessário o conhecimento de equações numéricas que regem o sistema. A solução é obtida por interações, métodos de convergência, interpolação, sendo fornecido um valor estimado juntamente com o erro do método.
4. Modelos de simulação: Quando as relações que compõem o modelo são relativamente simples, é possível empregar métodos matemáticos como a álgebra, o cálculo ou a teoria da probabilidade para obter informações sobre questões de interesse (LAW e KELTON, 2000). Neste tipo de modelo, as representações são feitas por meio de procedimentos lógicos ou matemáticos e para obtenção de resultados, eles são “executados” ao invés de serem “resolvidos”. A simulação não é uma teoria, mas uma metodologia de resolução de problemas. Nesse sentido, trata-se de um método de análise que não gera soluções por si só, como os modelos analíticos, mas servem como técnica ou ferramenta para “atuar da mesma maneira” que o sistema estudado, de onde são obtidos dados estatísticos de desempenho para análise (STRACK, 1984).

Este trabalho irá utilizar o último tipo de modelo, que pode ser construído manualmente, ou com auxílio de programas de computador específicos para simulação. A simulação propriamente dita e suas peculiaridades serão tratados no tópico a seguir.

3.4.3. Simulação

A simulação computacional pode ser definida como a representação virtual de um sistema da vida real através de um modelo, tornando possível o estudo do sistema sem que seja necessário construí-lo na realidade, ou mesmo fazer modificações nesse sistema, e estudar os resultados dessas modificações, sem que haja necessidade de se alterá-lo previamente (HARREL *et al*, 2000).

Para SHANNON (1998), a simulação computacional é uma poderosa ferramenta na análise de processos e sistemas complexos. Tornando possível o estudo, análise e avaliação de situações que não seriam possíveis na vida real. Em um mundo em crescente competição, tem se tornado uma metodologia indispensável de resolução de problemas para os tomadores de decisão nas mais diversas áreas. JOHANSSON (2002) reforça esta diversidade de áreas de aplicação mostrando como a simulação tem sido utilizada desde a representação de operações militares, operações logísticas, linhas de manufatura até operações na área de saúde.

Para LAW e KELTON (2000), simulação computacional é a representação de um sistema real através de um computador para a posterior realização de experimentos para avaliação e melhoria de seu desempenho. Significando a importação da realidade para um ambiente controlado onde se pode estudar seu comportamento sob diversas condições, sem os riscos físicos e/ou custos envolvidos em um estudo convencional.

Para BANKS (1999), a simulação é uma metodologia indispensável de solução para inúmeros problemas da vida real. A simulação é utilizada para descrever e analisar o comportamento dos sistemas através de questões do tipo “o que aconteceria se” em relação a situação real. Tanto sistemas existentes como conceituais podem ser modelados através da simulação.

Todos os autores citados acima concordam sobre a possibilidade de aplicação da simulação nas mais diferentes áreas, desde a área de saúde, até a militar. No entanto, neste trabalho a linha de

estudo é a manufatura onde possibilidade de realização de estudos de simulação torna-se muito interessante.

Alguns exemplos de estudos que podem ser realizados na manufatura através da simulação são a análise de necessidade de material e pessoal, Avaliação de desempenho de um processo ou fluxo produtivo, avaliação dos procedimentos operacionais, como programação da produção, revisão das políticas de inventário e políticas de controle da qualidade (LAW e MACCOMAS, 1999).

Alguns exemplos de medidas de desempenho comuns na simulação na manufatura são: entrada e saída de produtos; o tempo das peças no sistema; o tempo das peças em espera/fila; o tamanho das filas; tempo de entrega; utilização dos equipamentos e pessoas;

A maneira como a simulação do sistema trabalha (com o uso de distribuições estatísticas, considerando-se o tempo ou não, ou com variáveis discretas ou não, etc) 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 e terminante ou não terminante.

3.4.3.1. Simulação estática e dinâmica

Para LAW e KELTON (2000), simulação estática é a representação de um sistema em um dado momento. Um exemplo é a simulação de Monte Carlo, utilizada desde a área financeira até na de estoques. A simulação dinâmica é 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 sofrem influência do tempo. Um exemplo é a simulação das atividades ocorridas em um banco ao longo de suas oito horas de funcionamento diárias.

3.4.3.2. Simulação determinística ou estocástica

Para PEREIRA (2000), Os modelos de simulação são determinísticos quando as variáveis de entrada assumem valores exatos, assim, os resultados (saídas) desse tipo de simulação serão sempre os iguais independentemente do número de replicações. Os modelos estocásticos permitem que as variáveis de entrada assumam diversos 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.

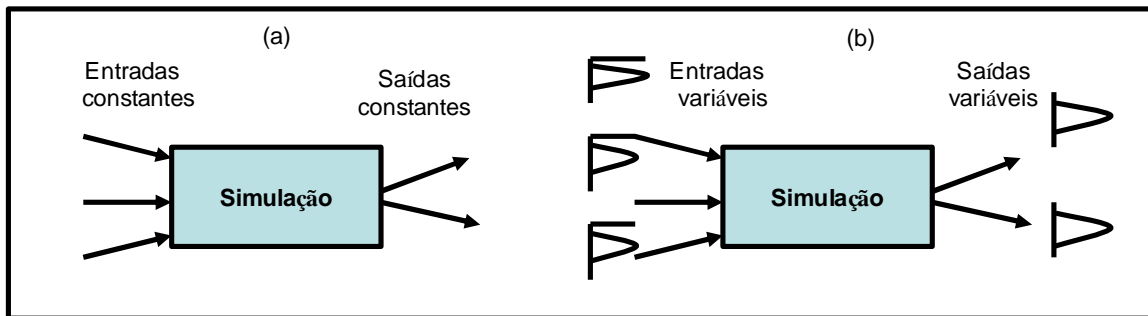


Figura 3.1: Simulação determinística X estocástica
Fonte: DUARTE (2003)

3.4.3.3. Simulação discreta e contínua

Para STRACK (1984), a simulação contínua modela sistemas em que suas variáveis mudam continuamente de valor. É utilizada em estudos que consideram os sistemas constituídos por um fluxo contínuo de informações ou itens. A simulação discreta caracteriza-se por eventos onde as mudanças ocorrem de maneira descontínua, ou seja, sofrem mudanças bruscas. A **Figura 3.2** ilustra essa diferença entre os tipos de simulação discreta e contínua.

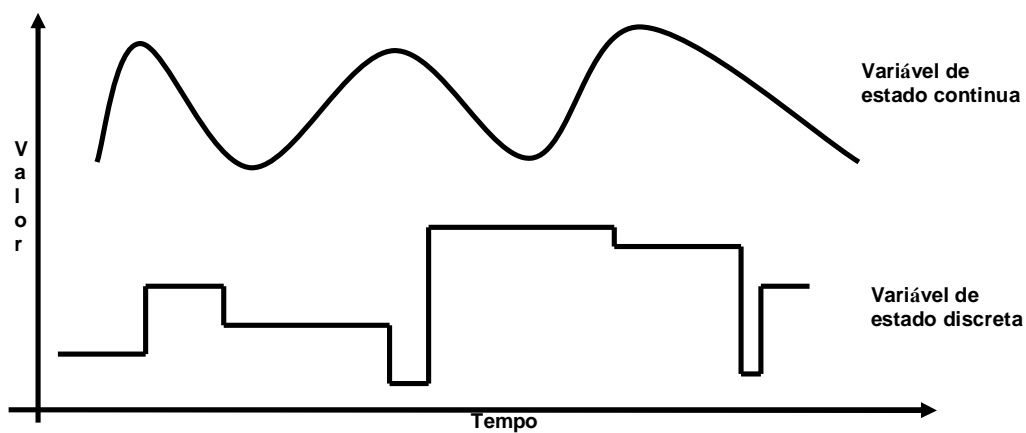


Figura 3.2: Simulação discreta X contínua

Fonte: DUARTE (2003)

3.4.3.4. Simulação terminante e não terminante

Segundo LAW e KELTON (2000), a simulação é *terminante* quando o objetivo é estudar um sistema num dado intervalo de tempo, ou seja, conhecer seu comportamento ao longo deste intervalo, sendo definidas as datas de início e término da simulação. Exemplos são simulações realizadas em postos de cobrança de pedágio onde se deseja definir o número de postos em funcionamento de acordo com a hora do dia; assim, o intervalo de interesse está entre zero e vinte e quatro horas.

A simulação pode ser classificada como *não terminante* quando o objetivo é estudar o sistema a partir do momento em que este atinge um estado estável, alcançado após um período de aquecimento, onde se determina e elimina as tendências iniciais. Presumindo que a simulação poderia continuar infinitamente sem nenhuma mudança estatística no comportamento. Assim, uma simulação do comportamento das pás de uma turbina pode ser considerada *não terminante*, desde que o interesse seja estudar as características de seu escoamento em condições estáveis, após um período de aquecimento.

Na **Tabela 3.2**, PEREIRA (2000) mostra de maneira sucinta a classificação de sistemas e modelos para simulação, além da classificação da própria simulação.

Tabela 3.2: Classificação de Sistema, Modelo e Simulação

SISTEMA	MODELO		SIMULAÇÃO
DISCRETO: Variáveis envolvidas assumem valores finitos ou infinitos numeráveis.	DETERMINÍSTICO: Variáveis assumem valores determinados.	ESTÁTICO: Estuda o sistema sem levar em conta sua variabilidade com o tempo.	TERMINANTE: Há interesse em se estudar o sistema num dado intervalo de tempo.
CONTÍNUO: Variáveis mudam constantemente com o tempo.	ESTOCÁSTICO: Variáveis assumem valores diversos segundo uma determinada distribuição de probabilidades.	DINÂMICO: Representa o sistema a qualquer tempo.	NÃO TERMINANTE: Há o interesse em estudar o sistema a partir de um determinado estado estável, podendo o estudo, prolongar-se indefinidamente.

Fonte: PEREIRA (2000)

3.5. Metodologias de simulação

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

Segundo STRACK (1984), os resultados insatisfatórios do uso da ferramenta ocorrem em função da não valorização 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 simulações mal sucedidos tem como causa a ausência de um planejamento condizente com a importância de seu estudo. Por isso, simular requer mais do que o conhecimento de um *software* específico, mas também, pessoas com conhecimento dos passos a serem seguidos, bem como experiência analítica, estatística e organizacional (SILVA, 2005).

A metodologia nos trabalhos de simulação busca sistematizar estes passos de seu desenvolvimento, otimizando a integração entre *software*, modelador e usuário, e evitando desperdício de tempo, dinheiro e resultados frustrantes.

O tipo de simulação a ser abordada neste trabalho será a simulação de eventos discretos, onde as variáveis envolvidas assumem valores finitos e infinitos numeráveis.

Entretanto, a construção de um modelo computacional possa trazer resultados satisfatórios deve passar pelas seguintes etapas (SHANNON, 1998, CARLSON, 2003, LAW e KELTON, 2000, PEREIRA, 2000, DUARTE, 2003 e SILVA, 2005).

1. Formulação do problema: Todos os estudos em simulação se iniciam com a descrição do problema. Tal descrição é feita pelas pessoas que presenciam esta situação (os clientes) e o analista/modelador precisa ter muito cuidado com o entendimento do problema. Para isso sugere-se que este formule uma série de hipóteses ou alternativas e as exponha ao cliente, em busca de sua concordância. Mesmo com esses cuidados, as chances do problema ter que ser reformulado assim como os objetivos de estudo, são muito grandes.

2. Definição dos objetivos e planejamento do projeto: Ou estabelecer um propósito. Este passo deve envolver um acompanhamento do analista/modelador e do cliente como consultores internos e externos. Is objetivos indicam a questão que deve ser respondida com o estudo. O planejamento deve envolver uma descrição dos cenários que devem ser investigados. Além disso, deve envolver o tempo necessário, pessoas, recursos computacionais. Assim como os estágios do estudo, resultados de cada um e seus custos.

3. Conceitualização do modelo: O modelo real sob investigação é resumido através do modelo conceitual, que nada mais é do que uma série de relacionamentos matemáticos e lógicos relativos aos componentes e estrutura do sistema.

4. Coleta de dados: Depois da proposta ter sido aceita, uma lista com os dados necessários deve ser entregue ao cliente. Geralmente os clientes possuem esses dados para disponibilizar. No entanto, muitos dados depois de fornecidos podem ser muito diferentes do previsto. Por exemplo, durante a simulação do sistema de reservas de passagens aéreas de uma dada empresa foi afirmado que a empresa possuía cada dado referente as chamadas dos últimos cinco anos. Entretanto, durante a realização do estudo verificou que tais dados eram as médias. São os valores individuais é que devem ser usados e não os valores médios.

5. Construção do modelo: A construção do modelo deve iniciar de forma simples e a complexidade ocorra de maneira evolutiva. Isso pode ser feito adicionando-se detalhes ao modelo de maneira gradativa. Um exemplo é a simulação de uma linha produtiva qualquer, o modelo básico com as chegadas, filas e maquinas é construído. Depois adiciona-se os carregamentos e as peças e logo em seguida, algumas características especiais. A construção de um modelo de grande complexidade significa dispêndio de tempo e dinheiro. O envolvimento do cliente é essencial para a qualidade e desenvolvimento do modelo.

6. Execução do modelo: Nesta etapa o modelo conceitual é executado em um microcomputador através de um software.

7. Verificação: Nesta etapa, o modelador deverá verificar a consistência dos dados coletados em relação ao modelo considerado, já que de nada adianta um modelo que não reflita a realidade do sistema.

8. Validação: A validação é a certeza de que o modelo construído reflete o funcionamento do sistema real. Uma das maneiras é chamar o sistema real de sistema base e comparar seus resultados com os do modelo. Se elas forem similares, pode-se dizer então que o modelo é válido. Outra forma de validar o modelo é consultar os operadores do sistema sob a validade do mesmo. Existem muitos métodos de validação de modelos.

9. Planejamento dos experimentos: Para cada execução da simulação e sua posterior análise, decisões precisam ser tomadas em relação ao tempo de duração da simulação e o número de replicações para cada cenário;

10. Realização e análise dos experimentos: As simulações e suas posteriores análises são realizadas com o objetivo de estabelecer medidas de desempenho para próximos cenários a serem simulados.

11. Replicações extras: Com base nas análises realizadas o modelador determina se outras replicações ou execuções serão necessárias assim como a utilização ou não de novos cenários.

12. Documentação e relato: A documentação é necessária por inúmeras razões. Se um modelo precisar ser utilizado novamente para a mesma análise ou outra, pode ser necessário lembrar como o modelo funciona. Isso dará mais confiabilidade ao modelo. Por um outro lado, se o modelo precisar ser modificado, através da documentação, essa tarefa se tornará menos árdua. Os resultados devem ser relatados de maneira clara e consciente. Permitindo ao cliente revisar a formulação final, as alternativas criadas, seus critérios de criação e acima de tudo, as recomendações do modelador/analista.

13. Implementação: O analista/modelador atua mais como um relator do que como um defensor do modelo final. Já que o relatório construído item anterior irá ajudar o cliente em sua tomada de decisão. Se o cliente esteve envolvido durante todas as etapas e estas realizadas conforme o recomendado, a probabilidade de sucesso na implementação é muito grande.

LAW (2003) ilustra tais etapas de maneira mais resumida agrupando alguns passos, conforme exposto na **Figura 3.3:**

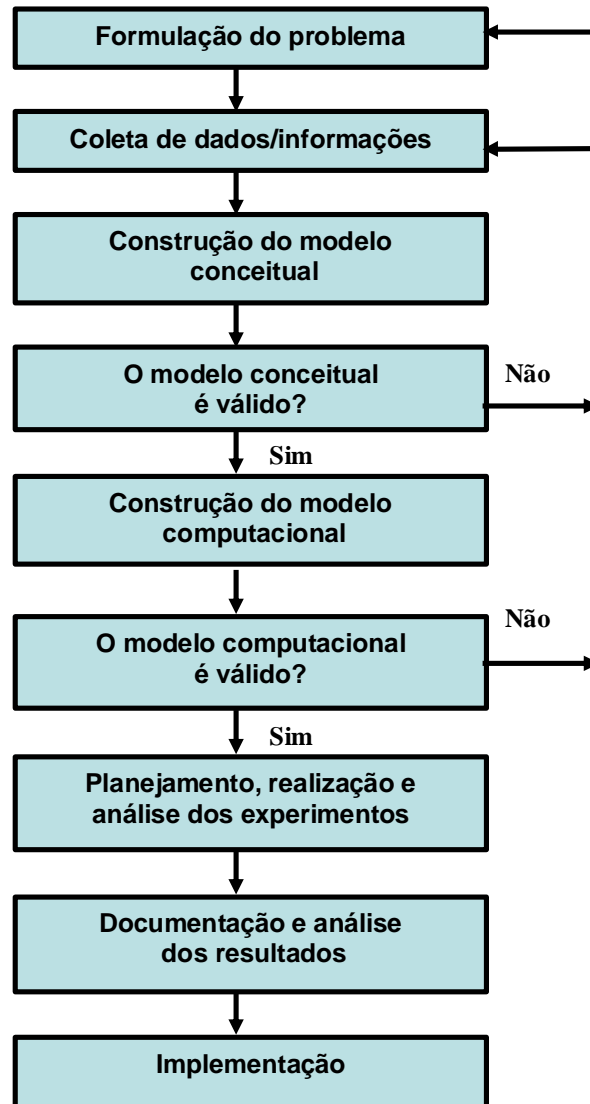


Figura 3.3: Metodologia de condução de um estudo de simulação
Fonte: Adaptado de LAW (2003)

A visualização das etapas e da complexidade de se criar uma representação virtual de um sistema real implicam em grandes responsabilidades para o profissional desta área. Resultados podem ser alcançados de maneira muito proveitosa, desde é claro que se siga a metodologia recomendada.

Inúmeras simulações mal sucedidas têm como causa a deficiência de um planejamento eficiente de seus procedimentos de estudo. Simular requer mais do que o conhecimento de um *software*

específico, mas também, pessoas com conhecimento da metodologia a ser seguida, bem como experiência analítica, estatística, organizacional e de engenharia (SILVA, 2005).

Segundo SILVA (2005), a modelagem computacional de um sistema é uma tarefa que exige muito esforço por parte do modelador e que, se conduzida com raciocínio cuidadoso e planejado, poderá trazer benefícios muito proveitosos. Para que isto ocorra, o autor propõe cinco princípios básicos e indispensáveis em qualquer metodologia para a implementação da simulação:

1. O modelo deve ser simples apesar de partir de pensamentos complicados. Os modelos não necessitam ser tão complicados quanto a realidade;
2. Ser parcimonioso começando do simples e acrescentar complexidade na medida do necessário;
3. Evitar grandes modelos pela dificuldade em entendê-los. A regra é dividir esses grandes modelos;
4. A definição dos dados a serem coletados deve ser orientada pelo modelo. O modelador não deve “se apaixonar pelos dados” como descreve o autor;
5. O comportamento do modelador na construção do modelo seria como este estivesse desembaraçando-se dos problemas pois a modelagem de alguma forma é um processo desordenado.

3.6. Verificação e validação de um modelo computacional

Para HARREL *et al* (2000) verificar o modelo é realizar um trabalho de depuração da programação procurando dois tipos de erros: erros de sintaxe e erros de semântica. As principais técnicas de verificação são:

1. Revisão da codificação do modelo;
2. Verificação dos resultados do modelo;

3. Verificação da animação e sua coerência com o modelo real;
4. Utilização dos recursos de detecção de erros do pacote de software.

Validação, para o mesmo autor, é o processo onde se determina a relação entre o modelo e a realidade que o mesmo representa. É de extrema importância, uma vez que todas as decisões sobre o que fazer no sistema real serão baseadas nos resultados que o modelo produz. Para HARREL *et al* (2000) o processo de validação não é trivial e o modelador somente poderá atestar a validade do modelo baseado em evidências.

Segundo SARGENT (2004), existem diversas técnicas para validar um modelo, desde uma simples visualização até métodos estatísticos de alta complexidade, sendo comum a comunicação de mais de duas técnicas. As técnicas mais comuns de validação são:

1. Observação da animação do modelo para atestar se aspecto visual e o funcionamento do mesmo condizem com o sistema real;
2. Comparação com outros modelos já validados, realizando-se a simulação de entradas que já possuem saídas predefinidas para a avaliação dos resultados;
3. Teste de degeneração e condições extremas do sistema, permitindo-se observar se o modelo construído possui as mesmas características que o sistema real, como por exemplo o aumento de peças em fila em uma determinada máquina durante o período de funcionamento do sistema.
4. Validação por aparência, onde pessoas que dominam o sistema são convidadas a opinar sobre sua aparência final;
5. Testes com dados históricos do sistema real, utilizados na construção do modelo e na visualização dos resultados já alcançados pelo sistema real no sistema modelado.
6. Análise da sensibilidade de resposta e alterações nos dados de entrada e posterior comparação com o sistema real;

7. Condução de *turing tests*, onde os gestores do sistema modelado expressam sua opinião sobre a consistência do modelo computacional em relação ao sistema real.

Segundo HARREL *et al* (2000), o tipo de validação mais utilizado é a comparação entre os resultados do modelo com os resultados do sistema real. Segundo o mesmo autor, o propósito da validação é diminuir os riscos associados às decisões tomadas com base no modelo, no entanto, deve existir um equilíbrio entre o custo, o esforço e o risco envolvido. A **Figura 3.4** ilustra este equilíbrio, evidenciando que quanto menor o risco de existirem erros ou baixa precisão do modelo, maiores serão os custos da validação do sistema, vindo inclusive a inviabilizar a execução da simulação devido ao a relação custo total do projeto x a verba destinada ao mesmo.

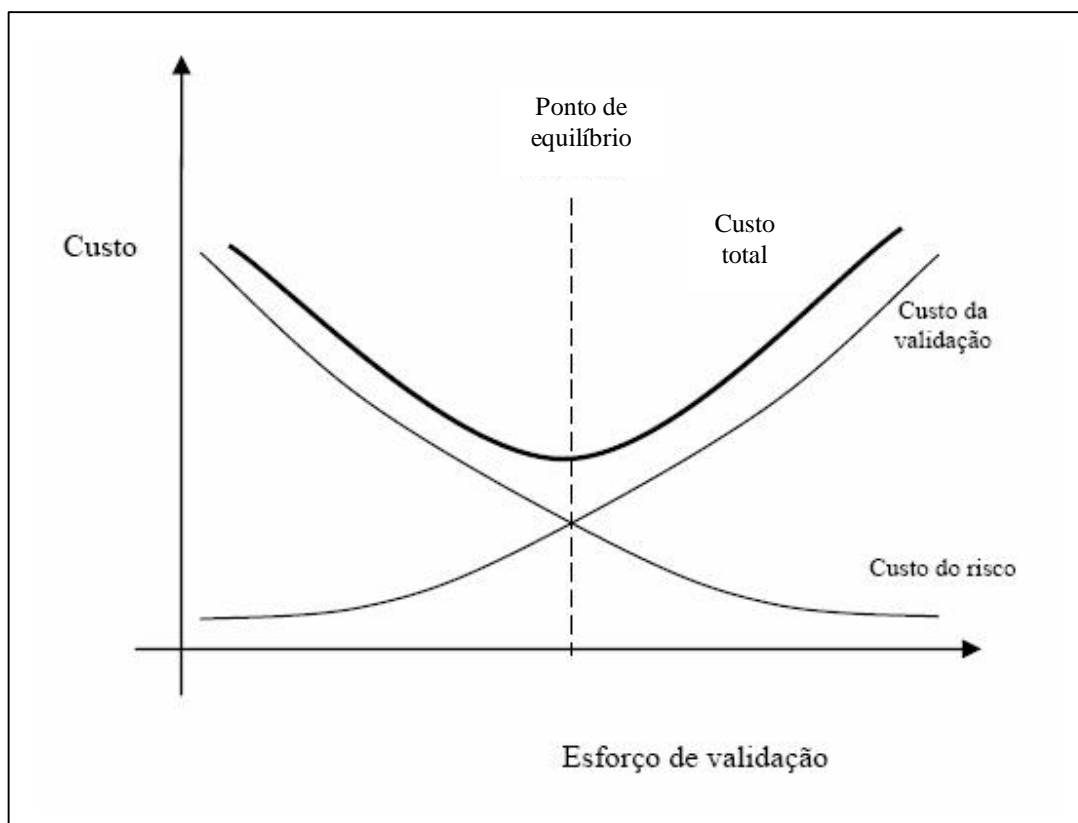


Figura 3.4: Ponto de equilíbrio entre esforço e custo de validação.
Fonte: HARREL *et al* (2000).

3.6. A simulação computacional em ambientes de manufatura

Segundo HARREL *et al.* (2000) e LAW e KELTON (2000), simulação é a imitação de um sistema real, modelado em computador, para avaliação e melhoria de seu desempenho. Ou seja, simulação é a importação da realidade para um ambiente controlado onde se pode estudar o comportamento do mesmo, sob diversas condições, sem riscos físicos e/ou grandes custos envolvidos. BANKS (2000) afirma que a simulação envolve a criação de uma história artificial da realidade e, com base nesta história artificial, são realizadas observações e inferências nas características de operação do sistema real representado. A **Figura 3.5** esquematiza este conceito da transformação da realidade em modelo e novamente dos resultados em realidade.

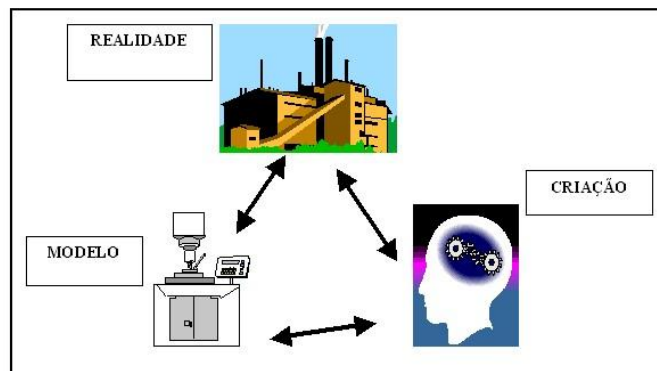


Figura 3.5 – Realidade x Modelo
Fonte: DUARTE (2003)

A simulação não é uma ferramenta mágica que substitui o trabalho de interpretação humano, mas sim uma ferramenta, capaz de fornecer resultados para uma análise mais elaborada a respeito da dinâmica do sistema. Desta maneira, a simulação permite uma interpretação mais profunda e abrangente do sistema estudado (DUARTE, 2003).

O’KANE *et al.* (2000) afirmam que a simulação tem se tornado uma das técnicas mais populares para se analisar problemas complexos em ambientes da manufatura. Para TAVEIRA (1997), a simulação é uma ferramenta muito útil, uma vez que prevê o comportamento de sistemas complexos, calculando os movimentos e interações dos componentes deste sistema.

Segundo BANKS *et al.* (2005), a simulação é uma das ferramentas mais amplamente utilizada em sistemas de manufatura do que em qualquer outra área. Algumas razões são:

1. O aumento da produtividade e qualidade na indústria é um resultado direto da automação. Como os sistemas de automação são cada vez mais complexos, estes só podem ser analisados pela simulação;
2. Os custos de equipamentos e instalações são relativamente altos;
3. Os custos dos computadores estão cada vez mais baixos e os processadores mais rápidos;
4. Melhorias nos softwares de simulação reduziram o tempo de desenvolvimento de modelos.
5. A disponibilidade de animação resultou em maior compreensão e utilização dos gestores da manufatura.

3.7. Considerações finais

Através da descrição e explicação do histórico e dos conceitos relacionados com simulação, suas vantagens e desvantagens, assim como sua metodologia de implementação pode-se verificar que se trata de uma ferramenta que se mostra adequada às mais diferentes áreas de aplicação, em especial em ambientes de manufatura. Entretanto nota-se que o processo de validação deve ser conduzido com cuidado uma vez que quanto maior a confiabilidade na representação virtual do sistema, maior a confiança nos resultados a serem alcançados e nas experimentações a serem realizadas.

A maior dificuldade na aplicação da ferramenta está na obtenção de modelos que possam ser considerados válidos e aptos a serem objeto de experimentações que possam trazer resultados pertinentes e aplicáveis na prática.

Capítulo 4

OTIMIZAÇÃO

4.1. Considerações iniciais

Este capítulo aborda a otimização a partir de sua definição geral fazendo uma breve descrição sobre seu histórico. Dentre os métodos de otimização existentes serão abordados os algoritmos evolutivos, e de maneira mais detalhada os algoritmos genéticos, uma vez que o *software* de otimização utilizado na fase de aplicação deste trabalho se baseia neste método.

4.2. Definições

Otimizar é ato de obter os melhores resultados em dadas circunstâncias (RAO, 1979). Para FLETCHER (1980), otimização é a ciência que objetiva determinar as melhores soluções para algum problema definido matematicamente, podendo ser geralmente, uma representação de um modelo real.

Segundo HARREL *et al.* (2000), a otimização é o processo de tentar diferentes combinações de valores para variáveis que podem ser controladas (variáveis independentes), buscando uma combinação de valores que provê a saída mais desejada. Na maioria das vezes este processo de tentar diferentes combinações para as variáveis se torna difícil ou mesmo impossível de serem feitas em um sistema real, e por isso é feito através de modelos.

Técnicas de otimização são conhecidas há mais de um século e podem ser aplicadas em diversos campos da ciência. No entanto, estas técnicas apresentam algumas limitações, como: falta de continuidade das funções a serem otimizadas ou de suas restrições, funções não convexas, multimodalidade (vários pontos ótimos), existência de ruídos nas funções, necessidade de se trabalhar com valores discretos para as variáveis, entra outros. (SILVA, 2005). As soluções para estes problemas estão nas técnicas heurísticas de otimização, que proporcionam soluções boas, mas não garante que as soluções encontradas sejam as ótimas. Assim, tais métodos baseados na busca aleatória controlada por critérios probabilísticos, tiveram um importante desenvolvimento

nos últimos anos, principalmente devido ao avanço dos recursos computacionais, uma vez que esses métodos necessitam de um número elevado de avaliações da função objetivo.

Um dos objetivos dos usuários de simulação é se tornarem capazes de realizar simulações que possam trazer resultados mais efetivos ao invés de simplesmente realizar experimentos “cegos” e assumir que ao menos um dos resultados dos experimentos irá trazer o melhor resultado. Como consequência, os desenvolvedores dos softwares de simulação estão se conscientizando cada vez mais da importância de se chegar a resultados ótimos, ou mais próximos do ótimo em poucos minutos, ao invés de se realizar experimentos exaustivos que muitas vezes duram dias ou até meses (CARSON e MARIA, 1997).

Em 1995, a ProModel *corporation* decidiu incorporar o SimRunner ao seu software de simulação com o objetivo de facilitar o trabalho dos usuários que desejavam associar os conceitos de otimização e simulação com o objetivo de obter melhores resultados em seus trabalhos. O SimRunner utiliza um método baseado em algoritmos evolucionários. Embora seja fácil de usar, pode ser melhor utilizado se houver um entendimento básico de como ele busca a solução ótima.

A procura pela solução ótima pode ser conduzida manualmente ou automatizada com algoritmos especialmente designados para procurar a solução ótima sem executar a avaliação de todas as soluções possíveis. Dentre as maiores vantagens de se utilizar tais algoritmos estão (HARREL *et al*, 2000):

1. Automatização de parte do processo de análise, economizando o tempo do analista;
2. Um método lógico é usado para eficientemente explorar a superfície de resposta buscando as melhores combinações de possíveis respostas;
3. O método mostra diversos resultados considerados como os “melhores” para o analista.

A última vantagem é a mais importante, pois possibilita ao analista a visualização de soluções que o mesmo desconhecia.

4.3. Algoritmos Evolutivos

Algoritmos evolutivos (EAs) são uma classe de técnicas de procura direta baseados na teoria da evolução. Eles simulam o processo de evolução onde os seres vivos se adaptam ao meio ambiente para sobreviver. EAs manipulam uma população de solução de forma que as piores soluções desaparecem e as melhores evoluem em busca da solução ótima. Técnicas de procura baseadas nesse conceito possuem grande robustez e tem sido utilizadas para resolver problemas de grande complexidade (HARREL *et al*, 2000).

As teorias evolucionárias mais amplamente aceitas baseiam-se no paradigma neo-Darwiniano (FOGEL,1994). Tais teorias afirmam que a história da vida pode ser completamente descrita por processos físicos (reprodução, mutação, competição e seleção) que operam sobre ou entre populações e espécies. Os indivíduos e as espécies podem ser vistos como uma combinação de sua programação genética (genótipo) e a expressão de seu comportamento, determinado pelas condições ambientais (fenótipo), sobre os quais estes processos físicos agem. A evolução é vista como um processo que age em busca pelo ótimo, pois a seleção produz fenótipos tão próximos do ótimo quanto possível, a partir da especificação de um estado inicial e das restrições ambientais.

Algoritmos evolutivos diferem das técnicas não lineares de otimização em vários aspectos. As diferenças mais significantes estão na forma como eles conduzem a busca pela superfície de resposta utilizando uma população de soluções ao invés de utilizar somente um valor, permitindo a coleta de informações sobre a superfície de resposta de diversas formas simultaneamente. Tal abordagem aumenta as chances de se encontrar a solução ótima (HARREL *et al*, 2000).

Os AEs são divididos, de uma forma geral, nos seguintes grupos:

1. Estratégias evolutivas: As estratégias evolutivas foram desenvolvidas pelos alemães Ingo Rechenberg e Hans-Paul Schwefel em meados da década de 70, com o objetivo inicial de solucionar problemas complexos, discretos e contínuos, principalmente experimentais (BÄCK *et al.*, 1997).

Ao contrário dos algoritmos genéticos, que também empregam cruzamentos, este método emprega apenas operadores de mutação. Em um esquema bem simplificado da estratégia evolutiva, um indivíduo-pai gera um só filho através de aplicação de mutações de distribuição Gaussiana, média zero e variância variável, de modo que pequenas mutações ocorram mais freqüentemente que mutações mais radicais. Sempre que um filho “melhor” que o pai é gerado, o pai é substituído e o processo é reiniciado (TANOMARU, 1995).

2. Programação evolutiva: Os métodos de programação evolutiva (também chamados de programação evolucionária) foram desenvolvidos por Lawrence J. Fogel em 1962, e foi originalmente concebido como uma tentativa de criar inteligência artificial (BÄCK *et al.*, 1997). Porém este método tem sido bastante usado em problemas de otimização e é, neste caso, virtualmente equivalente às estratégias evolutivas, diferenciando-se apenas nos procedimentos de seleção e codificação de indivíduos.

Segundo TANOMARU (1995), na programação evolutiva há uma população de N indivíduos que são copiados na totalidade numa população temporária e sofrem mutações variáveis. Um torneio estocástico é realizado para extrair a população seguinte desse grupo de indivíduos. Não há nenhuma restrição que implique em um tamanho de população constante e não há recombinação entre os indivíduos.

3. Algoritmos genéticos: Algoritmos genéticos (AGs) são uma técnica de resolução de problemas com a interessante propriedade de buscar a solução de forma similar a teoria da evolução. São algoritmos de pesquisa baseados no mecanismo de genética e na seleção natural, utilizando operadores que emulam estes princípios para guiá-los até que a solução seja encontrada (BARCELLOS, 2000).

Segundo DIAS (2004) “as melhores técnicas de computação evolutiva são os algoritmos genéticos: muito freqüentemente os termos ‘métodos de computação evolutiva’ e ‘métodos baseados em algoritmos genéticos’ são usados de forma alternada”. Neste capítulo serão apresentadas as principais características de um AG tradicional, que pode ser utilizado para resolver um grande número de problemas de otimização. Também serão apresentadas propostas encontradas na literatura para aperfeiçoamento do algoritmo genético tradicional, com o intuito de melhorar a qualidade das soluções obtidas e o seu desempenho global.

4.4. Algoritmos Genéticos

No mundo real, o processo de seleção natural controla a evolução dos organismos, de forma que aqueles mais adaptados ao meio-ambiente, têm vida mais longa e se reproduzem mais. Por outro lado, os organismos menos adaptados tendem a morrer precocemente ou procriar-se menos (CHAMBERS, 1995). De acordo com a teoria da evolução, as principais características dos organismos que tornam possível sua sobrevivência no planeta são determinadas por reações químicas nos cromossomos feitos de DNA.

A utilização de algoritmos genéticos traduz-se na aplicação de uma forma simples da evolução para ensinar os organismos a encontrar um determinado objetivo. Se este objetivo é a formulação de um problema que se quer resolver, é possível encontrar a solução do mesmo pela observação do comportamento e das características das sucessivas gerações de organismos (CHAMBERS, 1995).

Comparados com os tradicionais métodos de otimização, os algoritmos genéticos apresentam vantagens como: robustez, paralelismo natural, independência, simplicidade e integração (DORNELLAS, 1997). Uma vantagem adicional é a oportunidade de modelar problemas de otimização para os quais não existe uma função objetivo explícita, ou não se pode apresentar numa função matemática, sendo necessário um modelo de simulação para avaliar o desempenho da solução candidata. Uma dificuldade que pode surgir na aplicação dos algoritmos genéticos é a definição de uma regra de parada. A solução ideal seria aquela na qual o algoritmo pára sempre que a solução ótima é atingida. As regras de paradas usuais são número máximo de gerações ou um limite de tempo de processamento. Uma alternativa é parar o algoritmo sempre que não exista aperfeiçoamento substancial após algumas gerações.

Ainda que sejam inerentemente aleatórios, os algoritmos genéticos utilizam, de forma bastante eficiente, as informações históricas contidas em seu código para a partir daí sugerir novas soluções que melhoram o resultado final do problema. Por isso, são classificados como algoritmos de inteligência emergente (BARCELLOS, 2000). HOLLAND (1975) demonstrou que os algoritmos genéticos percorrem o espaço amostral em ordem cúbica, ou seja, enquanto as iterações crescem em ordem p (o que é computacionalmente interessante) a quantidade de pontos percorridos no espaço amostral cresce em ordem cúbica. Isto é particularmente interessante em

casos de problemas extremamente onerosos em tempo. AGUIAR e COSTA (1997) mostram uma aplicação na qual uma pesquisa exaustiva (varredura completa do espaço amostral) que levaria em média 50 horas pode ser realizada por uma implementação de algoritmos genéticos em aproximadamente 38 segundos.

4.4.1. Funcionamento e nomenclatura

Os estágios que compõem o funcionamento dos algoritmos genéticos são:

1. Criação da população inicial, constituída por indivíduos que representam possíveis soluções do problema;
2. Avaliação da aptidão de cada indivíduo da população com a função objetivo;
3. Seleção dos indivíduos para reprodução conforme a aptidão dos mesmos;
4. Reprodução dos indivíduos selecionados utilizando-se operadores genéticos como o *crossover* e a mutação;
5. Parâmetros e critérios de parada e substituição de parte da população com uma nova geração de indivíduos (GOONATILAKE e TRELEAVEN, 1995).

A unidade básica dos AGs é o gene, que controla uma determinada propriedade de um indivíduo. O conjunto de genes forma o cromossomo, que representa, através de uma codificação apropriada, uma possível solução do problema para uma dada função objetivo. Nos algoritmos genéticos, os cromossomos são tipicamente representados como uma *string* de bits. Contudo, esta representação também pode ser realizada através de números inteiros ou reais (CASTRO, 2001). A **Figura 4.1** representa de um exemplo de cromossomo composto por seis genes, por meio de uma *string* de valores binários.

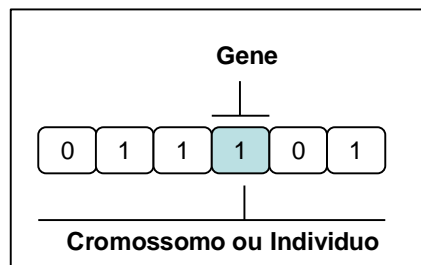


Figura 4.1: Representação de um cromossomo
Fonte: ISIDORO (2001)

A população é o conjunto de cromossomos, ou indivíduos, envolvidos em um determinado momento do processo de busca da solução. O tamanho da população indica o número de pontos do espaço de busca que estão sendo considerados em paralelo. A solução do problema apresenta-se de duas formas: o cromossomo com a informação genética crua (genótipo) manipulada pelos AGs e o fenótipo que é a expressão do cromossomo em termos do modelo (CHAMBERS, 1995).

A avaliação da adequabilidade de cada indivíduo ao meio é realizada através da função de *Fitness*, que estabelece critérios para determinar que cromossomos são boas soluções para o fenótipo, dentro do espaço de otimização. Para cada cromossomo está associado uma função de *Fitness*.

A **Figura 4.2** representa o ciclo de funcionamento dos algoritmos genéticos (CASTRO, 2001):

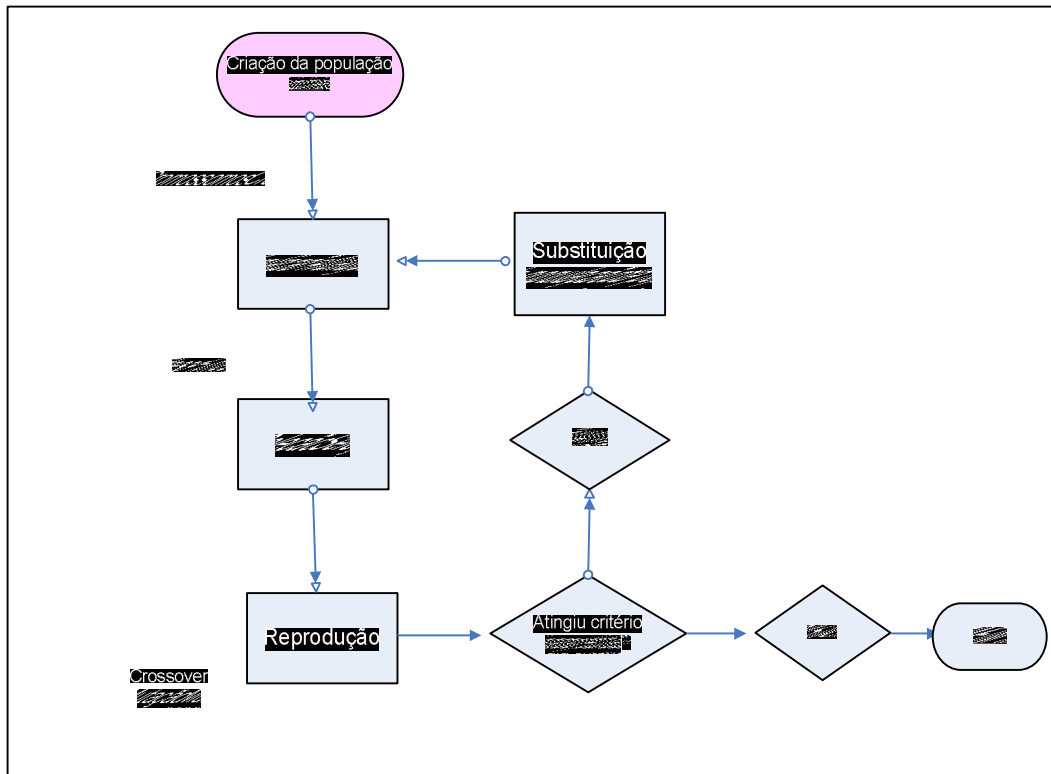


Figura 4.2 - Ciclo de operações dos algoritmos genéticos
Adaptado de CASTRO 2004

O pseudocódigo de um AG básico é mostrado na **Figura 4.3**. Nele pode-se ver que os AG's começam com uma população de p estruturas aleatórias (indivíduos), onde cada estrutura codifica uma solução do problema. O desempenho de cada indivíduo é avaliado com base numa função de avaliação de aptidão. Os melhores tenderão a ser os progenitores da geração seguinte, melhorando, de geração para geração, através da troca de informação.

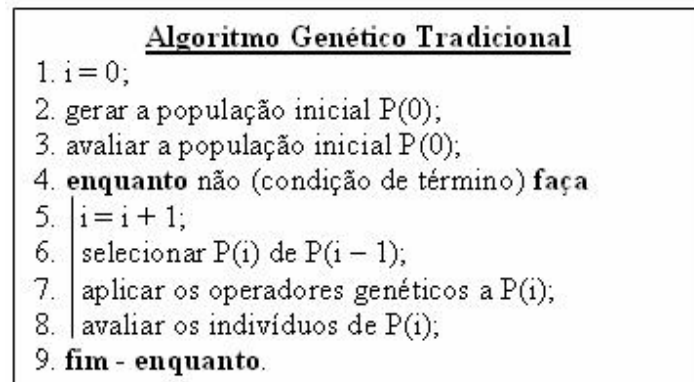


Figura 4.3 – Pseudocódigo de um algoritmo genético
Fonte: DIAS (2004)

4.4.2. Operadores Genéticos

A função dos operadores genéticos é através de um processo iterativo, transformar a população inicial em uma população que represente um resultado satisfatório (ISIDORO, 2004). Um algoritmo genético clássico é composto de três operações: seleção, cruzamento e mutação (CASTRO, 2001).

1. Seleção/Reprodução

A cada iteração, ou geração, de um algoritmo genético, partindo-se de uma população de p indivíduos da geração anterior deve ser obtida uma nova população de p indivíduos, sobre a qual serão aplicados os operadores genéticos. Este processo envolve os passos de seleção e reprodução dos indivíduos. A partir dos indivíduos escolhidos no processo de seleção, a reprodução consiste em copiar o código genético dos indivíduos selecionados para a nova população.

Tendo em vista que, ao término de uma iteração, todos os indivíduos são avaliados, é possível saber quais deles possuem qualidades superiores aos demais, como solução para o problema modelado. Dessa forma, a seleção de indivíduos deve corresponder a um mecanismo que permita a sobrevivência dos melhores indivíduos para que estes possam compartilhar suas características com as gerações seguintes.

Existem várias estratégias para a seleção. BEASLEY, BULL e MARTIN (1993) afirmam que não existe um método considerado absoluto, tendo em vista que os ajustes nas estratégias podem levar a desempenhos semelhantes.

As duas estratégias mais utilizadas para a seleção são descritas a seguir:

1.1. Seleção proporcional à aptidão: criada por HOLLAND (1975), esta estratégia é inspirada na seleção natural, que envolve a seleção dos indivíduos conforme o valor da sua aptidão. Uma implementação desta estratégia é conhecida como “método da roleta”, que utiliza-se de uma analogia com o jogo de roleta encontrado em cassinos. No método da roleta cada indivíduo possui uma região da roleta proporcional ao valor da sua aptidão e assim uma determinada probabilidade de ser selecionado. Cada vez que a roleta é girada um indivíduo é selecionado. O número total de vezes que a roleta é girada correspondente ao tamanho da população, podendo um mesmo indivíduo ser selecionado mais de uma vez;

1.2. Seleção por torneio: existem muitas variações desta estratégia. A mais simples envolve a escolha de forma aleatória de t indivíduos da população. Em seguida os indivíduos escolhidos competem entre si e o indivíduo selecionado será aquele que possuir o melhor valor para a aptidão. Normalmente t é utilizado com o valor 2, e um aumento neste valor irá acelerar a convergência da população.

Outro aspecto a ser considerado no processo de seleção é o elitismo, que é uma estratégia utilizada em conjunto com as outras já descritas para a seleção. O elitismo é utilizado para garantir que o melhor indivíduo da população de uma geração seja reproduzido na população da geração seguinte.

Segundo BEASLEY, BULL E MARTIN (1993), o processo de seleção influencia muito o comportamento de um AG, sendo um dos aspectos críticos para que a evolução da população ocorra. O tipo de seleção utilizado pode produzir problemas como a convergência prematura para ótimos locais distantes de um ótimo global ou o caso oposto, onde a convergência é muito lenta.

A convergência prematura ocorre quando as características de indivíduos com uma elevada aptidão (mas que não correspondem à solução ótima) dominam rapidamente a população, que converge para um ótimo local de baixa qualidade. A convergência lenta, ou a não convergência, é

um problema oposto à convergência prematura, em que ótimos locais de boa qualidade, ou um ótimo global, podem também não serem alcançados. Neste caso o valor médio da aptidão normalmente é alto e a diferença entre o melhor indivíduo e o valor médio é muito pequena. Dessa forma, não existe diversificação suficiente na população para que o AG consiga continuar a sua evolução em direção à solução ótima.

2. Cruzamento (*Crossover*)

O operador de cruzamento (*crossover* ou recombinação) cria novos indivíduos através da combinação de dois ou mais indivíduos. A idéia intuitiva por trás deste operador é a troca de informação entre diferentes soluções candidatas. Geralmente dois indivíduos progenitores são escolhidos da população, por um método aleatório com probabilidade definida por uma “taxa de cruzamento” para produzir dois novos indivíduos (TANOMARU, 1995).

Segundo DIAS (2004), o operador de cruzamento é a principal força direcionadora em um AG. O operador de cruzamento realiza a troca de partes de pares de indivíduos com o objetivo de tentar obter indivíduos melhores a partir dos indivíduos selecionados. Dessa forma, o principal objetivo do cruzamento é utilizar o conhecimento obtido em pontos do espaço de busca visitados previamente. A aplicação do operador de cruzamento a um par de indivíduos normalmente está sujeita a uma taxa de probabilidade de aplicação, definida como parâmetro para a execução do AG.

A partir da seleção de um par de indivíduos, existem diversas formas de utilização do operador de cruzamento. As mais comuns são:

2.1. Cruzamento de um ponto (*single-point crossover*): a partir de um ponto de cruzamento dos indivíduos envolvidos, obtido de forma aleatória, os valores dos trechos situados após o ponto de cruzamento são trocados entre os indivíduos do par. Um exemplo é apresentado na **Figura 4.4**;

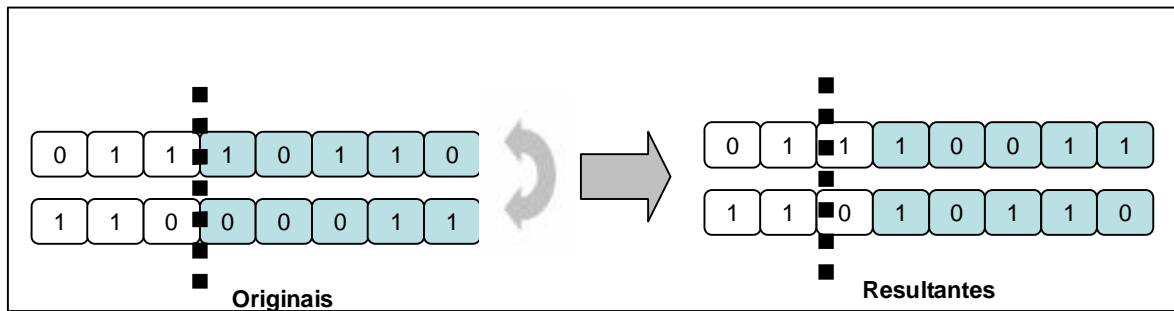


Figura 4.4 – Exemplo de aplicação do operador de cruzamento de um ponto
 Fonte: DIAS (2004)

2.2. Cruzamento de múltiplos pontos (*two-point crossover*): é uma generalização do cruzamento de um ponto, em que pares de pontos de cruzamento são obtidos de forma aleatória e os valores dos indivíduos localizados entre cada par de pontos de cruzamento são trocados. Um exemplo é apresentado na **Figura 4.5**;

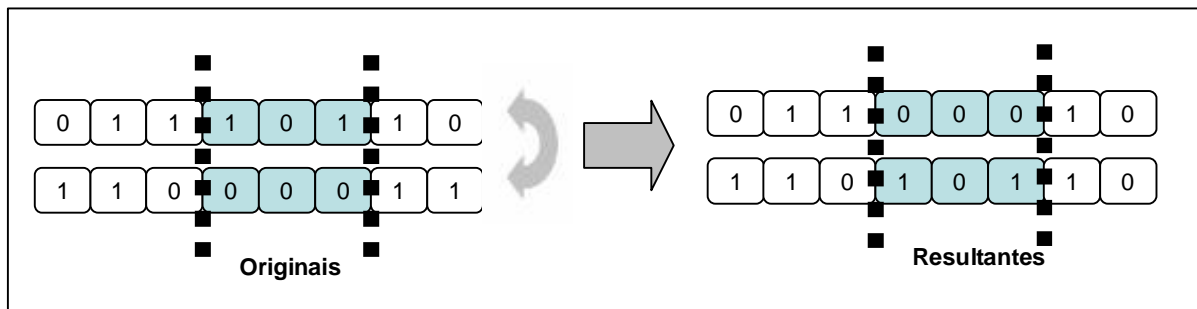


Figura 4.5 – Exemplo de aplicação do operador de cruzamento de múltiplos pontos
 Fonte: DIAS (2004)

2.3. Cruzamento uniforme (*uniform crossover*): uma máscara de dígitos binários é obtida de forma aleatória, onde o dígito 1 indica que o valor na respectiva posição dos indivíduos deverá ser trocado e o dígito 0 indica que os valores originais das posições dos indivíduos envolvidos, equivalentes às posições da máscara, devem ser mantidos com os valores originais. Um exemplo é apresentado na **Figura 4.6**.

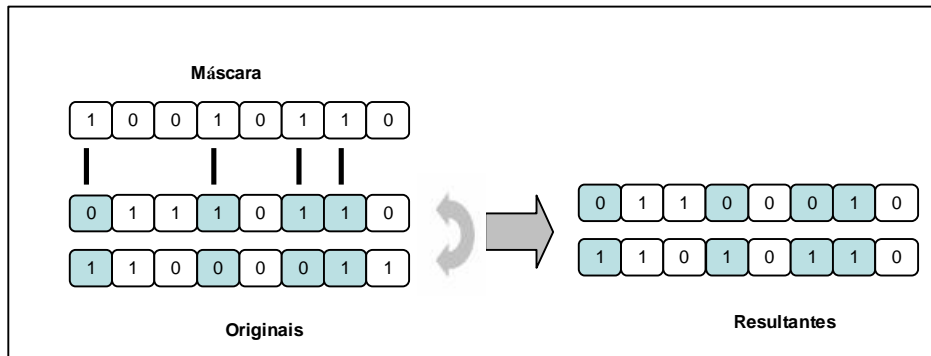


Figura 4.6 – Exemplo de aplicação do operador de cruzamento uniforme
 Fonte: DIAS (2004)

3. Mutação

Tem o objetivo de alterar aleatoriamente a característica de um indivíduo, através da mudança do valor de um gene. Uma vez que as características dos “descendentes” são limitadas à constituição de seus “ancestrais”, a mutação possibilita o aparecimento de indivíduos com características até então não identificadas (CUNHA e PINTO, 2001). Esta alteração garante que a probabilidade de chegar a qualquer ponto da superfície de resposta nunca seja zero, além de contornar o problema de ótimos locais.

Quando são utilizadas representações binárias, o operador de mutação escolhe aleatoriamente uma posição do cromossomo e altera-o de acordo com o procedimento ilustrado na **Figura 4.7** (TAVARES, 2000). Este operador não ocorre em todos os cromossomos, sendo sua aplicação determinada por uma “taxa de cruzamento”.

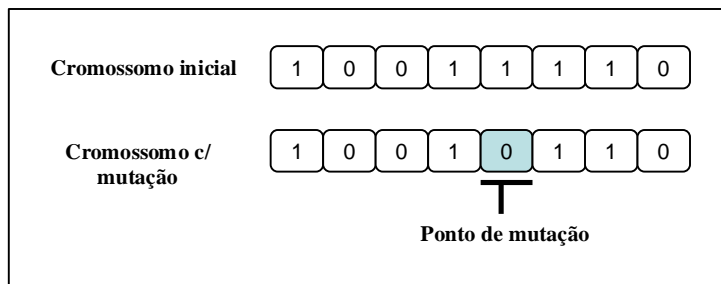


Figura 4.7 – Exemplo de aplicação do operador mutação
 Fonte: ISIDORO (2001)

3.1. Parâmetros: Os AGs dependem essencialmente de um conjunto de parâmetros que devem ser definidos. Os principais parâmetros são (TAVARES, 2000):

3.1.1. Taxa de Cruzamento: é definida como a medida da possibilidade de aplicação do operador de cruzamento a um dado par de indivíduos. Os valores típicos para esta taxa situam-se no intervalo de 0,6 a 1,0. Quanto maior for esta taxa, maior é a quantidade de indivíduos introduzidos na população. Para valores abaixo desta taxa, gera-se menos indivíduos em cada geração, o que pode originar um aumento do número de gerações para obter os mesmos resultados.

3.1.2. Taxa de Mutação: é uma medida da taxa de ocorrência da operação mutação sobre um dado cromossomo. Dado que uma taxa de mutação elevada tornará o processo essencialmente aleatório, é usual esta taxa assumir valores relativamente baixos que, tipicamente, estão no intervalo de 0,001 a 0,1.

3.1.3. Taxa de Substituição: define qual a proporção de indivíduos da população sera substituída em cada geração. Se a percentagem de indivíduos a substituir for de 100% todos os indivíduos da população atual são substituídos pelos novos indivíduos resultantes da reprodução. Quanto menor for o valor desta taxa, menor será a diferenciação genética entre gerações e deste modo existirá uma convergência do algoritmo mais lenta.

3.1.4. Critério de parada: depende do problema e do esforço computacional exigido. Em face do tempo e dos recursos disponíveis, é necessário definir qual a qualidade da solução desejada. Um critério usado com frequência passa por definir o número máximo de gerações em que a evolução deve ocorrer. Um segundo critério possível passa pela definição de um valor mínimo para o desvio padrão do valor de aptidão dos indivíduos na população. Uma vez atingido esse valor mínimo o algoritmo pára. Outro critério bastante comum de parada consiste em fazer evoluir o algoritmo até se verificar que não se registram melhorias significativas das soluções ao longo de um dado número de gerações.

Segundo CUNHA e PINTO (2001), não existem parâmetros ótimos para os “operadores genéticos”, eles são específicos para cada problema e cada esquema de codificação, não podendo ser adotados de forma genérica, uma vez que comprometem o desempenho global do algoritmo.

4.5. Considerações finais

Este capítulo teve início buscando apresentar o conceito e alguns métodos de otimização, passando pelos algoritmos evolutivos e se focando nos algoritmos genéticos. Através da revisão bibliográfica pôde-se verificar que se trata de um método flexível e indicado para o uso conjunto com simulação. Tal associação tem se tornado bastante difundida.

Capítulo 5

OTIMIZAÇÃO via SIMULAÇÃO

5.1 Considerações iniciais

O objetivo deste capítulo é ilustrar a associação entre os conceitos de simulação e otimização, apresentados anteriormente, explicando seu funcionamento e metodologia. Além disso, será mostrado ainda o funcionamento do SimRunner®, incorporado ao pacote de simulação ProModel®, utilizado neste estudo.

5.2 Introdução

Conforme descrito anteriormente, dentre as vantagens de se utilizar a simulação está a possibilidade de se responder a questões do tipo “o que aconteceria se...?”, ou seja, avaliando os resultados do modelo para determinadas condições. Assim, para usar a simulação na avaliação e melhoria do desempenho de um processo, é necessário construir cenários e logo em seguida executar a simulação para cada uma deles, analisando os resultados encontrados (OPTQUEST FOR ARENA USER’S GUIDE, 2002). Tal processo, apesar de ser capaz de gerar resultados úteis, pode ser muito cansativo consumir muito tempo, além disso, não garante na maioria das vezes que as melhores configurações sejam experimentadas.

O intuito de se associar as técnicas de otimização e simulação é justamente resolver tais problemas. Esta associação apresenta algumas terminologias diferentes como “otimização para simulação”, “otimização via simulação” e “otimização em simulação”, sendo estas duas últimas as mais utilizadas (FU, 2002).

Nas últimas décadas simulação e otimização eram mantidas separadamente na prática. Atualmente essa integração tem se mostrado bastante difundida, principalmente pelo fato de alguns pacotes de simulação incluírem rotinas de otimização. FU (2002)

A **Tabela 5.1** apresenta exemplos de alguns *softwares* de otimização associados a pacotes de simulação e as técnicas de otimização utilizadas.

Tabela 5.1: Softwares de Otimização

Software de Otimização	Pacote de simulação	Técnica de otimização
AutoStat	AutoMod	Algoritmos Evolutivos e Algoritmos genéticos
OptQuest	Arena, Crystal Ball, etc.	Busca Scatter , Tabu e Redes Neurais
OPTIMIZ	SIMUL8	Redes Neurais
SimRunner	ProModel	Algoritmos Evolutivos e Algoritmos genéticos
Optimizer	WITNESS	Simulated annealing e busca tabu

Fonte: FU (2002)

5.3 Funcionamento

Conforme definido anteriormente, um sistema pode ser descrito como uma relação *input-output*, onde X é o *input*, Y o *output* e M é uma representação a qual correlaciona as informações de entrada e saída.

A partir deste conceito de sistema é possível fazer uma comparação entre modelagem, simulação e otimização: a modelagem é a busca das inter-relações existentes entre os dados de entrada e de saída de um determinado sistema, ou seja, uma representação de seu comportamento; a simulação manipula as entradas de um modelo e verifica suas diferentes saídas; já a otimização busca obter um *output* ótimo, previamente definido, alterando a composição dos *inputs* (PROTIL, 2001).

Tal comparação é apresentada na **Figura 5.1**.

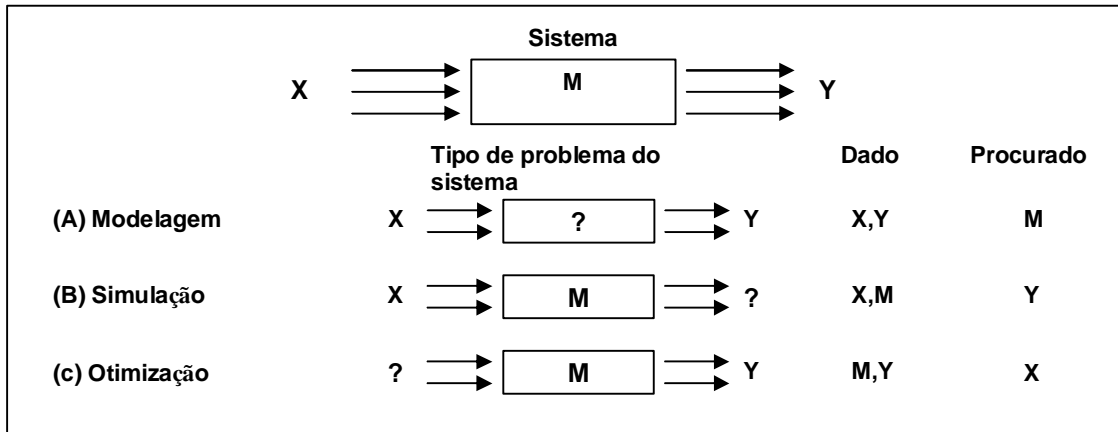


Figura 5.1: Questionamentos na modelagem, simulação e otimização
 Fonte: Adaptado de PROTIL (2001).

O processo de otimização testa várias combinações de valores para variáveis que podem ser controladas (variáveis independentes), na busca da solução ótima. Em muitos casos não é possível avaliar todas as combinações prováveis de variáveis devido ao grande número de combinações possíveis. Nestes casos o uso de algoritmos heurísticos de otimização devem ser utilizados. Para FU (2002), a otimização deve ocorrer de maneira complementar a simulação, fornecendo as variáveis de uma possível solução (*inputs*) à simulação, e esta, fornecendo respostas (*outputs*) para a situação proposta, que retornam à otimização, caso a solução não seja considerada satisfatória.

A otimização gera novas variáveis, utilizando técnicas de otimização específicas, que serão novamente testadas pela simulação. Este ciclo, representado na **Figura 5.2**, é repetido até sua parada, definida de acordo com o método de otimização utilizado. Quando o método de otimização é baseado em algoritmos genéticos, para cada possível solução é efetuada uma tentativa, ou seja, um ciclo.

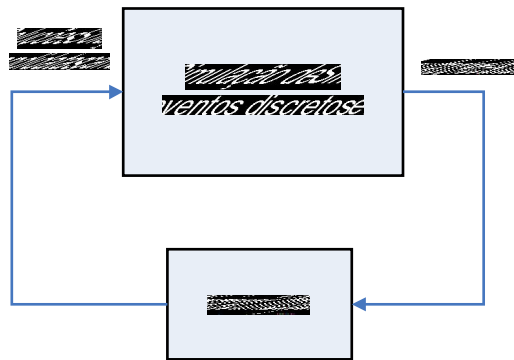


Figura 5.2: Otimização em simulação
Fonte: FU (2002).

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

5.4. Metodologia para otimização via simulação

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 função objetivo, que pode de maximização ou minimização, cujo o 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 (SILVA, 2005).

Uma metodologia específica para o uso do SimRunner® é proposta por HARREL *et al.* (2000). Segundo o autor, após a construção e validação de um modelo, são necessários alguns passos para uma otimização bem sucedida, que são:

1. Definir as variáveis que afetarão as respostas do modelo e que serão testadas pelo algoritmo de otimização. São estas variáveis que terão o valor alterado a cada rodada de simulação;

2. Definir o tipo de variável (real ou inteira) e limites inferiores e superiores. Durante a busca, o algoritmo de otimização gerará soluções respeitando o tipo das variáveis e seus limites. O número de variáveis de decisão e a gama de valores possíveis afetam o tamanho do espaço de busca, alterando a dificuldade e o tempo consumido para identificar a solução ótima. É por isso que se recomenda que somente as variáveis que afetem significativamente o modelo sejam usadas;

3. Definir a função objetivo para avaliar as soluções testadas pelo algoritmo. Na verdade, a função objetivo já poderia ter sido estabelecida durante a fase de projeto do estudo de simulação. Esta função pode ser construída tendo por base peças (*entities*), equipamentos (*locations*), operários (*resources*) entre outros, buscando minimizar, maximizar ou fazer uso de ambos em diferentes variáveis, dando inclusive pesos diferentes para compor a função objetivo;

4. Selecionar o tamanho da população do Algoritmo Evolutivo. O tamanho da população de soluções usado para conduzir o estudo afeta a confiabilidade e o tempo requerido para a condução da busca, assim, é necessário que haja um equilíbrio entre o tempo requerido e o resultado esperado da otimização. Nesta fase também é importante definir outros parâmetros como: precisão requerida, nível de significância e número de replicações;

5. Após a conclusão da busca um analista deve estudar as soluções encontradas, uma vez que, além da solução ótima, o algoritmo encontra várias outras soluções competitivas. Uma boa prática é comparar todas as soluções tendo como base a função objetivo. Mesmo seguindo metodologias para a execução da otimização em simulação alguns fatores afetam diretamente o desempenho da busca, entre eles: precisão do modelo, número de variáveis, complexidade da função objetivo, valores iniciais das variáveis e seus limites (OPTQUEST FOR ARENA USER'S GUIDE, 2002).

5.5. Casos na literatura

O'FERREL (1995) utilizou a simulação para modelar um sistema de manufatura durante o período em que parte de seus funcionários se encontram em treinamento. A otimização foi utilizada para otimizar o funcionamento do sistema durante o período de ausência normal dos seus funcionários para treinamento.

LEE e KIM (2000) utilizaram a simulação associada a otimização para determinar a ordem de produção que maximiza o lucro de uma empresa e a distribuição de mercadorias juntamente com os clientes que minimiza o risco, a técnica de otimização utilizada foi a programação linear.

JOINES *et al* (2000) aplicam a simulação para modelar o complexo transporte de cana de açúcar de Cuba, identificando gargalos e suas causas e ao mesmo tempo, otimizam os recursos alocados no processo como um todo. Técnicas de planejamento de experimentos também foram utilizadas para a identificação de fatores críticos e alcance dos resultados esperados.

PINTO (2001) utilizou a simulação associada a otimização para determinar o tipo de petróleo a ser processado nas refinarias e minimizar os custos globais de suprimento. A aplicação de tais ferramentas possibilitou a criação de uma metodologia para definir a programação da entrada sequencial de itens nos oleodutos e ordem de descarga de navios de petróleo mais adequada segundo critérios pré-definidos de avaliação em que se considera as datas de chegada de navios como variáveis não determinísticas.

DING *et al* (2003) utilizaram a simulação associada a otimização para resolver o problema da seleção de fornecedores a partir de um portfólio pré-definido de possíveis candidatos. Os autores aplicam as ferramentas em um estudo de caso real e resultados satisfatórios são obtidos.

TRUONG E AZADIVAR (2003) aplicaram a simulação para entender os resultados das diferentes configurações de uma dada cadeia de fornecimento e das diferentes decisões estratégicas envolvidas no processo, como decisões de alocação de armazéns, políticas de produção, capacidades de produção e diferentes modais de transporte e distribuição. A otimização foi utilizada para alcançar combinações ótimas capazes de trazer resultados satisfatórios e passíveis de se serem aplicados.

SILVA (2005), em sua dissertação de mestrado utiliza a simulação para aplicar o sistema ABC de custeio em uma linha de produção, verificando as mudanças encontradas nos custos para uma nova configuração do sistema produtivo. Além disso, utilizou a otimização para a redução de tais custos. O software de simulação utilizado foi o ProModel e o software de otimização foi o SIMRUNNER.

5.6. Considerações finais

A simulação e a otimização são métodos de pesquisa operacional bastante difundidos, porém, somente a pouco mais de uma década estes dois métodos vem sendo usados conjuntamente. Este uso conjunto se mostra bastante eficiente e de uso relativamente simples sendo aplicável nas mais diversas áreas. Em um estudo de otimização em simulação tão importante quanto conhecer o funcionamento dos programas é a seqüência e o modo de execução de cada etapa do projeto. Assim é preciso que se siga uma metodologia de implementação consistente e devidamente planejada.

Capítulo 6

APLICAÇÃO

6.1 Considerações iniciais

Este capítulo faz uma descrição do objeto de estudo, caracterizado conforme descrito anteriormente como um sistema puxado de manufatura. O problema abordado neste trabalho consiste em reduzir as quantidades de *kanbans* utilizadas pela organização foco de estudo para cada peça modelada reduzindo o estoque final de peças armazenadas no almoxarifado de materiais acabados da empresa.

6.2. Descrição do objeto de estudo

6.2.1. Aspectos gerais

Segundo SILVA E MENESES (2005), um estudo de caso envolve a análise de um objeto de maneira que se permita o seu amplo e detalhado conhecimento. A análise deste trabalho ocorre em uma empresa do setor automobilístico produtora de componentes eletrônicos localizada no sul de Minas Gerais. Devido a questões de confidencialidade impostas pela organização, o nome da empresa e do produto analisado não serão informados, assim como informações específicas da linha produtiva que possam vir a identificá-la.

A linha de produção escolhida para a realização do estudo produz quatro famílias de peças, totalizando um total de 900 peças diárias. No entanto, para fins de pesquisa, foi escolhida uma família de produtos, referentes a produção de 500 unidades por dia. Tal escolha se justifica pelo fato desta família de peças englobar um número maior de processos que as outras famílias e também pelo fato de se encontrar na classificação A, no que diz respeito a classificação ABC de estoques, conforme **Figura 6.1** (SLACK et al, 2002):

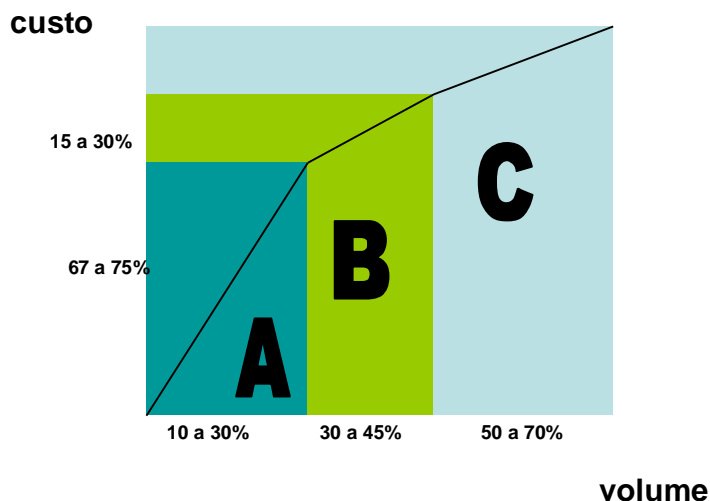


Figura 6.1: Estrutura ABC de estoques
Fonte: SLACK (2002).

Ao longo dos meses de abril, maio, junho e julho de 2006 esse sistema produziu sete tipos de peças, e as mesmas foram devidamente modeladas para que fosse possível chegar aos resultados pretendidos.

6.2.2. Funcionamento

Conforme já afirmado anteriormente, a linha de produção foco de estudo é um sistema puxado, cuja principal característica é a produção das peças conforme as mesmas vão sendo consumidas pelos clientes.

O processo tem início quando as peças são requisitadas pelos clientes diariamente ao responsável no departamento de logística e uma ordem de pedido é enviada pelo mesmo ao almoxarifado de materiais acabados da instituição. No almoxarifado de material acabado os itens estão organizados de três formas:

1. Quantidades de peças estocadas referentes aos cartões verdes;
2. Quantidades de peças estocadas referentes aos cartões amarelos;

3. Quantidades de peças estocadas referentes aos cartões vermelhos.

Os cartões verdes são referentes às condições normais de operação da empresa. O consumo dos cartões amarelos significa que existe uma variação na demanda ou a organização está tendo problemas em alcançar o ritmo de produção necessário ao atendimento do cliente, chamando atenção dos gestores da linha. Já o consumo dos cartões vermelhos traduz uma situação de urgência para a empresa, onde os estoques se encontram quase vazios, traduzindo um consumo excessivo do cliente ou ainda uma segurança extra em caso de quebra de máquinas ou escassez de matéria prima.

Uma vez que os itens são enviados, os *kanbans* que acompanham os itens durante a produção e armazenamento são retirados e enviados a linha produtiva para sua posterior produção, conforme criticidade e prioridades de produção.

A regra de envio ao cliente obedece aos fundamentos do PEPS (primeiro que entra é o primeiro que são), de forma que mesmo sendo necessárias a manutenção de certa quantidade sob condições de armazenagem, as peças que serão produzidas acabam ficando pouco tempo paradas.

Quando os *kanbans* estão dispostos no quadro *kanban* e a produção de um dado dia deve ser iniciada, a produção deve então seguir uma ordem de prioridade que deve ser: Se vários produtos se encontram na fila de produção, terá prioridade sobre os demais aquele que possuir a menor quantidade em estoque no almoxarifado de materiais acabados e o maior número de cartões no quadro *kanban* de produção.

Estabelecida a ordem de produção a ser seguida e disposta no quadro de nivelamento de produção, responsável pela visualização da ordem a ser seguida, cada *kanban* que inicia o processo de produção é quebrado em cento e cinquenta cartões, condizentes com os cento e cinquenta itens de cada lote e são dispostos no início do sistema produtivo para que a produção seja iniciada.

Para a descrição das atividades na linha de produção escolhida foi utilizada a técnica do mapeamento de processo. Como justificativa da utilização de tal técnica pode-se citar o fato desta ser uma ferramenta gerencial analítica e de comunicação que tem a intenção de ajudar a melhorar os processos existentes ou de implantar uma nova estrutura voltada para eles. Além é claro, de ser

uma excelente ferramenta para possibilitar o melhor entendimento dos processos atuais ou simplificar aqueles que necessitam de mudanças (THEISEN, 2004).

É definido também como um auxílio visual na descrição das atividades relacionadas com as entradas, saídas e demais tarefas envolvidas nesse processo de conversão. O mapeamento proporciona aos gestores uma nova percepção de como o trabalho é realizado. Ressaltando os principais passos realizados para produzir um dado produto e onde os principais problemas estão ocorrendo. O mapeamento de processos evidencia as áreas onde uma mudança causaria grande impacto na melhoria da qualidade (ANJARD, 1995). O mapeamento de processo da linha de produção foco do estudo é apresentado na **Figura 6.2**:

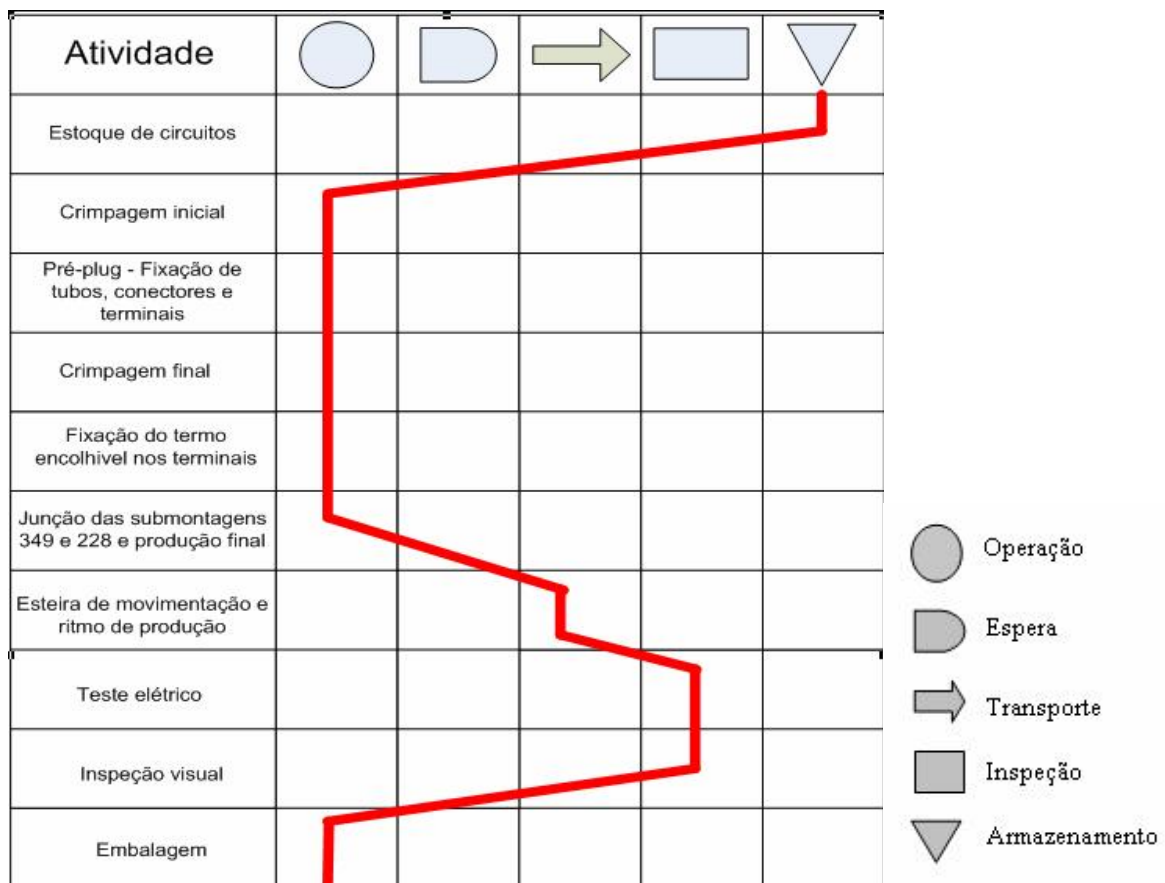


Figura 6.2: Mapeamento de processos

O processo de produção tem início na prensa 1 onde o operador 1 recolhe a matéria prima necessária à produção na loja 1, como circuitos advindos do corte e de outras máquinas. Os itens dispostos na loja 1 seguem um sistema *kanban* como regra de armazenamento sob a forma de dois recipientes de armazenamento. O consumo total de um recipiente indica que o mesmo deve ser substituído por um cheio, tarefa que é realizada por um abastecedor.

De posse dos materiais necessários, o operador 1 realiza a crimpagem dos cabos juntamente com os circuitos e todo o material crimpado passa a ser chamado de submontagem. Em seguida os itens seguem para o pré-plug 1, caracterizada como uma mesa onde as submontagens são dispostas e ocorre a fixação de tubos e outras submontagens pelo operador 2. Em seguida, os itens seguem para o pré-plug 2, também caracterizado como uma mesa, assim como o pré-plug 1, onde o operador 3 realiza a fixação de conectores e dá o *looping*, uma espécie de laço que facilita a movimentação das mercadorias.

Em seguida as submontagens seguem para a prensa 2, onde o operador 4 realiza a crimpagem do outro lado das submontagens e as encaminha para o termo encolhível, onde um material isolante é fixado nos terminais para que o contato com a água durante seu manuseio não possa danificar o material. Concretizada esta etapa tem-se a armazenagem das peças produzidas em um “carrinho de abastecimento”, constituindo o *buffer 1* que deve ficar uma hora adiantado em relação ao restante da produção.

Paralelamente a este processo, outra submontagem vai sendo produzida e sua produção ocorre conforme o *buffer 2*, que se encontra a sua frente vai sendo consumido. Este consumo ocorre juntamente com o constituído anteriormente e também deve estar uma hora adiantado em relação ao restante da produção. A montagem deste item ocorre na prensa 3 pelo operador 6, onde ocorre a crimpagem do material advindo do corte e da loja 1 e em seguida segue para o pré-plug 3, onde ocorre a fixação de tubos e conectores e em seguida a constituição do *buffer 2*.

A etapa final da produção ocorre nas oito mesas de montagens que se encontram dispostas quatro de cada lado e devidamente separadas por uma esteira que dita o ritmo de produção. Cada mesa de montagem possui um operador e realiza a junção das submontagens armazenadas nos *buffers 1* e *2* com diversos componentes e constituindo o produto final produzido pela organização foco de estudo.

A esteira que separa as mesas de montagens possui "oito ganchos" enumerados de um a oito. Cada vez que o gancho um passa pela mesa um, uma peça deve ser disposta na esteira, devendo ocorrer o mesmo com o gancho dois na mesa dois, com o gancho três na mesa três, e assim sucessivamente. Garantindo dessa forma a produção da quantidade necessária diariamente e ditando o ritmo da produção do sistema produtivo.






















A cada cinquenta e cinco segundos uma peça sai da esteira em direção ao teste elétrico e o funcionário responsável atesta se a mercadoria funciona conforme as especificações dos produtos requisitados pelos clientes.

Se a peça não estiver funcionando corretamente então ocorre o envio para o re-trabalho, e após o teste e o funcionamento conforme o esperado, uma etiqueta de conformidade é anexada a peça que em seguida segue para a inspeção visual, que averigua se existem alterações visuais no equipamento. Em caso de não conformidade o equipamento é re-trabalhado, em caso de conformidade o item segue para a embalagem e ao completar cento e cinquenta peças é lacrada, devidamente identificada e enviada por um abastecedor ao almoxarifado de materiais acabados.

6.3. Definição do problema

O objetivo principal deste estudo é o de otimizar as quantidades de *kanbans* verdes das peças produzidas no sistema produtivo foco de estudo durante os meses de abril, maio, junho e julho. Para isso foi necessária a coleta dentre outros dados, das quantidades em estoque dos cartões verde, amarelo e vermelho das sete peças modeladas durante esses meses. A **Tabela 6.1** mostra as quantidades de cartões verde, amarelo e vermelho para as sete peças nos meses de abril, maio, junho e julho.

Tabela 6.1: Quantidades de cartões nos meses de abril, maio, junho e julho.

Peça	1			2			3			4			5			6			7		
Cartões																					
Abril	0	1	0	0	1	0	1	1	0	2	1	1	2	1	1	11	6	5	0	1	0
Maio	0	1	0	1	1	0	1	1	0	1	1	0	2	1	1	8	5	4	1	1	0
Junho	0	1	0	1	1	0	2	1	1	1	1	0	2	1	1	8	5	4	1	1	0
Julho	0	1	0	1	1	0	1	1	0	0	1	0	1	1	0	10	5	5	0	1	0

6.4. Modelagem do sistema

O próximo passo do estudo é a construção de um modelo para simulação do sistema produtivo foco de estudo.

Esta modelagem se deu segundo as metodologias citadas na literatura (SHANNON, 1998, CARLSON, 2003, LAW e KELTON, 2000, PEREIRA, 2000, DUARTE, 2003 e SILVA, 2005). Estes passos são apresentados a seguir.

6.4.1. Formulação do problema

O objetivo a ser alcançado ao final do estudo é a redução do número de *kanbans* do modelo computacional construído através da aplicação da otimização. Segundo HARRELL *et al.* (2000), é aconselhável que os objetivos a serem atingidos na otimização sejam estabelecidos já na primeira fase do estudo de simulação. Um esboço do problema foi feito, definindo os elementos mais importantes do sistema e os dados a serem coletados.

6.4.2. Definição dos objetivos e planejamento do projeto

Para atingir o objetivo de otimizar as quantidades de *kanbans* para as peças devidamente escolhidas nos meses de abril, maio, junho e julho é necessário construir o modelo de forma que o funcionamento do sistema *kanban* seja prioritário e sua otimização cientificamente possível.

Sendo assim, o planejamento envolve duas fases principais, a modelagem, simulação e validação do sistema puxado de produção objeto de estudo, seguida por sua posterior otimização.

6.4.3. Conceitualização do modelo

Segundo os autores citados no início deste tópico, nesta etapa o modelo real sob investigação é resumido através do modelo conceitual, que nada mais é do que uma série de relacionamentos matemáticos e lógicos relativos aos componentes e estrutura do sistema. No caso deste trabalho isso ocorreu através de um fluxograma, conforme **Figura 6.3**:

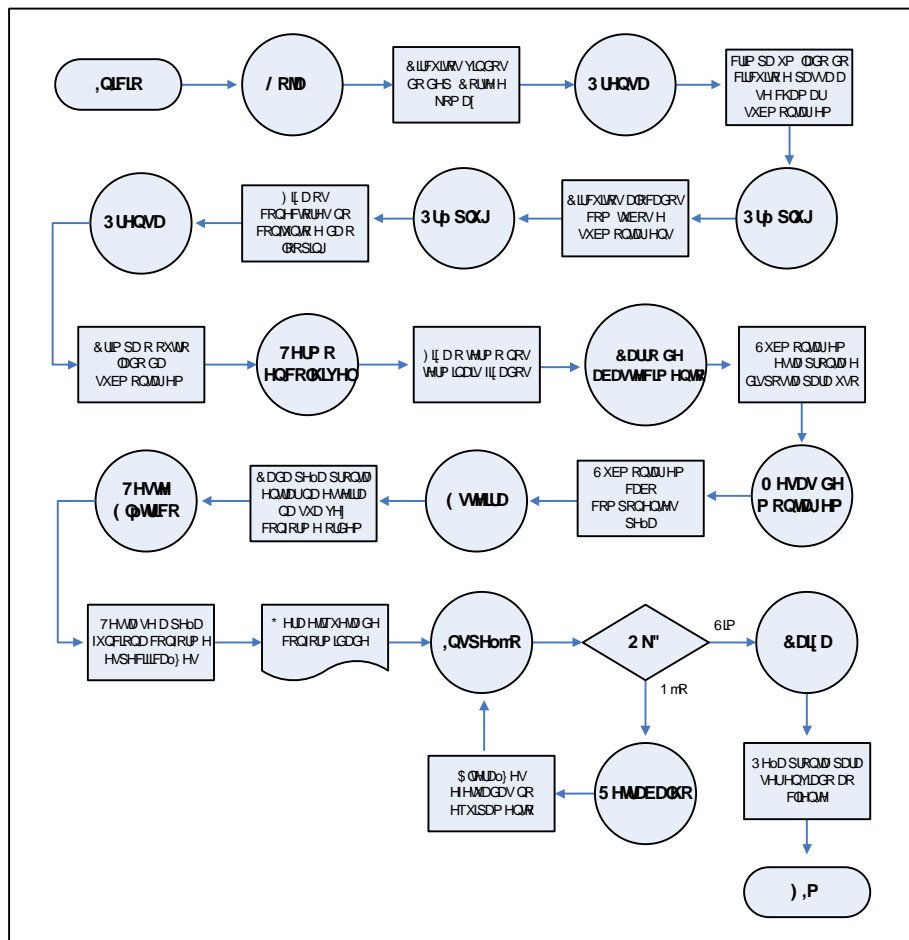


Figura 6.3: Fluxograma do objeto de estudo

6.4.4. Coleta de dados

Segundo HARRELL *et al.* (2000), a coleta de dados é um dos pontos mais importantes do processo de simulação, pois se o dados coletados não foram consistentes o modelo também não o será.

Após a realização de algumas visitas na organização foco de estudo e a realização do mapeamento de processos, a requisição de informações deve ser realizada. Esta etapa deve ocorrer após o mapeamento da linha, de forma que a maior parte das informações necessárias sejam identificadas e devidamente requisitadas.

A listagem das informações requeridas junto ao objeto de estudo para a construção dos modelos computacionais segue abaixo:

1. Quantidade de peças produzidas;
2. Demanda diária;
3. Tamanho dos lotes;
4. Número de operadores;
5. *Takt time*;
6. Tempos de *setup*;
7. Número de máquinas e capacidades de produção;
8. Lead times de produção;
9. Gatilhos de produção;
10. Tempo médio de estocagem em supermercados;

6.4.5. Construção do modelo

Para a realização deste trabalho foi utilizado o pacote de *software* ProModel® da ProModel Corporation, utilizado na disciplina “Simulação” do curso de mestrado em engenharia de produção da UNIFEI, e um dos *softwares* de simulação mais utilizados no mercado (DOLOI e JAFARI, 2003). Este pacote incorpora três programas principais: ProModel® (para simulação de elementos discretos), SimRunner® (para otimização de modelos de otimização) e Stat::Fit® (para estudo de distribuições de probabilidade).

O ProModel é o único simulador no mundo com versão em português. Trata-se do mais avançado software de simulação de eventos discretos, usado para avaliar, planejar e projetar produção, estocagem, logística e outras situações operacionais e estratégicas.

A tecnologia de simulação do ProModel é uma poderosa aliada na constante batalha para se reduzir custos, aumentar capacidade, acelerar ciclos de produção e aumentar serviços a clientes. Empresas em todo o mundo têm constatado a importância desta ferramenta para seu sucesso aplicando suas soluções à milhares de problemas de alta complexidade.

A **Figura 6.4** mostra a tela de abertura do ProModel®.



Figura 6.4 – Tela inicial do ProModel®.

Para construção de um modelo, o ProModel® apresenta os seguintes elementos: “locais”, “entidades”, “recursos”, “processamento” e “chegadas”. Apresenta, ainda, outros elementos auxiliares como: custo, que define as taxas de consumo de recursos, e turnos para definição de turnos de trabalho. As definições e o funcionamento dos principais elementos seguem em seguida:

1. Locais. Representam os lugares fixos do sistema, onde se realizam os processos, são usadas para representar elementos como: estações de trabalho, *buffers*, *conveyors* e filas. Neste elemento pode-se definir: capacidade, unidades (simples ou múltiplas), *setups*, manutenção, nível de detalhamento estatístico, além de regras de chegada e saída de matéria.

A **Figura 6.5** exibe a tela do programa na edição dos locais do caso estudado, representada pelas máquinas.

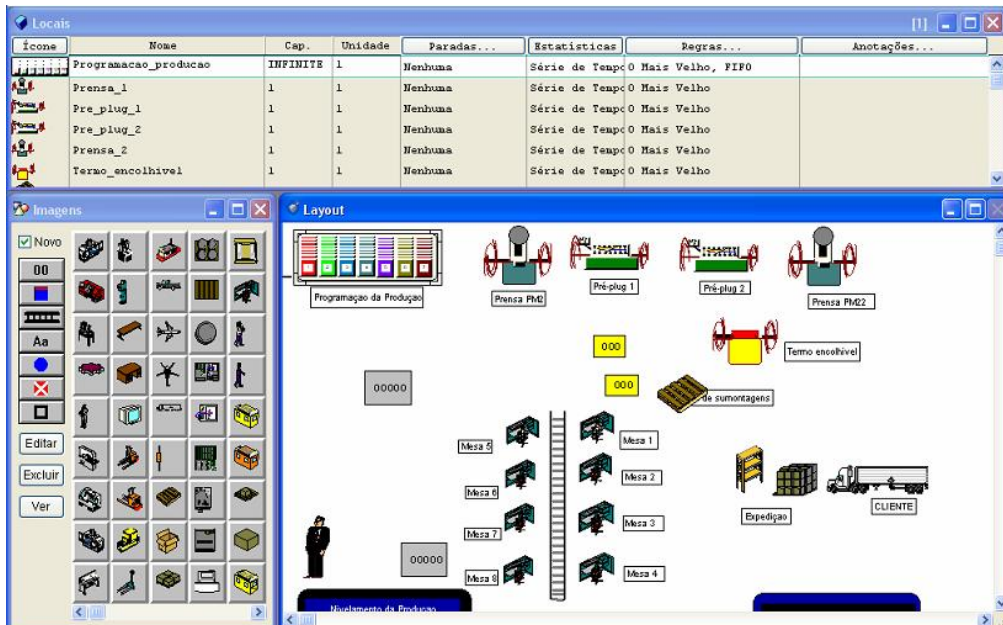


Figura 6.5 – Configuração dos locais.

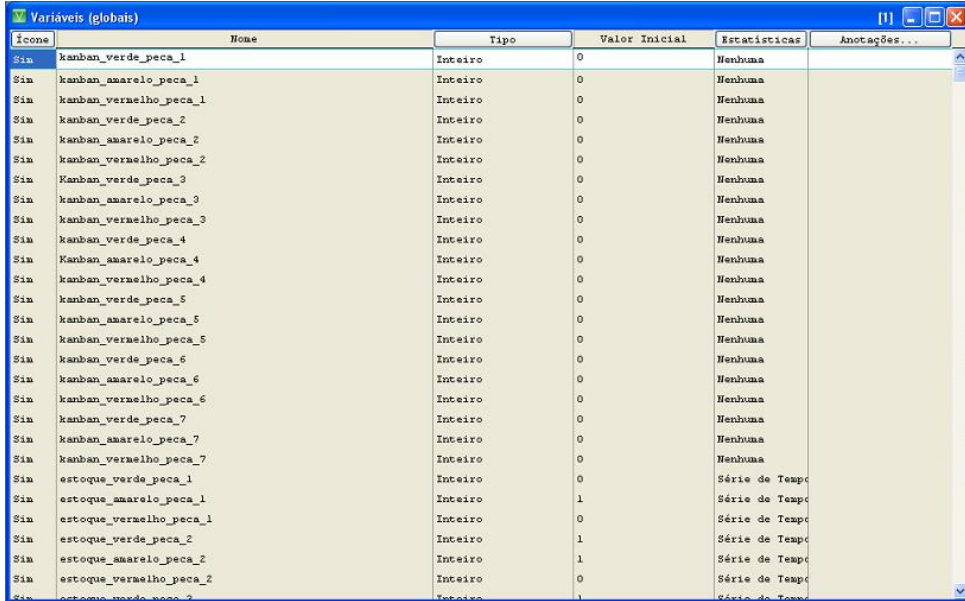
2. Entidades: As entidades são os itens a serem processados pelo sistema, podendo ser: matéria-prima, produtos, *pallets*, pessoas ou documentos. As entidades possuem velocidades definidas, além de nível estatístico como os locais. Elas podem ser agrupadas ou divididas ao longo do processo produtivo, se movimentando de um local para outro utilizando uma rota definida ou uma rede de trabalho. A **Figura 6.6** exibe uma tela do programa com algumas entidades, neste caso com as duas entidades modeladas, a peça e o *kanban*.



Figura 6.6 – Configuração das entidades.

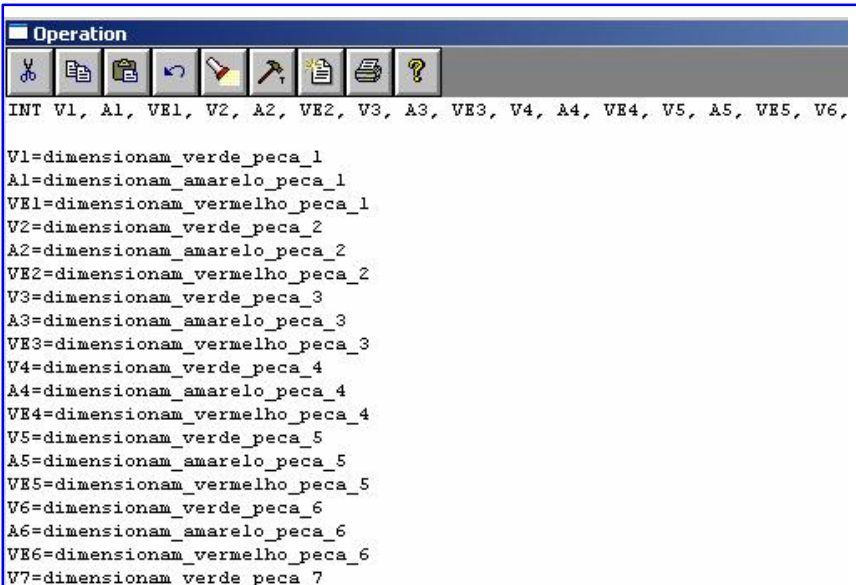
3. Variáveis: Podem ser globais ou locais. As variáveis globais são utilizadas para representar valores numéricos mutáveis. As variáveis locais só estabelecem funções na parte da lógica em que são declaradas. Podendo ambas conter tanto valores numéricos como reais. Uma variável global pode ser referenciada em qualquer lugar do modelo. Já uma local, só dentro de um determinado bloco onde a mesma foi declarada. A **Figura 6.7** ilustra as variáveis globais que

foram criadas para o modelo deste trabalho, responsáveis por monitorar as quantidades em estoque das peças modeladas. Já a **Figura 6.8** ilustra as variáveis locais utilizadas para crescer ou decrescer as quantidades em estoque das peças que são enviadas ao cliente e posteriormente, produzidas.



Ícone	Nome	Tipo	Valor Inicial	Estatísticas	Anotações...
Sim	kanban_verde_peca_1	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_1	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_1	Inteiro	0	Nenhuma	
Sim	kanban_verde_peca_2	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_2	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_2	Inteiro	0	Nenhuma	
Sim	kanban_verde_peca_3	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_3	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_3	Inteiro	0	Nenhuma	
Sim	kanban_verde_peca_4	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_4	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_4	Inteiro	0	Nenhuma	
Sim	kanban_verde_peca_5	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_5	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_5	Inteiro	0	Nenhuma	
Sim	kanban_verde_peca_6	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_6	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_6	Inteiro	0	Nenhuma	
Sim	kanban_verde_peca_7	Inteiro	0	Nenhuma	
Sim	kanban_amarelo_peca_7	Inteiro	0	Nenhuma	
Sim	kanban_vermelho_peca_7	Inteiro	0	Nenhuma	
Sim	estoque_verde_peca_1	Inteiro	0	Série de Tempo	
Sim	estoque_amarelo_peca_1	Inteiro	1	Série de Tempo	
Sim	estoque_vermelho_peca_1	Inteiro	0	Série de Tempo	
Sim	estoque_verde_peca_2	Inteiro	1	Série de Tempo	
Sim	estoque_amarelo_peca_2	Inteiro	1	Série de Tempo	
Sim	estoque_vermelho_peca_2	Inteiro	0	Série de Tempo	
Sim	estoque_verde_peca_3	Inteiro	1	Série de Tempo	

Figura 6.7 – Configuração das variáveis globais.



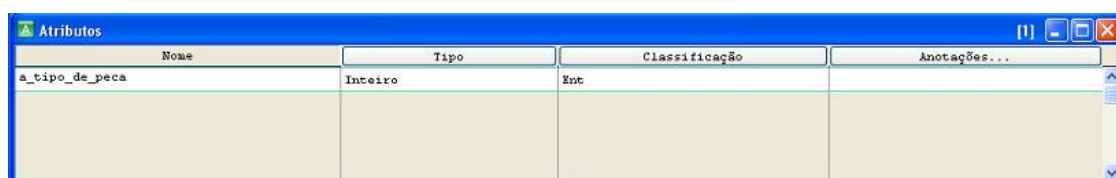
```

INT W1, A1, VE1, V2, A2, VE2, V3, A3, VE3, V4, A4, VE4, V5, A5, VE5, V6,
W1=dimensionam_verde_peca_1
A1=dimensionam_amarelo_peca_1
VE1=dimensionam_vermelho_peca_1
W2=dimensionam_verde_peca_2
A2=dimensionam_amarelo_peca_2
VE2=dimensionam_vermelho_peca_2
W3=dimensionam_verde_peca_3
A3=dimensionam_amarelo_peca_3
VE3=dimensionam_vermelho_peca_3
W4=dimensionam_verde_peca_4
A4=dimensionam_amarelo_peca_4
VE4=dimensionam_vermelho_peca_4
W5=dimensionam_verde_peca_5
A5=dimensionam_amarelo_peca_5
VE5=dimensionam_vermelho_peca_5
W6=dimensionam_verde_peca_6
A6=dimensionam_amarelo_peca_6
VE6=dimensionam_vermelho_peca_6
W7=dimensionam_verde_peca_7

```

Figura 6.8 – Configuração das variáveis locais.

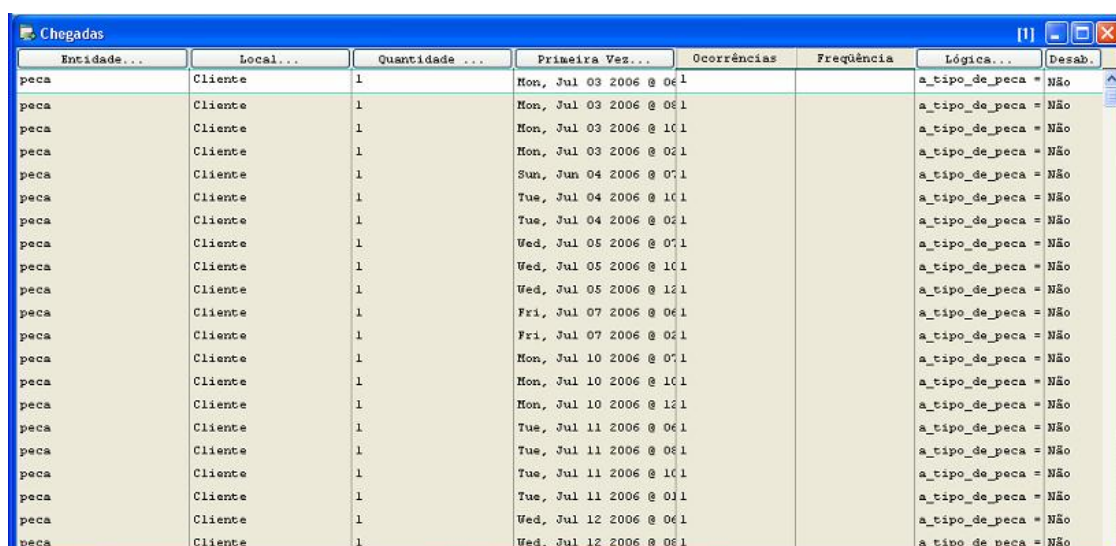
4. Atributos: Similares as variáveis, são definidos juntamente com entidades e locais específicos e geralmente contém informação sobre os mesmos. Podem conter valores reais ou inteiros. Através de sua utilização pode-se criar somente uma entidade referente a um tipo de peça e sete atributos para diferenciar os sete tipos de peças a serem modeladas, como ocorre neste trabalho. A **Figura 6.9** ilustra a configuração dos atributos para este trabalho.



Nome	Tipo	Classificação	Anotações...
a_tipo_de_peca	Inteiro	Ent	

Figura 6.9 – Configuração dos atributos.

5. Chegadas: Define a entrada das entidades dentro do modelo. Podendo ser definidas as quantidades, frequência e períodos de chegada, bem com a lógica de chegada. Pode-se também definir as chegadas através de um arquivo externo de chegada de peças referenciado no Editor de arquivos. A **Figura 6.10** apresenta uma tela com algumas das chegadas do modelo descrito neste trabalho. Trata-se das chegadas do mês de julho, chegadas referentes aos meses de abril, maio e junho foram estabelecidas da mesma forma representando a puxada do cliente que desencadeia o início do processo produtivo.



Entidade...	Local...	Quantidade...	Primeira Vez...	Ocorrências	Frequência	Lógica...	Desab.
peca	Cliente	1	Mon, Jul 03 2006 @ 06:1			a_tipo_de_peca = Não	
peca	Cliente	1	Mon, Jul 03 2006 @ 08:1			a_tipo_de_peca = Não	
peca	Cliente	1	Mon, Jul 03 2006 @ 10:1			a_tipo_de_peca = Não	
peca	Cliente	1	Mon, Jul 03 2006 @ 02:1			a_tipo_de_peca = Não	
peca	Cliente	1	Sun, Jun 04 2006 @ 07:1			a_tipo_de_peca = Não	
peca	Cliente	1	Tue, Jul 04 2006 @ 10:1			a_tipo_de_peca = Não	
peca	Cliente	1	Tue, Jul 04 2006 @ 02:1			a_tipo_de_peca = Não	
peca	Cliente	1	Wed, Jul 05 2006 @ 07:1			a_tipo_de_peca = Não	
peca	Cliente	1	Wed, Jul 05 2006 @ 10:1			a_tipo_de_peca = Não	
peca	Cliente	1	Wed, Jul 05 2006 @ 12:1			a_tipo_de_peca = Não	
peca	Cliente	1	Fri, Jul 07 2006 @ 06:1			a_tipo_de_peca = Não	
peca	Cliente	1	Fri, Jul 07 2006 @ 02:1			a_tipo_de_peca = Não	
peca	Cliente	1	Mon, Jul 10 2006 @ 07:1			a_tipo_de_peca = Não	
peca	Cliente	1	Mon, Jul 10 2006 @ 10:1			a_tipo_de_peca = Não	
peca	Cliente	1	Tue, Jul 11 2006 @ 06:1			a_tipo_de_peca = Não	
peca	Cliente	1	Tue, Jul 11 2006 @ 08:1			a_tipo_de_peca = Não	
peca	Cliente	1	Tue, Jul 11 2006 @ 10:1			a_tipo_de_peca = Não	
peca	Cliente	1	Tue, Jul 11 2006 @ 01:1			a_tipo_de_peca = Não	
peca	Cliente	1	Wed, Jul 12 2006 @ 06:1			a_tipo_de_peca = Não	
peca	Cliente	1	Wed, Jul 12 2006 @ 06:1			a_tipo_de_peca = Não	

Figura 6.10 – Configuração das chegadas.

6. Processamento: Consiste em uma tabela onde são definidas as operações de cada entidade em cada local e o recurso necessário para esta operação, e uma tabela de rotas que define o destino e a movimentação de cada entidade, bem como o modo como se dá essa movimentação e o recursos necessários. Nesta etapa são inseridos os tempos de processamento de cada local de forma probabilística. Para isso foi utilizado o Statfit, um software que faz parte do pacote de simulação do ProModel. Os tempos de processamento foram cronometrados no objeto de estudo e inseridos no software para que se obtivesse as distribuições de probabilidade. Assim em cada replicação o modelo produz um número diferente de peças.

No caso estudado as configurações do processamento dizem respeito a inserção dos tempos de processamento, dos tipos de peças e seus roteiros e a lógica de incremento e decremento das quantidades de *kanbans* conforme o envio dos lotes aos clientes. A **Figura 6.11** apresenta a utilização do menu processamento para o caso estudado.

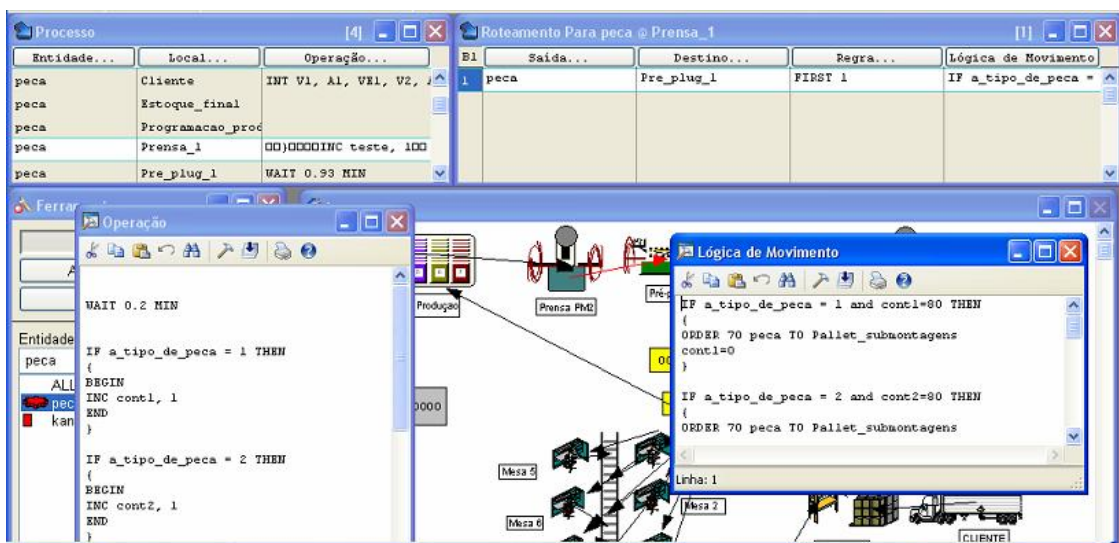


Figura 6.11 – Configuração do processamento.

7. Recursos: São os elementos usados para transportar entidades, executar operações, realizar manutenção nos locais ou outros, podem ser: pessoas ou equipamentos. Um sistema pode ter um ou mais recursos, sendo dotado de movimento ou não. Contudo, para cada recurso deve ter designado uma rede de caminhos, ou seja, um percurso na qual a movimentação se dará.

Toda a programação inserida no ProModel é mostrada no **Anexo 1**. Uma questão importante é a flexibilidade que o *software* permite na criação e utilização de elementos gráficos para locais, entidades e recursos, além da possibilidade do uso de elementos de fundo, o que pode tornar o modelo bastante representativo e de fácil entendimento.

6.4.6. Execução do modelo

Após a construção do modelo, conforme ilustrado no último tópico o modelo foi executado através do ProModel® e a próxima etapa tem início.

6.4.7. Verificação

Nesta etapa, foi verificada a consistência dos dados coletados para evitar que o modelo se torne inutilizado, já que de nada adianta um modelo que não reflita a realidade do sistema. Isso é feito após uma rodada e o a produção total diária e seqüência de operação são analisadas e comparadas com o modelo conceitual estabelecido inicialmente.

Se estas informações corresponderem ao planejado inicialmente e às informações fornecidas pela empresa, então se pode passar para a próxima etapa.

6.4.8. Validação:

Validar um modelo é avaliar o quanto ele se aproxima do sistema real, assegurando que o modelo serve para o propósito que foi criado. Buscou-se validar o modelo através da comparação do resultado do sistema real com o do modelo simulado e da confrontação direta destes resultados com dados históricos do sistema produtivo estudado, metodologias sugeridas por SARGENT (2004). A **Tabela 6.2** ilustra essa comparação.

Tabela 6.2: Comparação Real X Virtual

	Modelo real	Modelo Virtual	Variação
Média diária	549	522	5%
Takt time	55s	58s	5,2%
Locais	18	18	0%
Peças	7	7	0%

O valor do *Takt time* fornecido pela organização foco de estudo, definido como o ritmo de produção necessário a atender a demanda do cliente (SLACK et al, 2002), foi de 58 segundos. Esse tempo corresponde a produção de 549 peças por dia, justamente o valor da média de produção diária também fornecida. Com o objetivo de refinar ainda mais o processo de validação do modelo computacional construído buscou-se associar as médias de produção alcançadas e dessa forma calcular um *Takt time* para o modelo computacional construído, que foi de 55 segundos.

Outra forma de validação utilizada foi através das quantidades totais produzidas nos meses modelados de maio, junho e julho, os dados do mês de abril não foram disponibilizados pela empresa. A **Tabela 6.3** ilustra essa comparação.

Tabela 6.3 Comparação Produção real X Produção simulada

Produção total		
Meses	Real	Simulado
Maio	10357	10500
Junho	9616	8681
Julho	10643	8895

6.4.9. Planejamento dos experimentos

Nesta fase, sendo esta simulação terminante, é definido o tempo necessário para se obter resultados aceitáveis na simulação e o número de replicações necessárias para se obter uma amostragem estatística confiável.

O tempo de simulação para este estudo foi definido de acordo com os meses simulados de abril, maio, junho e julho, representando o funcionamento normal da linha de produção objeto de estudo e seu funcionamento a partir da puxada dos clientes, representando seu consumo.

Segundo o manual do ProModel o número de replicações deve ser determinado de acordo com a precisão requerida dos resultados a serem obtidos pela simulação. Se uma avaliação superficial de desempenho é planejada, de três a cinco replicações são suficientes. Entretanto, para uma

análise mais aprofundada mais replicações são necessárias. Por esse motivo, o número de replicações aplicadas a busca de resultados deste trabalho é de 5.

6.4.10. Realização e análise dos experimentos

Nesta fase são executadas as simulações e testadas as diferentes alternativas propostas para a melhoria do sistema. A **Figura 6.12** apresenta a tela do ProModel® onde foi executada a simulação.

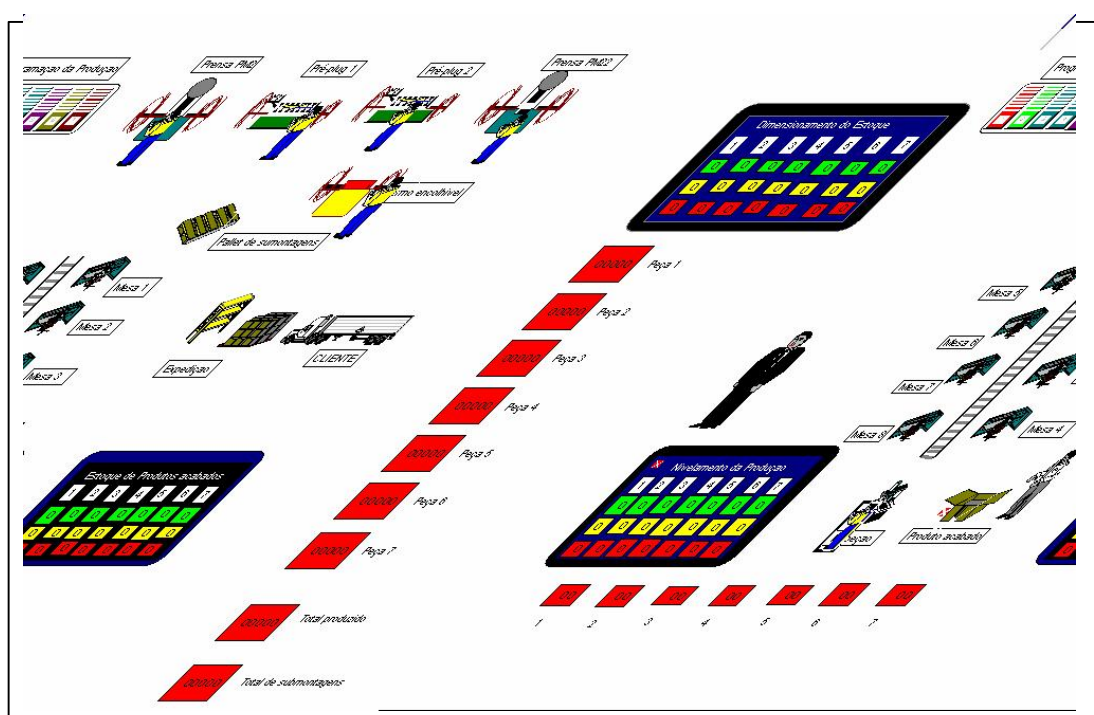


Figura 6.12 – Modelo computacional do objeto de estudo.

Uma vez obtidos os resultados da otimização, realizou-se a experimentação com os resultados obtidos e como resultado averiguou-se a validade dos resultados encontrados. A metodologia de otimização utilizada e seus resultados são descritos no tópico 6.5.

6.4.11. Replicações extras

Com base nas análises realizadas o modelador determina se outras replicações ou simulações serão necessárias assim como a utilização ou não de novos cenários. Tais atitudes não foram necessárias uma vez que os resultados puderam ser obtidos sem problemas.

6.4.12. Documentação e relato

A documentação é necessária por inúmeras razões. Se um modelo precisar ser utilizado novamente para a mesma análise ou outra, pode ser necessário lembrar como o modelo funciona. Isso dará mais confiabilidade ao modelo. Por um outro lado, se o modelo precisar ser modificado, através da documentação, essa tarefa se tornará menos árdua. Os resultados devem ser relatados de maneira clara e consciente. Permitindo ao cliente revisar a formulação final, as alternativas criadas, seus critérios de criação e acima de tudo, as recomendações do modelador/analista.

Este capítulo apresenta a descrição detalhada de como o modelo foi construído e como anexo teremos a documentação que o ProModel oferece, facilitando o acesso a informações relativas a construção do modelo caso seja necessário modifica-lo ou construí-lo novamente.

6.5. Otimização

Nesta etapa do estudo buscou-se otimizar as quantidades de *kanbans* na linha produtiva foco de estudo nos meses de abril, maio, junho e julho através do *software* SimRunner®, que usa Algoritmos Evolutivos como método de otimização. Para ter acesso ao SimRunner® a partir do ProModel®, basta dar um click no ícone específico na barra de ferramentas, conforme **Figura 6.13**:



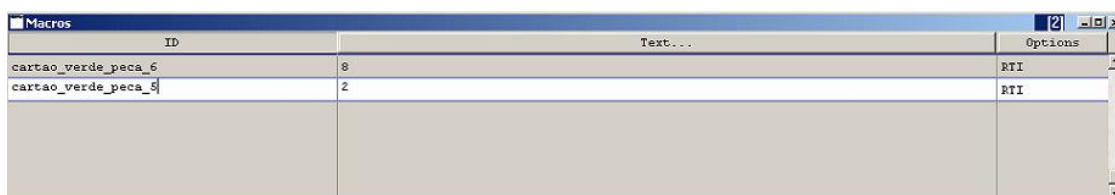
Figura 6.13 – Tela de acesso ao SimRunner® a partir do ProModel®.

Como na simulação, a realização de uma otimização segue uma metodologia. A metodologia utilizada nesta otimização é a definida por HARREL *et al.* (2000). Segundo o autor, após a construção e validação do modelo, alguns passos devem ser seguidos:

6.5.1. Definição das variáveis

Nesta fase definiu-se as variáveis que foram testadas pelo algoritmo de otimização, também chamadas de variáveis de decisão ou *inputs*, e que têm seu valor alterado a cada rodada simulação. As variáveis de decisão deste problema de otimização foram definidas como sendo todas as quantidades de *kanbans* verdes das peças modeladas nos meses de abril, maio, junho e julho.

Para que estas variáveis estejam disponíveis para escolha no *software* SimRunner® foi necessário que, no modelo, a quantidade de cada uma destas variáveis sejam definidas como uma macro, como mostra a **Figura 6.14**.



ID	Text...	Options
cartao_verde_peca_6	8	RTI
cartao_verde_peca_5	2	RTI

Figura 6.14 – Definição das Macros.

6.5.2. Definição dos tipos de variáveis

Após as variáveis serem definidas foi necessário definir o tipo de variável e seus limites para que o algoritmo de otimização gere soluções respeitando estas definições. No presente problema, as variáveis representam as quantidades de *kanbans*, assim elas devem ser do tipo inteiro. Na definição dos limites das variáveis, foi tomada a quantidade atual de cartões como limite máximo (uma vez que não se deseja utilizar quantidades maiores que estas) e uma única unidade como limite mínimo (planejou-se utilizar ao menos um cartão nas quantidades de estoque “no verde”). A **Figura 6.15** apresenta a definição das variáveis, seus tipos e limites.

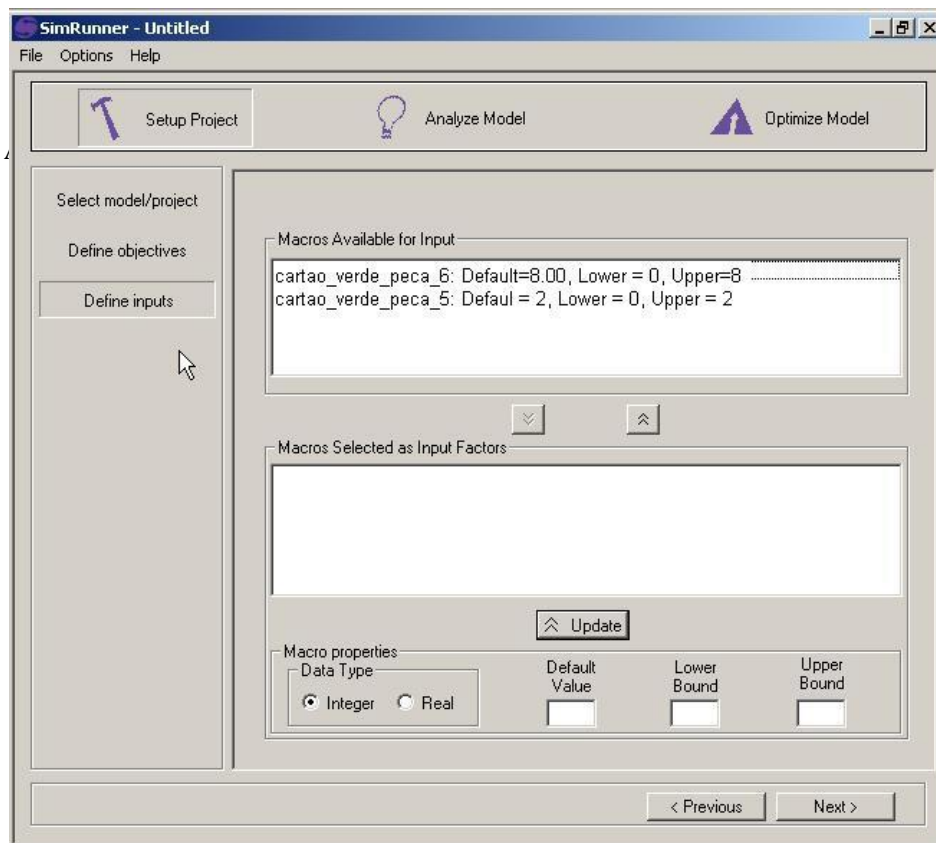


Figura 6.15 – Definição dos Fatores de entrada e seus limites.

6.5.3. Definição da função objetivo

A função objetivo, que algumas vezes já é estabelecida durante a fase de projeto do estudo de simulação, pode ser construída tendo por base peças, equipamentos, operários entre outros, buscando minimizar, maximizar ou fazer uso de ambos em diferentes variáveis, dando inclusive pesos diferentes para compor a função.

Como este trabalho busca a otimização da quantidade de cartões, que são representados por variáveis, a função objetivo foi definida como a minimização das quantidades de cartões nos meses de abril, maio, junho e julho das peças 5 e 6, uma vez que só estas possuíam quantidades em estoque passíveis de serem otimizadas, e as outras quantidades mínimas de cartões que não proporcionariam a utilização da otimização como exemplifica a **Figura 6.16**.

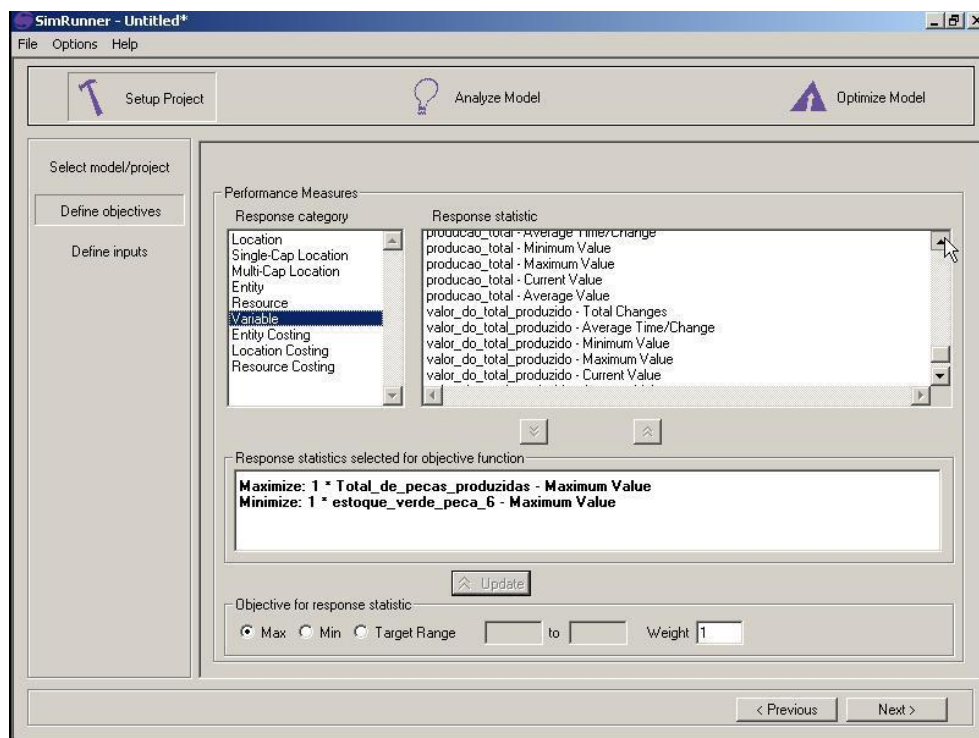


Figura 6.16 – Definição da função objetivo.

6.5.4. Seleção do tamanho da população do algoritmo evolutivo

O tamanho da população de soluções afeta a confiabilidade e o tempo necessário para a condução da uma busca, assim, é necessário um equilíbrio entre o tempo requerido e o resultado esperado.

No SimRunner® a seleção do tamanho da população é feita através da opção *setup options*, onde é possível escolher três tamanhos diferentes: *aggressive*, tem uma população pequena o que permite convergir para uma solução mais rapidamente, porém a confiabilidade de tal solução é menor; *cautious*, tem uma grande população, o que implica em maiores tempos de processamento, porém a confiabilidade da resposta encontrada é maior; *moderate*, apresenta um equilíbrio entre tempo de processamento e confiabilidade da resposta, sendo o tamanho de sua população um número intermediário entre o *aggressive* e *cautious*.

Nesta fase foi definida a precisão requerida na resposta do problema, o nível de significância e número de replicações. A precisão requerida neste problema foi de 1, uma vez que a resposta será dada em unidades. O nível de confiança adotado foi o nível padrão no *software*, 95%. Já o

número de replicações, adotou-se o mesmo usado na fase da simulação, ou seja, 10 replicações. A **Figura 6.17** ilustra esses passos.

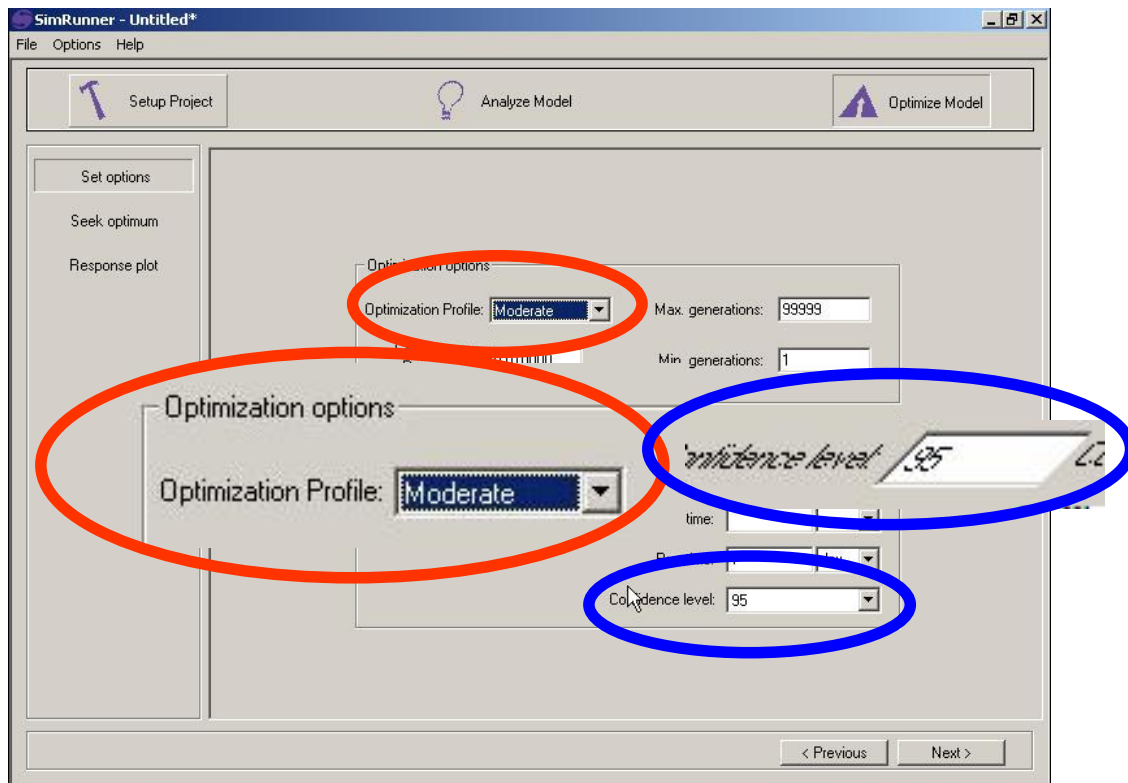


Figura 6.17 – Configuração do *setup*.

6.5.5. Análise dos resultados

O último passo da metodologia para o estudo de otimização é a análise dos dados obtidos através da busca efetuada pelo algoritmo otimizador.

A **Tabela 6.4** mostra as peças que tiveram suas quantidades de *kanbans* otimizadas e seus respectivos meses.

Tabela 6.4: Resultados da otimização em abril, maio, junho e julho.

Mês	Peça	Cartões Verdes	Redução	Total	% Reduzida
Abril	5	2	1	4	25 %
	6	11	4	22	18 %
Maio	5	2	1	4	25 %
	6	8	3	17	17 %
Junho	3	2	1	4	25 %
	5	2	1	4	25 %
	6	8	3	17	17 %
Julho	6	10	3	20	15 %

A **Tabela 6.4** mostra os itens que tiveram seus *kanbans* verdes otimizados e as porcentagens de redução levando-se em conta as quantidades de cartões no amarelo e no vermelho.

Entretanto cabe aqui a discussão de outros resultados que ocorreram em alguns meses, que foram casos de múltiplas soluções. Nesses casos, o relatório do SimRunner® retornou respostas diferentes com o mesmo valor da função objetivo.

Em tais casos foram realizadas simulações com cada uma das soluções que o SimRunner® retornou e alguns aspectos puderam ser observados: a medida em que se reduz as quantidades de cartões, os estoques no amarelo e no vermelho também passam a ser utilizados, nesses casos foram considerados como ótimos os valores que não utilizavam estas quantidades em estoque, ou seja os maiores valores retornados que juntamente com os demais possuíam o maior valor da função objetivo.

Neste ponto seria muito pertinente representar também a redução em termos financeiros e a economia total que seria proporcionada à organização foco de estudo no caso da aplicação das soluções encontradas. No entanto, tais dados não foram disponibilizados devido a questões de confidencialidade impostas pela empresa já citadas anteriormente.

6.6. Considerações finais

A combinação de simulação e otimização mostrou-se eficiente e de uso relativamente simples. Através da aplicação destas duas ferramentas podê-se reduzir a quantidade em estoque dos itens modelados em 20%. Assim, foi possível verificar na prática a utilidade das ferramentas abordadas, conferindo ao estudo um caráter científico.

Capítulo 7

CONCLUSÃO

7.1. Considerações iniciais

Este capítulo apresenta as conclusões obtidas e as recomendações para futuros trabalhos. Apresentando os resultados obtidos com a aplicação das ferramentas propostas no sistema produtivo foco do estudo, as contribuições fornecidas com a elaboração deste trabalho e propondo sugestões para futuros trabalhos envolvendo a gestão de custos e o aprimoramento de processos.

7.2. Conclusões e contribuições do trabalho

A simulação computacional é uma ferramenta que vem apresentando constantes evoluções devido, principalmente, aos avanços nas áreas de *hardware* e *software*, o que também aconteceu com a otimização, bem como a integração destas duas. Porém, no desenvolvimento do trabalho pode-se constatar que não basta ter os mais avançados recursos tecnológicos. É necessário que haja pessoas treinadas e preparadas, uma vez que todas estas ferramentas não substituem o homem na hora da decisão final.

A vantagem de se utilizar a simulação para a modelagem de um sistema puxado é a possibilidade de se realizar experimentações e se alcançar modelos que estejam em maior concordância com os fundamentos do sistema just in time. Sendo assim, pode-se estabelecer cenários e simulá-los realizando análises que só seriam possíveis na vida real mediante grande dispêndio de tempo e custo.

O uso da otimização pode ser decisivo, uma vez que o ProModel fornece um módulo de otimização fácil de ser utilizado, o SinRunner, que procura de maneira inteligente e eficaz a solução ótima para um dado problema tomando como base os resultados do modelo de simulação.

Para a execução da fase de simulação foi seguida a metodologia comum aos autores SHANNON, 1998, CARLSON, 2003, LAW e KELTON, 2000, PEREIRA, 2000, DUARTE, 2003 e SILVA, 2005, utilizando o *software* de simulação ProModel® , já na fase de otimização foi utilizado o SimRunner®.

A utilização dos conceitos de simulação associados a otimização na modelagem de um Sistema *kanban* se mostrou válida uma vez que resultados relevantes foram obtidos. Dados históricos foram utilizados com o intuito de validar esta metodologia e para trabalhos futuros, sugere-se a utilização de técnicas de previsão associadas a simulação e a otimização para que resultados futuros de médio e longo prazo possam ser alcançados.

Reduções médias de até 18% na quantidade de *kanbans* de cada peça modelada evidenciam o superdimensionamento das quantidades atuais em estoque do objeto de estudo.

Comprovando também a possibilidade de se gerar outros resultados como redução dos custos de armazenagem e preservação dos estoques, economia de espaço no almoxarifado, sem falar no custo de oportunidade, representando a possibilidade de se aplicar o excesso gasto em estoques em outros investimentos.

As principais contribuições deste trabalho estão na aplicação de uma metodologia de otimização para a redução do número de *kanbans* e nos resultados encontrados. A associação dos conceitos de simulação e otimização se mostrou válida pelo fato de propor a redução das quantidades em estoque da organização foco de estudo e como consequência reduzir custo. Caberia também neste estudo uma representação desta redução de custos, entretanto tais dados não foram disponibilizados ao pesquisador devido a questões de confidencialidade impostas pela instituição.

7.3 Sugestões para trabalhos futuros

Como sugestão para trabalhos futuros seguindo a mesma linha desta dissertação, propõe-se:

1. Modelagem de outros sistemas puxados e a utilização de técnicas de previsão associadas a simulação e a otimização para que resultados futuros de médio e longo prazo possam ser alcançados;

2. Modelagem de outros sistemas puxados e a aplicação da simulação para evidenciar os resultados obtidos e a partir de múltiplos estudos de caso, propor uma fórmula ótima para o cálculo do número de *kanbans*;
3. Modelar toda a rede de distribuição interna de matéria-prima da organização, que também ocorre por meio de *kanbans* e realizar sua otimização;
4. Modelar o mesmo sistema utilizando outro *software* de simulação de eventos discretos, fazendo um comparativo deste com o ProModel®;
5. Utilizar outro *software* de otimização que faz integração com o ProModel®, o OptQuest® da OptTek, e comparar os resultados obtidos com os dois programas;

7.4. Considerações finais

O presente trabalho buscou reduzir a quantidade de *kanbans* da organização foco de estudo através da associação entre os conceitos de simulação e otimização. As etapas seguidas para a concretização do estudo mostraram que a condução de um estudo de simulação e otimização, mesmo com a facilidade trazida com os novos programas, não é de fácil aplicação, devendo haver uma metodologia de implementação e uma sinergia entre usuário e modelador para que o mesmo entenda o processo a ser simulado, os dados necessários para a construção do modelo e os resultados esperados. Também deve ser ressaltado o poder que tem a simulação como ferramenta de análise de um sistema produtivo complexo onde existem diversas interações entre as suas variáveis, nem sempre claras ao entendimento. A vantagem de se utilizar a simulação para a modelagem de um sistema puxado é a possibilidade de se realizar experimentações e se alcançar modelos que estejam em maior concordância com os fundamentos do sistema *just in time*.

REFERÊNCIAS BIBLIOGRÁFICAS

AGUIAR, E.; COSTA, J. F. S. Uma implementação de pesquisa estocástica utilizando algoritmos evolutivos. Cadernos do IME, Rio de Janeiro, volume 1, 1997.

Andrade, G. J. P. Metodologia para a análise de viabilidade e implementação do sistema kanban interno em malharias pertencentes a uma cadeia produtiva têxtil. Dissertação de mestrado, Universidade Federal de Santa Catarina, 2000.

ANGELINE, P. J. Evolutionary algorithms and emergent intelligence. Dissertação de mestrado, The Ohio State University, 1994, USA.

ANJARD R. P., Process Mapping: One of three, new, special quality tools for management, quality and all other professionals. Microelectron. Reliab., volume. 36, 1995.

AZADIVAR F., Simulation optimization methodologies. Proceedings of the 1999 Winter Simulation Conference. Manhattan, USA.

BÄCK, Thomas; HAMMEL, Ulrich; SCHWEFEL, Hans-Paul. Evolutionary computation: comments on the history and current state. Evolutionary Computation, Abril, volume 1, 1997.

BANKS J., Introduction to Simulation. Proceeding of the Winter conference, Atlanta, 1999.

BANKS, J. Introduction to simulation. Proceedings of the Winter Simulation Conference. Atlanta, 2000.

BARCELLOS, João C.H., Algoritmos genéticos adaptativos: Um estudo comparativo. Dissertação de mestrado, Universidade de São Paulo, 2000.

BEASLEY, D., BULL, D. R. e MARTIN R. R. An Overview of Genetic Algorithms: Part I, Fundamentals. University Computing, volume 15, 1993.

Bertrand J.W.M., Fransoo J.C. Modelling and simulation: Operations management research methodologies using quantitative modeling. *International Journal of operations & Production Management*. Volume 22, 2002.

BOWDEN R.O. Simulation optimization research and development. *Proceedings of the 1998 Winter Simulation Conference*. Mississippi, USA.

Bratley Paul, Fox Bennet L., Schrage Linus E. *A Guide to Simulation*, Second Edition. New York, Springer-Verlag, 1987.

CARNEIRO, Lúcia F., Síntese automatizada de colunas de destilação: Uma abordagem alternativa ao processo de projeto. *Dissertação de mestrado*, Pontifícia Universidade Católica do Rio Grande do Sul, Portp Alegre, 1996.

CARSON, Y., MARIA, A. Simulation optimization: Methods and applications. *Proceedings of the 1997 Winter Simulation Conference*, NewYork.

Carson II John S. Introduction to modeling and simulation. *Proceedings of the 2003 Winter Simulation Conference*, Marietta, USA.

CASTRO, Herval P. Utilização de algoritmos genéticos para solução de problemas de produção de uma refinaria de petróleo. *Dissertação de mestrado*. Universidade Federal de Santa Catarina, 2001.

COELHO , Rodrigo Marcelo. Implantação e simulação do sistema KANBAN de movimentação de materiais: Estudo de caso na industria eletrônica do segmento EMS. *Dissertação de mestrado*, Universidade Federal de Minas Gerais, 2003.

CO H., JACOBSON S.H. The kanban assignment problem in serial just-in-time production systems. *IIE Transactions*, 1994.

CUNHA, Alexandre S.; PINTO, Ricardo L. U. S. Uma técnica para ajuste dos parâmetros de um Algoritmo Genético. *XXXIII Simpósio Brasileiro de Pesquisa Operacional*, Campos do Jordão, SP, 2001.

CHAMBERS, Lance. Practical Handbook of Genetic Algorithms - Applications Volume I , 1a Edição, 1995.

CHICK S., SÁNCHEZ P. J., FERRIN D., MORRICE D.J. A simulation-optimization approach using genetic search for supplier selection. Proceedings of the 2003 Winter Simulation Conference, Metz, France.

CLETO M. G. A Gestão da Produção nos últimos 45 anos. Revista FAE BUSINESS, nº 4, dezembro de 2002.

DIAS, Carlos R., Algoritmos evolutivos para o problema de clusterização de grafos orientados: Desenvolvimento e análise experimental. Dissertação de mestrado. Universidade Federal Fluminense, 2004.

DING H., BENYOUCEF L., XIE X. A simulation optimization approach using genetic search for supplier selection. Proceedings of the 2003 winter simulation conference.

DORNELLAS, C.R.R. Otimização dos despachos reativos utilizando algoritmos genéticos. Tese de Mestrado, COPPE – UFRJ, Rio de Janeiro. 1997.

DOLOI, H.; JAAFARI, A. Conceptual simulation model for strategic decision evaluation in project management. Logistics Information Management, V 15, N 2, 2003.

DUARTE, Roberto N. Simulação computacional: Análise de uma célula de manufatura em lotes do setor de auto-peças. Dissertação de mestrado em engenharia de produção. UNIFEI, Itajubá, MG, 2003.

ESPARRAGO, Romeo A. Jr. Kanban. Production and Inventory Management Journal; First Quarter , ABI/INFORM Global, 1988.

FOGEL, D. An Introduction to Simulated Evolutionary Optimization. IEEE. Transactions on Neural Networks. Volume 5, 1994.

FU, Michael C. Optimization for Simulation: Theory vs. Practice. Journal on Computing, volume 14, 2002.

FLETCHER R. Methods of Optimization, Volume 1. Unconstrained Optimization. Wiley,1980.

GAPSKI, Otavio L. Controle de nível de estoque no setor varejista com base no gerenciamento do inventário pelo fornecedor: aplicação do modelo no A. Angeloni cia Ltda e Procter & Gamble S.A. Dissertação de mestrado, Universidade Federal de Santa Catarina, Julho de 2003.

GAVIRA, Muriel O. Simulação computacional como uma ferramenta de aquisição de conhecimento. Dissertação de mestrado em engenharia de produção. USP, São Carlos, SP, 2003.

GOMES G. S. O Papel da Área de Planejamento e Controle da Produção na Integração entre Clientes e Fornecedores dentro de uma Cadeia de Suprimentos JIT: O caso da VW/Audi e um dos Seus Fornecedores JIT. Dissertação de Mestrado, Universidade Federal de Santa Catarina, 2003.

GOONATILAKE, Suran; TRELEAVEN, Philip. Intelligent Systems for Finance and Business. New York : John Wiley & Sons, 1995.

GROUMPOS P. P., MERKURYEV Y. A Methodology of Discrete-Event Simulation of Manufacturing Systems: An Overview. International Journal of SIMULATION Vol. 2 No. 1, 2003.

HARREL, C. R.; BATEMAN, R. E.; GOGG, T. J.; MOTT, J. R. A. System Improvement Using Simulation. Orem, Utah: ProModelL Corporation. 1996.

HARREL, Charles R.; GHOSH, Biman K.; BOWDEN, Royce. Simulation Using ProModel. McGraw-Hill, 2000.

HOLLAND, J. H. Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor, 1975.

HONDA Auro K., VIVEIRO Carlos T. Qualidade e Excelência Através da Metodologia Kaizen. Editora Érica LTDA, Tatuapé, SP,1993,.

HUTCHINS David, Just In Time. Editora Atlas S.A., São Paulo, 1992.

IZIDORO, Sandro C. Determinação do número de agrupamentos em conjunto de dados multidimensionais utilizando Algoritmos Genéticos. Dissertação de mestrado em engenharia elétrica. UNIFEI, Itajubá, MG, 2001.

JOHANSSON B., Discrete Event Simulation – present situation and future potential. Department of Product and Production Development, Chalmers University of Technology, Göteborg, Sweden 2002.

JOINES J.A., BARTON R.R., KANG K., FISHWICK P.A. Simulation and Optimization of Sugar Cane Transportation in Harvest Season. Proceedings of the 2000 Winter Simulation conference, USA.

JÚNIOR M. K. Planejamento e Acompanhamento Logístico – Industrial como Diferencial Competitivo na Cadeia de Logística Integrada. Tese de doutorado, Universidade Federal de Santa Catarina, 2004.

KÖCHEL, P., and NIELÄNDER, U., 2000, Evolutionary Optimisation of Kanbans. INFORMS-KORMS, 2000.

LABAS C.A. ALTIPARMAK F., DENGIZ B., The Optimization of Number of Kanbans with Genetic Algorithms, Simulated Annealing and Tabu Search. Gazi University, Department of Industrial Engineering, 2000.

LAW Averill M. How to conduct a successful simulation study. Proceedings of the 2003 Winter Simulation Conference, 2003.

LAW Averill M., McComas Michael G. Simulation of manufacturing systems. Proceedings of the 1999 Winter Simulation Conference, Tucson, USA.

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

LEE Young H., KIM Sook H. Optimal production-distribution planning in supply chain management using hybrid simulation-analytic approach. Proceedings of the 2000 Winter Simulation Conference. Ansan, South Korea.

LUBBEN R. T. Just in Time: Uma Estratégia Avançada de Produção, Editora McGraw-Hill, 2ª Edição, São Paulo, 1989.

MARTIM A.D., CHANG T.M. KINCAID Y.R.K. Using tabu search to Determine the number of kanbans and lotsizes in a generic kanban system. Annals of Operations Research, VOL 78, pages, 201 – 217, 1998.

O'FERREL D.S., Manufacturing modeling and optimization. IEE, Semi advanced Semiconductor Manufacturing conference, 1995.

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

OLIVEIRA, Francisco Alexandre de. A Gestão Baseada em Atividade aplicada em ambientes celulares: uma abordagem metodológica. Dissertação de mestrado em engenharia de produção. UNIFEI, Itajubá, MG, 2003.

O'KANE, J.F. SPENCELEY, J.R. TAYLOR, R. Simulation as an essential tool for advanced manufacturing technology problems. Journal of Materials Processing Technology, 107, pp. 412-424, 2000.

LAW, M. A. MCCOMAS, M. G. Simulation of Manufacturing Systems. Proceedings of the Winter Simulation Conference. Tucson, 1999.

OPTQUEST FOR ARENA USER'S GUIDE, Rockwell Software Inc., 2002.

PEINADO, J. O Papel do Sistema de Abastecimento Kanban na Redução dos Inventários. Revista FAE, pág. 27-34, Volume 2, Nº 2, maio/agosto, 1999, Curitiba.

PEINADO, J. Implantação do kanban como base de um programa Just in Time: Uma proposta de metodologia para empresas industriais. Dissertação de mestrado, Universidade Federal de Santa Catarina, 2000.

PEREIRA, I. C. Proposta de sistematização da simulação para fabricação em lotes. Dissertação de mestrado em engenharia de produção. UNIFEI, Itajubá, MG, 2000.

PINTO, Orlando P.F.J. Simulação e otimização: Desenvolvimento de uma ferramenta de análise de decisão para suprimento de refinarias de petróleo através de uma rede de oleodutos. Dissertação de Mestrado. Universidade Federal de Santa Catarina, Florianópolis, 2001.

PRICE, Rochelle N.; HARREL, Charles R. Simulation modeling and optimization using ProModel. Proceedings of the Winter Simulation Conference. Phoenix, Arizona, Estados Unidos, 1999.

ProModel USER'S GUIDE. ProModel Corporation, 2002.

PROTIL, R. M. Otimização do Processo Decisório Utilizando Simulação Computacional. XXXIII Simpósio Brasileiro de Pesquisa Operacional, Campos do Jordão, SP, p. 1535- 1546, 2001.

RAO S. S. Optimization: Theory and Applications. Wiley Eastern Limited, India, 1979.

REINALDO A. KANBAN: A Simplicidade do Controle de Produção. Instituto de Movimentação e Montagem de Materiais, IMAM, São Paulo, 2000.

ROGERS P. Optimum-seeking simulation in the design and control of manufacturing systems: experience with optquest for ARENA. Proceedings of the 2002 Winter Simulation Conference. Calgary, Canadá.

SHANNON R. E., Introduction to the art and science of simulation. Proceedings of the Winter Simulation Conference, 1998.

SHINGO, Shigeo. O Sistema Toyota de Produção: Do Ponto de Vista da Engenharia de Produção. Editora BOOKMAN COMPANHIA EDITORA LTDA, 1996.

SILVA, Wesley A. Otimização de parâmetros da gestão baseada em atividades aplicada em uma célula de manufatura. Dissertação de mestrado, Universidade Federal de Itajubá, 2005.

SILVA E. L. MENEZES E. M., Metodologia de Pesquisa e Elaboração de Dissertação, Florianópolis 2005.

SIMÕES, Anabela B. Transposição: estudo de um novo operador genético inspirado biologicamente. Dissertação de mestrado em engenharia de informática. Universidade de Coimbra, Coimbra, Portugal, 1999.

SLACK N., CHAMBERS, S., HARRISON A., Administração da Produção. Editora Atlas, segunda edição, 2002.

SOARES, Antônio Carlos S. Qualidade: uma estratégia de competitividade industrial uma análise na indústria sul brasileira. Florianópolis: UFSC, 1999, 109 p. Dissertação de mestrado em engenharia de produção.

STAMM, Harro. Simulação industrial: uma avaliação de sua utilização no sudeste e sul do Brasil. Dissertação de mestrado em engenharia de produção. UFSC, Florianópolis, SC, 1998.

STRACK, Jair. GPSS: modelagem e simulação de sistemas. Rio de Janeiro: LTC, 1984.

TANOMARU, J. Motivação, fundamentos e Aplicações de Algoritmos Genéticos. II Congresso Brasileiro de Redes Neurais. Curitiba, PR, 1995.

TAVARES, José A. R. Geração de Configurações de Sistemas Industriais com o Recurso à Tecnologia das Restrições e Computação Evolucionária. Tese de doutorado em Informática. Universidade de Minho, Braga, Portugal, 2000.

TAVEIRA, Ricardo Ayer. Uma metodologia para aperfeiçoamento da mudança para um sistema de produção Just-in-Time em uma indústria Metalúrgica, usando simulação discreta e técnicas de projeto de experimentos de Taguchi. Dissertação de mestrado em engenharia de produção. UFSC, Florianópolis, SC, 1997.

THEISEN, R.M. Sistemática de análise e identificação de perdas operacionais em processos logísticos: um estudo de caso na empresa via LOG. Dissertação (Mestrado em Engenharia de Produção) Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal do Rio Grande do Sul, 2004.

W. Skinner. Manufacturing - the missing link in corporate strategy. Harvard Business Review, 1969.

TRUONG T.T., AZADIVAR F. Simulation Based Optimization for Supply Chain Configuration Design. Proceedings Of The 2003 Winter Simulation Conference, USA.

ANEXO 1

Programação do modelo computacional no ProModel

```

*****
**
*                               Formatted Listing of Model:
*
*                               C:\Documents and Settings\Bruno\Desktop\DOE\model.MOD
*
*****
**
Time Units:                      Minutes
Distance Units:                   Meters
Initialization Logic:
*****
**
*                               Locations
*
*****
**
Name                               Cap          Units  Stats          Rules          Cost
-----
-
Programacao_producao  INFINITE  1      Time Series  Oldest, FIFO,
Prensa_1              1         1      Time Series  Oldest, , First
Pre_plug_1            1         1      Time Series  Oldest, ,
Pre_plug_2            1         1      Time Series  Oldest, ,
Prensa_2              1         1      Time Series  Oldest, ,
Termo_encolhivel     1         1      Time Series  Oldest, ,
Pallet_submontagens  100       1      Time Series  Oldest, , First
Esteira              6         1      Time Series  Last Loc, FIFO,
Mesa_1               1         1      Time Series  Oldest, ,
Mesa_2               1         1      Time Series  Oldest, ,
Mesa_3               1         1      Time Series  Oldest, ,
Mesa_4               1         1      Time Series  Oldest, ,
Mesa_5               1         1      Time Series  Oldest, ,
Mesa_6               1         1      Time Series  Oldest, ,
Mesa_7               1         1      Time Series  Oldest, ,
Mesa_8               1         1      Time Series  Oldest, ,
INspecao             1         1      Time Series  Oldest, , First
Caixa_aberta         150       1      Time Series  Oldest, , First
Cliente              INFINITE  1      Time Series  Oldest, ,
Estoque_final        INFINITE  1      Time Series  Oldest, ,
Prateleira           INFINITE  1      Time Series  Oldest, ,
*****
**
*                               Entities
*
*****
**
Name          Speed (mpm)  Stats          Cost
-----
peca          50           Time Series
kanban        50           Time Series

```

```

*****
**
*
*                               Path Networks
*
*****
**
Name      Type      T/S      From      To      BI      Dist/Time
Speed Factor
-----
--
caminho  Passing      Speed & Distance N1      N2      Bi      55.67      1
*****
**
*
*                               Processing
*
*****
**
*
*                               Process
Routing
Entity    Location      Operation      Blk  Output      Destination
Rule      Move Logic
-----
-- peca    Cliente      INT V1, A1, VE1, V2, A2, VE2, V3, A3, VE3,
V4, A4, VE4, V5, A5, VE5, V6, A6, VE6, V7, A7, VE7, NV1, NA1, NVE1, NV2, NA2,
NVE2, NV3, NA3, NVE3, NV4, NA4, NVE4, NV5, NA5, NVE5, NV6, NA6, NVE6, NV7,
NA7, NVE7, KV1, KA1, KVE1, KV2, KA2, KVE2, KV3, KA3, KVE3, KV4, KA4, KVE4,
KV5, KA5, KVE5, KV6, KA6, KVE6, KV7, KA7, KVE7
V1=dimensionam_verde_peca_1
A1=dimensionam_amarelo_peca_1
VE1=dimensionam_vermelho_peca_1
V2=dimensionam_verde_peca_2
A2=dimensionam_amarelo_peca_2
VE2=dimensionam_vermelho_peca_2
V3=dimensionam_verde_peca_3
A3=dimensionam_amarelo_peca_3
VE3=dimensionam_vermelho_peca_3
V4=dimensionam_verde_peca_4
A4=dimensionam_amarelo_peca_4
VE4=dimensionam_vermelho_peca_4
V5=dimensionam_verde_peca_5
A5=dimensionam_amarelo_peca_5
VE5=dimensionam_vermelho_peca_5
V6=dimensionam_verde_peca_6
A6=dimensionam_amarelo_peca_6
VE6=dimensionam_vermelho_peca_6
V7=dimensionam_verde_peca_7
A7=dimensionam_amarelo_peca_7
VE7=dimensionam_vermelho_peca_7
V1=dimensionam_verde_peca_1
A1=dimensionam_amarelo_peca_1
VE1=dimensionam_vermelho_peca_1
V2=dimensionam_verde_peca_2
A2=dimensionam_amarelo_peca_2
VE2=dimensionam_vermelho_peca_2
V3=dimensionam_verde_peca_3

```

```
A3=dimensionam_amarelo_peca_3
VE3=dimensionam_vermelho_peca_3
V4=dimensionam_verde_peca_4
A4=dimensionam_amarelo_peca_4
VE4=dimensionam_vermelho_peca_4
V5=dimensionam_verde_peca_5
A5=dimensionam_amarelo_peca_5
VE5=dimensionam_vermelho_peca_5
V6=dimensionam_verde_peca_6
A6=dimensionam_amarelo_peca_6
VE6=dimensionam_vermelho_peca_6
V7=dimensionam_verde_peca_7
A7=dimensionam_amarelo_peca_7
VE7=dimensionam_vermelho_peca_7
NV1 = estoque_verde_peca_1
NA1 = estoque_amarelo_peca_1
NVE1 = estoque_vermelho_peca_1
NV2 = estoque_verde_peca_2
NA2 = estoque_amarelo_peca_2
NVE2 = estoque_vermelho_peca_2
NV3 = estoque_verde_peca_3
NA3 = estoque_amarelo_peca_3
NVE3 = estoque_vermelho_peca_3
NV4 = estoque_verde_peca_4
NA4 = estoque_amarelo_peca_4
NVE4 = estoque_vermelho_peca_4
NV5 = estoque_verde_peca_5
NA5 = estoque_amarelo_peca_5
NVE5 = estoque_vermelho_peca_5
NV6 = estoque_verde_peca_6
NA6 = estoque_amarelo_peca_6
NVE6 = estoque_vermelho_peca_6
NV7 = estoque_verde_peca_7
NA7 = estoque_amarelo_peca_7
NVE7 = estoque_vermelho_peca_7
KV1=V1-NV1
KA1 = A1 - NA1
KVE1 = VE1 - NVE1
KV2 = V2 - NV2
KA2 = A2 - NA2
KVE2 = VE2 - NVE2
KV3 = V3 - NV3
KA3 = A3 - NA3
KVE3 = VE3 - NVE3
KV4 = V4 - NV4
KA4 = A4 - NA4
KVE4 = VE4 - NVE4
KV5 = V5 - NV5
KA5 = A5 - NA5
KVE5 = VE5 - NVE5
KV6 = V6 - NV6
KA6 = A6 - NA6
KVE6 = VE6 - NVE6
KV7 = V7 - NV7
KA7 = A7 - NA7
KVE7 = VE7 - NVE7
```

```

IF a_tipo_de_peca = 1 and
estoque_verde_peca_1>0 and estoque_verde_peca_1<=V1 THEN
{
BEGIN
DEC estoque_verde_peca_1, 1
INC kanban_verde_peca_1,1
END
}
ELSE
{
IF a_tipo_de_peca = 1 and
estoque_verde_peca_1=0 and estoque_amarelo_peca_1>0 and
estoque_amarelo_peca_1<=A1 THEN
{
BEGIN
DEC estoque_amarelo_peca_1, 1
INC kanban_amarelo_peca_1,1
END
}
ELSE
{
IF a_tipo_de_peca = 1 and
estoque_amarelo_peca_1=0 and estoque_vermelho_peca_1>0 and
estoque_vermelho_peca_1<=VE1 THEN
{
BEGIN
DEC estoque_vermelho_peca_1, 1
INC kanban_vermelho_peca_1,1
END
}
}
}
IF a_tipo_de_peca = 2 and
estoque_verde_peca_2>0 and estoque_verde_peca_2<=V2 THEN
{
BEGIN
DEC estoque_verde_peca_2, 1
INC kanban_verde_peca_2, 1
END
}
ELSE
{
IF a_tipo_de_peca = 2 and
estoque_verde_peca_2=0 and estoque_amarelo_peca_2>0 and
estoque_amarelo_peca_2<=A2 THEN
{
BEGIN
DEC estoque_amarelo_peca_2, 1
INC kanban_amarelo_peca_2, 1
END
}
ELSE
{
IF a_tipo_de_peca = 2 and
estoque_amarelo_peca_2=0 and estoque_vermelho_peca_2<>0 and
estoque_vermelho_peca_2<=VE2 THEN
{

```

```

BEGIN
    DEC estoque_vermelho_peca_2, 1
    INC kanban_vermelho_peca_2, 1
END
}
}
IF a_tipo_de_peca = 3 and
estoque_verde_peca_3>0 and estoque_verde_peca_3<=V3 THEN
{
BEGIN
    DEC estoque_verde_peca_3, 1
    INC kanban_verde_peca_3, 1
END
}
ELSE
{
IF a_tipo_de_peca = 3 and
estoque_verde_peca_3=0 and estoque_amarelo_peca_3>0 and
estoque_amarelo_peca_3<=A3 THEN
{
BEGIN
    DEC estoque_amarelo_peca_3, 1
    INC kanban_amarelo_peca_3, 1
END
}

ELSE
{
IF a_tipo_de_peca = 3 and
estoque_amarelo_peca_3=0 and estoque_vermelho_peca_3>0 and
estoque_vermelho_peca_3<=VE3 THEN
{
BEGIN
    DEC estoque_vermelho_peca_3, 1
    INC kanban_vermelho_peca_3, 1
END
}
}
}
IF a_tipo_de_peca = 4 and
estoque_verde_peca_4>0 and estoque_verde_peca_4<=V4 THEN
{
BEGIN
    DEC estoque_verde_peca_4, 1
    INC kanban_verde_peca_4, 1
END
}
ELSE
{
IF a_tipo_de_peca = 4 and
estoque_verde_peca_4=0 and estoque_amarelo_peca_4>0 and
estoque_amarelo_peca_4<=A4 THEN
{
BEGIN
    DEC estoque_amarelo_peca_4, 1
    INC kanban_amarelo_peca_4, 1
}
}
}
}
}

```

```

        END
    }

    ELSE
    {
        IF a_tipo_de_peca = 4 and
estoque_amarelo_peca_4=0 and estoque_vermelho_peca_4>0 and
estoque_vermelho_peca_4<=VE4 THEN
        {
            BEGIN
                DEC estoque_vermelho_peca_4, 1
                INC kanban_vermelho_peca_4, 1
            END
        }
    }
        IF a_tipo_de_peca = 5 and
estoque_verde_peca_5>0 and estoque_verde_peca_5<=V5 THEN
        {
            BEGIN
                DEC estoque_verde_peca_5, 1
                INC kanban_verde_peca_5, 1
            END
        }
    ELSE
    {
        IF a_tipo_de_peca = 5 and
estoque_verde_peca_5=0 and estoque_amarelo_peca_5>0 and
estoque_amarelo_peca_5<=A5 THEN
        {
            BEGIN
                DEC estoque_amarelo_peca_5, 1
                INC kanban_amarelo_peca_5, 1
            END
        }
    ELSE
    {
        IF a_tipo_de_peca = 5 and
estoque_amarelo_peca_5=0 and estoque_vermelho_peca_5>0 and
estoque_vermelho_peca_5<=VE5 THEN
        {
            BEGIN
                DEC estoque_vermelho_peca_5, 1
                INC kanban_vermelho_peca_5, 1
            END
        }
    }
        IF a_tipo_de_peca = 6 and
estoque_verde_peca_6>0 and estoque_verde_peca_6<=V6 THEN
        {
            BEGIN
                DEC estoque_verde_peca_6, 1
                INC kanban_verde_peca_6, 1
            END
        }
    }

```

```

ELSE
{
IF a_tipo_de_peca = 6 and
estoque_verde_peca_6=0 and estoque_amarelo_peca_6>0 and
estoque_amarelo_peca_6<=A6 THEN
{
BEGIN
DEC estoque_amarelo_peca_6, 1
INC kanban_amarelo_peca_6, 1
END
}
ELSE
{
IF a_tipo_de_peca = 6 and
estoque_amarelo_peca_6=0 and estoque_vermelho_peca_6>0 and
estoque_vermelho_peca_6<=VE6 THEN
{
BEGIN
DEC estoque_vermelho_peca_6, 1
INC kanban_vermelho_peca_6, 1
END
}
}
}

IF a_tipo_de_peca = 7 and
estoque_verde_peca_7>0 and estoque_verde_peca_7<=V7 THEN
{
BEGIN
DEC estoque_verde_peca_7, 1
INC kanban_verde_peca_7, 1
END
}
ELSE
{
IF a_tipo_de_peca = 7 and
estoque_verde_peca_7=0 and estoque_amarelo_peca_7>0 and
estoque_amarelo_peca_7<=A7 THEN
{
BEGIN
DEC estoque_amarelo_peca_7, 1
INC kanban_amarelo_peca_7, 1
END
}
ELSE
{
IF a_tipo_de_peca = 7 and
estoque_amarelo_peca_7=0 and estoque_vermelho_peca_7>0 and
estoque_vermelho_peca_7<=VE7 THEN
{
BEGIN
DEC estoque_vermelho_peca_7, 1
INC kanban_vermelho_peca_7, 1
END
}
}
}
}

```

```

Estoque_final          FIRST 1          1   peca
peca      Estoque_final

Programacao_producao  FIRST 1          1   peca
peca      Programacao_producao

FIRST 80              1   peca      Prensa_1
peca      Prensa_1

WAIT 0.2 MIN
IF a_tipo_de_peca = 1 THEN
{
BEGIN
INC cont1, 1
END
}
IF a_tipo_de_peca = 2 THEN
{
BEGIN
INC cont2, 1
END
}
IF a_tipo_de_peca = 3 THEN
{
BEGIN
INC cont3, 1
END
}

IF a_tipo_de_peca = 4 THEN
{
BEGIN
INC cont4, 1
END
}
IF a_tipo_de_peca = 5 THEN
{
BEGIN
INC cont5, 1
END
}
IF a_tipo_de_peca = 6 THEN
{
BEGIN
INC cont6, 1
END
}
IF a_tipo_de_peca = 7 THEN
{
BEGIN
INC cont7, 1
END
}

FIRST 1              1   peca      Pre_plug_1
IF a_tipo_de_peca = 1 and cont1=80 THEN

```



```

{
ORDER 70 peca TO Pallet_submontagens
cont1=0
}
IF a_tipo_de_pecas = 2 and cont2=80 THEN
{
ORDER 70 peca TO Pallet_submontagens
cont2=0
}
IF a_tipo_de_pecas = 3 and cont3=80 THEN
{
ORDER 70 peca TO Pallet_submontagens
cont3=0
}
IF a_tipo_de_pecas = 4 and cont4=80 THEN
{
ORDER 70 peca TO Pallet_submontagens
cont4=0
}
IF a_tipo_de_pecas = 5 and cont5=80 THEN
{
ORDER 70 peca TO Pallet_submontagens
cont5=0
}
IF a_tipo_de_pecas = 6 and cont6=80 THEN
{
ORDER 70 peca TO Pallet_submontagens
cont6=0
}
}
IF a_tipo_de_pecas = 7 and cont7=80 THEN
{
ORDER 70 peca TO Pallet_submontagens
cont7=0
}
}
peca      Pre_plug_1          WAIT 0.8 MIN          1      peca      Pre_plug_2
FIRST 1
peca      Pre_plug_2          WAIT 0.8 MIN          1      peca      Prensa_2
FIRST 1
peca      Prensa_2           WAIT 0.2 MIN          1      peca
Termo_encolhivel      FIRST 1
peca      Termo_encolhivel    WAIT 0.2 MIN          1      peca
Pallet_submontagens  IF a_tipo_de_pecas, 1
peca      Pallet_submontagens  INC wip, 1
                                           INC buffer, 1
                                           IF a_tipo_de_pecas = 1 THEN
                                           {
                                           BEGIN
                                           GRAPHIC 1
                                           END
                                           }
                                           IF a_tipo_de_pecas = 2 THEN
                                           {
                                           BEGIN
                                           GRAPHIC 2

```

```

END
}
IF a_tipo_de_peca = 3 THEN
{
BEGIN
GRAPHIC 3
END
}
IF a_tipo_de_peca = 4 THEN
{
BEGIN
  GRAPHIC 4
END
}
IF a_tipo_de_peca = 5 THEN
{
BEGIN
GRAPHIC 5
END
}
IF a_tipo_de_peca = 6 THEN
{
BEGIN
  GRAPHIC 6
END
}
IF a_tipo_de_peca = 7 THEN
{
BEGIN
  GRAPHIC 7
END
}
1      peca      Mesa_1
FIRST 1
peca      Mesa_2
FIRST
peca      Mesa_3
FIRST
peca      Mesa_4
FIRST
peca      Mesa_5
FIRST
peca      Mesa_6
FIRST
peca      Mesa_7
FIRST
peca      Mesa_8
FIRST
peca      Mesa_1      WAIT N(7.3, 0.66) MIN
FIRST 1
peca      Mesa_2      WAIT N(7.3, 0.66) MIN
FIRST 1
peca      Mesa_3      WAIT N(7.3, 0.66) MIN
FIRST 1
peca      Mesa_4      WAIT N(7.3, 0.66) MIN
FIRST 1
peca      Mesa_1      1 peca Esteira
FIRST 1
peca      Mesa_2      1 peca Esteira
FIRST 1
peca      Mesa_3      1 peca Esteira
FIRST 1
peca      Mesa_4      1 peca Esteira

```

```

                                1   peca   Esteira
FIRST 1
peca   Mesa_5   WAIT N(7.3, 0.66) MIN
                                1   peca   Esteira
FIRST 1
peca   Mesa_6   WAIT N(7.3, 0.66) MIN
                                1   peca   Esteira
FIRST 1
peca   Mesa_7   WAIT N(7.3, 0.66) MIN
                                1   peca   Esteira
FIRST 1
peca   Mesa_8   WAIT N(7.3, 0.66) MIN
                                1   peca   Esteira
FIRST 1
peca   Esteira   INC aVar8, 1
                                MOVE                                1   peca   INspecao
FIRST 1
peca   INspecao   WAIT N(0.648, 0.05514) MIN
                                INC Total_de_pecas_produzidas, 1
                                IF a_tipo_de_pecas = 1 THEN
                                {
                                INC opcao_1, 1
                                }
                                IF a_tipo_de_pecas = 2 THEN
                                {
                                INC opcao_2, 1
                                }
                                IF a_tipo_de_pecas = 3 THEN
                                {
                                INC opcao_3, 1
                                }
                                IF a_tipo_de_pecas = 4 THEN
                                {
                                INC opcao_4, 1
                                }
                                IF a_tipo_de_pecas = 5 THEN
                                {
                                INC opcao_5, 1
                                }
                                IF a_tipo_de_pecas = 6 THEN
                                {
                                INC opcao_6, 1
                                }
                                IF a_tipo_de_pecas = 7 THEN
                                {
                                INC opcao_7, 1
                                }
                                1   peca   Caixa_aberta   FULL 1
peca   Caixa_aberta   COMBINE 150
                                1   kanban   Prateleira
FIRST 1
{
BEGIN
GRAPHIC 1
MOVE FOR 5 MIN
INC Var1, 1
opcao_1=0

```

```
END
}
IF a_tipo_de_peca = 2 THEN
{
BEGIN
GRAPHIC 2
MOVE FOR 5 MIN
INC Var2, 1
opcao_2=0
END
}
IF a_tipo_de_peca = 3 THEN
{
BEGIN
GRAPHIC 3
MOVE FOR 5 MIN
INC Var3, 1
opcao_3=0
END
}
IF a_tipo_de_peca = 4 THEN
{
BEGIN
GRAPHIC 4
MOVE FOR 5 MIN
INC Var4, 1
opcao_4=0
END

}
IF a_tipo_de_peca = 5 THEN
{
BEGIN
GRAPHIC 5
MOVE FOR 5 MIN
INC Var5, 1
opcao_5=0
END
}
IF a_tipo_de_peca = 6 THEN
{
BEGIN
GRAPHIC 6
MOVE FOR 5 MIN
INC Var6, 1
opcao_6=0
END
}
IF a_tipo_de_peca = 7 THEN
{
BEGIN
GRAPHIC 7
MOVE FOR 5 MIN
INC Var7, 1
opcao_7=0
END
}
}
```

```
kanban Prateleira      INT V1, A1, VE1, V2, A2, VE2, V3, A3, VE3, V4,
A4, VE4, V5, A5, VE5, V6, A6, VE6, V7, A7, VE7
```

```
V1=dimensionam_verde_peca_1
A1=dimensionam_amarelo_peca_1
VE1=dimensionam_vermelho_peca_1
V2=dimensionam_verde_peca_2
A2=dimensionam_amarelo_peca_2
VE2=dimensionam_vermelho_peca_2
V3=dimensionam_verde_peca_3
A3=dimensionam_amarelo_peca_3
VE3=dimensionam_vermelho_peca_3
V4=dimensionam_verde_peca_4
A4=dimensionam_amarelo_peca_4
VE4=dimensionam_vermelho_peca_4
V5=dimensionam_verde_peca_5
A5=dimensionam_amarelo_peca_5
VE5=dimensionam_vermelho_peca_5
V6=dimensionam_verde_peca_6
A6=dimensionam_amarelo_peca_6
VE6=dimensionam_vermelho_peca_6
V7=dimensionam_verde_peca_7
A7=dimensionam_amarelo_peca_7
VE7=dimensionam_vermelho_peca_7
```

```
IF a_tipo_de_peca = 1 and
estoque_vermelho_peca_1=VE1 and estoque_amarelo_peca_1<A1 THEN
{
BEGIN
DEC kanban_amarelo_peca_1, 1
INC estoque_amarelo_peca_1, 1
END
}
IF a_tipo_de_peca = 2 and
estoque_vermelho_peca_2<VE2 THEN
{
BEGIN
DEC kanban_vermelho_peca_2, 1
INC estoque_vermelho_peca_2, 1
END
}
ELSE
IF a_tipo_de_peca = 2 and
estoque_vermelho_peca_2=VE2 and estoque_amarelo_peca_2<A2 THEN
{
BEGIN
DEC kanban_amarelo_peca_2, 1
INC estoque_amarelo_peca_2, 1
END
}
ELSE
IF a_tipo_de_peca = 2 and
estoque_vermelho_peca_2=VE2 and estoque_amarelo_peca_2=A2 THEN
```

```

        {
        BEGIN
        DEC kanban_verde_peca_2, 1
        INC estoque_verde_peca_2, 1
        END
        }
        IF a_tipo_de_peca = 3 and
estoque_vermelho_peca_3<VE3 THEN
        {
        BEGIN
        DEC kanban_vermelho_peca_3, 1
        INC estoque_vermelho_peca_3, 1
        END
        }
        ELSE
        IF a_tipo_de_peca = 3 and
estoque_vermelho_peca_3=VE3 and estoque_amarelo_peca_3<A3 THEN
        {
        BEGIN
        DEC kanban_amarelo_peca_3, 1
        INC estoque_amarelo_peca_3, 1
        END
        }
        ELSE
        IF a_tipo_de_peca = 3 and
estoque_vermelho_peca_3=VE3 and estoque_amarelo_peca_3=A3 THEN
        {
        BEGIN
        DEC kanban_verde_peca_3, 1
        INC estoque_verde_peca_3, 1
        END
        }
        IF a_tipo_de_peca = 4 and
estoque_vermelho_peca_4<VE4 THEN
        {
        BEGIN
        DEC kanban_vermelho_peca_4, 1
        INC estoque_vermelho_peca_4, 1
        END
        }
        ELSE
        IF a_tipo_de_peca = 4 and
estoque_vermelho_peca_4=VE4 and estoque_amarelo_peca_4<A4 THEN
        {
        BEGIN
        DEC kanban_amarelo_peca_4, 1
        INC estoque_amarelo_peca_4, 1
        END
        }
        ELSE
        IF a_tipo_de_peca = 4 and
estoque_vermelho_peca_4=VE4 and estoque_amarelo_peca_4=A4 THEN
        {
        BEGIN
        DEC kanban_verde_peca_4, 1
        INC estoque_verde_peca_4, 1
        END
        }

```

```

    }
    IF a_tipo_de_peca = 5 and
estoque_vermelho_peca_5<VE5 THEN
    {
    BEGIN
    DEC kanban_vermelho_peca_5, 1
    INC estoque_vermelho_peca_5, 1
    END
    }
    ELSE
    IF a_tipo_de_peca = 5 and
estoque_vermelho_peca_5=VE5 and estoque_amarelo_peca_5<A5 THEN
    {
    BEGIN
    DEC kanban_amarelo_peca_5, 1
    INC estoque_amarelo_peca_5, 1
    END
    }
    ELSE
    IF a_tipo_de_peca = 5 and
estoque_vermelho_peca_5=VE5 and estoque_amarelo_peca_5=A5 THEN
    {
    BEGIN
    DEC kanban_verde_peca_5, 1
    INC estoque_verde_peca_5, 1
    END
    }
    }

    IF a_tipo_de_peca = 6 and
estoque_vermelho_peca_6<VE6 THEN
    {
    BEGIN
    DEC kanban_vermelho_peca_6, 1
    INC estoque_vermelho_peca_6, 1
    END
    }
    ELSE
    IF a_tipo_de_peca = 6 and
estoque_vermelho_peca_6=VE6 and estoque_amarelo_peca_6<A6 THEN
    {
    BEGIN
    DEC kanban_amarelo_peca_6, 1
    INC estoque_amarelo_peca_6, 1
    END
    }
    ELSE
    IF a_tipo_de_peca = 6 and
estoque_vermelho_peca_6=VE6 and estoque_amarelo_peca_6=A6 THEN
    {
    BEGIN
    DEC kanban_verde_peca_6, 1
    INC estoque_verde_peca_6, 1
    END
    }
    }
    IF a_tipo_de_peca = 7 and
estoque_vermelho_peca_7<VE7 THEN
    {

```

```

BEGIN
DEC kanban_vermelho_peca_7, 1
INC estoque_vermelho_peca_7, 1
END
}
ELSE
IF a_tipo_de_peca = 7 and
estoque_vermelho_peca_7=VE7 and estoque_amarelo_peca_7<A7 THEN
{
BEGIN
DEC kanban_amarelo_peca_7, 1
INC estoque_amarelo_peca_7, 1
END
}
ELSE
IF a_tipo_de_peca = 7 and
estoque_vermelho_peca_7=VE7 and estoque_amarelo_peca_7=A7 THEN
{
BEGIN
DEC kanban_verde_peca_7, 1
INC estoque_verde_peca_7, 1
END
}
}
1 kanban EXIT
FIRST 1

```

```

*****
**
*
* Arrivals
*
*****
**

```

Entity	Location	Qty each	First Time	Occurrences
Frequency	Logic			
--				
peca	Cliente	1	Tue, Mar 28 2006 @ 06:00 AM	1
a_tipo_de_peca = 1				
peca	Cliente	1	Tue, Mar 28 2006 @ 08:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Mar 28 2006 @ 10:40 AM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Tue, Mar 28 2006 @ 01:00 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Mar 29 2006 @ 07:00 AM	1
a_tipo_de_peca = 1				
peca	Cliente	1	Wed, Mar 29 2006 @ 09:10 AM	1
a_tipo_de_peca = 3				
peca	Cliente	1	Wed, Mar 29 2006 @ 11:30 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Mar 29 2006 @ 02:05 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Mar 30 2006 @ 07:40 AM	1
a_tipo_de_peca = 6				

peca	Cliente	1	Thu, Mar 30 2006 @ 10:10 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Mar 30 2006 @ 02:20 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Fri, Mar 31 2006 @ 07:40 AM	1
a_tipo_de_peca = 7				
peca	Cliente	1	Fri, Mar 31 2006 @ 10:10 AM	1
a_tipo_de_peca = 2				
peca	Cliente	1	Fri, Mar 31 2006 @ 12:30 PM	1
a_tipo_de_peca = 3				
peca	Cliente	1	Mon, Apr 03 2006 @ 06:00 AM	1
a_tipo_de_peca = 4				
peca	Cliente	1	Mon, Apr 03 2006 @ 08:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Mon, Apr 03 2006 @ 10:40 AM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Mon, Apr 03 2006 @ 01:00 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 04 2006 @ 06:25 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 04 2006 @ 08:40 AM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Tue, Apr 04 2006 @ 10:30 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 04 2006 @ 10:30 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 04 2006 @ 01:00 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 05 2006 @ 07:40 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 05 2006 @ 10:10 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 05 2006 @ 02:20 PM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Thu, Apr 06 2006 @ 07:50 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Apr 06 2006 @ 10:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Apr 06 2006 @ 12:30 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Fri, Apr 07 2006 @ 06:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Fri, Apr 07 2006 @ 02:20 PM	1
a_tipo_de_peca = 3				
peca	Cliente	1	Mon, Apr 10 2006 @ 07:40 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Mon, Apr 10 2006 @ 10:10 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 11 2006 @ 12:30 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 12 2006 @ 06:20 AM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Wed, Apr 12 2006 @ 08:30 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 12 2006 @ 10:50 AM	1
a_tipo_de_peca = 6				

peca	Cliente	1	Wed, Apr 12 2006 @ 02:20 PM	1
a_tipo_de_peca = 3				
peca	Cliente	1	Thu, Apr 13 2006 @ 08:10 AM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Thu, Apr 13 2006 @ 10:30 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Apr 13 2006 @ 12:50 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Fri, Apr 14 2006 @ 02:20 PM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Mon, Apr 17 2006 @ 07:50 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Mon, Apr 17 2006 @ 10:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Mon, Apr 17 2006 @ 12:40 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 18 2006 @ 06:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 18 2006 @ 08:30 AM	1
a_tipo_de_peca = 1				
peca	Cliente	1	Tue, Apr 18 2006 @ 02:20 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 19 2006 @ 07:40 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 19 2006 @ 10:10 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 19 2006 @ 12:30 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Apr 20 2006 @ 06:20 AM	1
a_tipo_de_peca = 4				
peca	Cliente	1	Thu, Apr 20 2006 @ 08:30 AM	1
a_tipo_de_peca = 5				
peca	Cliente	1	Thu, Apr 20 2006 @ 10:50 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Thu, Apr 20 2006 @ 01:20 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Mon, Apr 24 2006 @ 06:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Mon, Apr 24 2006 @ 08:20 AM	1
a_tipo_de_peca = 2				
peca	Cliente	1	Mon, Apr 24 2006 @ 10:40 AM	1
a_tipo_de_peca = 3				
peca	Cliente	2	Mon, Apr 24 2006 @ 02:20 PM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 25 2006 @ 07:50 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 25 2006 @ 10:20 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Tue, Apr 25 2006 @ 02:20 PM	1
a_tipo_de_peca = 4				
peca	Cliente	1	Wed, Apr 26 2006 @ 07:40 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 26 2006 @ 10:10 AM	1
a_tipo_de_peca = 6				
peca	Cliente	1	Wed, Apr 26 2006 @ 12:30 PM	1
a_tipo_de_peca = 6				

peca Cliente 4 Thu, Apr 27 2006 @ 07:00 AM 1
 a_tipo_de_peca = 6

```
*****
**
*
* Shift Assignments
*
*****
**
```

Locations	Resources	Shift Files	Priorities
-----------	-----------	-------------	------------

Disable Logic

```
-----
--
Caixa_aberta D:\Bruno\UNIFEI\MESTRADO\Quart 99,99,99,99
No
Cliente
Esteira
Estoque_final
INspecao
Mesa_1
Mesa_2
Mesa_3
Mesa_4
Mesa_5
Mesa_6
Mesa_7
Mesa_8
Pallet_submontagens
Prateleira
Pre_plug_1
Pre_plug_2
Prensa_1
Prensa_2
Programacao_producao
Termo_encolhivel
```

producao:1 D:\Bruno\UNIFEI\MESTRADO\Quart 99,99,99,99

No

```
Mesa_1 D:\Bruno\UNIFEI\MESTRADO\Quart 99,99,99,99
Yes
Mesa_3
Mesa_2
Mesa_4
```

```
*****
**
*
* Attributes
*
*****
**
```

ID	Type	Classification
a_tipo_de_peca	Integer	Entity

```

*****
**
*                               Variables (global)
*
*****
**

```

ID	Type	Initial value	Stats
kanban_verde_peca_1	Integer	0	None
kanban_amarelo_peca_1	Integer	0	None
kanban_vermelho_peca_1	Integer	0	None
kanban_verde_peca_2	Integer	0	None
kanban_amarelo_peca_2	Integer	0	None
kanban_vermelho_peca_2	Integer	0	None
Kanban_verde_peca_3	Integer	0	None
kanban_amarelo_peca_3	Integer	0	None
kanban_vermelho_peca_3	Integer	0	None
kanban_verde_peca_4	Integer	0	None
Kanban_amarelo_peca_4	Integer	0	None
kanban_vermelho_peca_4	Integer	0	None
kanban_verde_peca_5	Integer	0	None
kanban_amarelo_peca_5	Integer	0	None
kanban_vermelho_peca_5	Integer	0	None
kanban_verde_peca_6	Integer	0	None
kanban_amarelo_peca_6	Integer	0	None
kanban_vermelho_peca_6	Integer	0	None
kanban_verde_peca_7	Integer	0	None
kanban_amarelo_peca_7	Integer	0	None
kanban_vermelho_peca_7	Integer	0	None
estoque_verde_peca_1	Integer	0	Time Series
estoque_amarelo_peca_1	Integer	1	Time Series
estoque_vermelho_peca_1	Integer	0	Time Series
estoque_verde_peca_2	Integer	0	Time Series
estoque_amarelo_peca_2	Integer	1	Time Series
estoque_vermelho_peca_2	Integer	0	Time Series
estoque_verde_peca_3	Integer	1	Time Series
estoque_amarelo_peca_3	Integer	1	Time Series
estoque_vermelho_peca_3	Integer	0	Time Series
estoque_verde_peca_4	Integer	1	Time Series
estoque_amarelo_peca_4	Integer	1	Time Series
estoque_vermelho_peca_4	Integer	0	Time Series
estoque_verde_peca_5	Integer	2	Time Series
estoque_amarelo_peca_5	Integer	1	Time Series
estoque_vermelho_peca_5	Integer	1	Time Series
estoque_verde_peca_6	Integer	11	Time Series
estoque_amarelo_peca_6	Integer	6	Time Series
estoque_vermelho_peca_6	Integer	5	Time Series
estoque_verde_peca_7	Integer	0	Time Series
estoque_amarelo_peca_7	Integer	1	Time Series
estoque_vermelho_peca_7	Integer	0	Time Series
Total_de_peças_produzidas	Integer	0	Time Series
Estoque_submontagens	Integer	0	Time Series
dimensionam_verde_peca_1	Integer	0	Time Series
dimensionam_amarelo_peca_1	Integer	1	Time Series
dimensionam_vermelho_peca_1	Integer	0	Time Series
dimensionam_verde_peca_2	Integer	0	Time Series

```

dimensionam_amarelo_pecas_2 Integer 1 Time Series
dimensionam_vermelho_pecas_2 Integer 0 Time Series
dimensionam_verde_pecas_3 Integer 1 Time Series
dimensionam_amarelo_pecas_3 Integer 1 Time Series
dimensionam_vermelho_pecas_3 Integer 0 Time Series
dimensionam_verde_pecas_4 Integer 1 Time Series
dimensionam_amarelo_pecas_4 Integer 1 Time Series
dimensionam_vermelho_pecas_4 Integer 0 Time Series
dimensionam_verde_pecas_5 Integer 2 Time Series
dimensionam_amarelo_pecas_5 Integer 1 Time Series
dimensionam_vermelho_pecas_5 Integer 1 Time Series
dimensionam_verde_pecas_6 Integer 11 Time Series
dimensionam_amarelo_pecas_6 Integer 6 Time Series
dimensionam_vermelho_pecas_6 Integer 5 Time Series
dimensionam_verde_pecas_7 Integer 0 Time Series
dimensionam_amarelo_pecas_7 Integer 1 Time Series
dimensionam_vermelho_pecas_7 Integer 0 Time Series
Var1 Integer 0 Time Series
Var2 Integer 0 Time Series
Var3 Integer 0 Time Series
Var4 Integer 0 Time Series
Var5 Integer 0 Time Series
Var6 Integer 0 Time Series
Var7 Integer 0 Time Series
opcao_1 Integer 0 Time Series
opcao_2 Integer 0 Time Series
opcao_3 Integer 0 Time Series
opcao_4 Integer 0 Time Series
opcao_5 Integer 0 Time Series
opcao_6 Integer 0 Time Series
opcao_7 Integer 0 Time Series
qdade1 Integer 0 Time Series
qdade2 Integer 0 Time Series
qdade3 Integer 0 Time Series
qdade4 Integer 0 Time Series
qdade5 Integer 0 Time Series
qdade6 Integer 0 Time Series
qdade7 Integer 0 Time Series
buffer Integer 0 Time Series
aVar8 Integer 0 Time Series
cont1 Integer 0 Time Series
cont2 Integer 0 Time Series
cont3 Integer 0 Time Series
cont4 Integer 0 Time Series
cont5 Integer 0 Time Series
cont6 Integer 0 Time Series
cont7 Integer 0 Time Series
chegada Integer 0 Time Series
wip Integer 0 Time Series
*****
**
*                               Macros
*
*****
**
ID                               Text
-----

```

```

Macl
      IF estoque_verde_peca_1>0 and estoque_verde_peca_1 <
dimensionam_verde_peca_1 THEN
      {
      DEC estoque_verde_peca_1, 1
      INC kanban_verde_peca_1, 1
      }
      ELSE
      IF  estoque_verde_peca_1 = 0 and estoque_amarelo_peca_1>0
THEN
      {
      DEC estoque_amarelo_peca_1, 1
      INC kanban_amarelo_peca_1, 1
      }
      ELSE
      IF  estoque_amarelo_peca_1 = 0 and estoque_vermelho_peca_1>0
THEN
      {
      DEC estoque_vermelho_peca_1, 1
      INC kanban_vermelho_peca_1, 1
      }

```

```
*****
```

```
**
```

```
*
```

Subroutines

```
*
```

```
*****
```

```
**
```

ID	Type	Parameter	Type	Logic
----	------	-----------	------	-------

Sub1	Integer			
------	---------	--	--	--

```
*****
```

```
**
```

```
*
```

Table Functions

```
*
```

```
*****
```

```
**
```

ID	Independent Value	Dependent Value
----	-------------------	-----------------

Fun1		
------	--	--

```
*****
```

```
**
```

```
*
```

User Distributions

```
*
```

```
*****
```

```
**
```

ID	Type	Cumulative	Percentage	Value
----	------	------------	------------	-------

distribuicao_de_chegada	Discrete	No	1.4	1
			1.4	2
			6.9	3
			2.8	4
			11.1	5
			73.6	6
			2.8	7

```
*****  
**  
*                               External Files  
*  
*****  
**  
ID           Type           File Name  
Prompt  
-----  
-----  
(null)      Shift          D:\Bruno\UNIFEI\MESTRADO\Quarto trimestre\  
Modelos\TURN01.sft  
(
```

ANEXO 2

Artigos escritos a partir do tema desta dissertação

Neste anexo estão relacionados os resumos dos artigos publicados decorrentes do tema desta dissertação. Os artigos completos podem ser obtidos com o autor, através do email: brunotorga@yahoo.com.br.

III SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA (SEGET), 2006.

Simulação e Otimização em sistemas puxados de manufatura: um estudo de caso em uma empresa do ramo automobilístico.

Resumo: Este artigo tem como objetivo explorar a simulação computacional nos sistemas de manufatura diferenciando os tipos de sistema de produção existentes e descrevendo sua aplicação em uma linha de produção puxada através de um estudo de caso. Objetiva-se também através da aplicação dos conceitos de otimização minimizar o número de *kanbans* da linha produtiva objeto de estudo. Não existem muitas aplicações dessa ferramenta neste tipo de linha de manufatura e uma das vantagens desta aplicação é a possibilidade de se realizar experimentações e alcançar modelos que estejam em maior concordância com os conceitos do Just in Time, cujo sistema puxado é um dos principais fundamentos.

Palavras-chave: Simulação computacional. Simulação de sistemas puxados. Otimização de *kanbans*.

XIII SIMPÓSIO DE ENGENHARIA DE PRODUÇÃO (SIMPEP), 2006.

Modelagem, simulação e otimização em sistemas puxados de manufatura.

Resumo: Este artigo tem como objetivo explorar a simulação computacional na manufatura diferenciando os tipos de sistema de produção existentes e descrevendo sua aplicação em uma linha de produção puxada através de um estudo de caso realizado em uma empresa automobilística localizada no Sul de Minas Gerais. Objetiva-se também através da associação entre os conceitos de simulação e otimização minimizar o número de *kanbans* da linha produtiva objeto de estudo. Não existem muitas aplicações dessa ferramenta neste tipo de linha de manufatura e uma das vantagens desta aplicação é a possibilidade de se realizar experimentações e alcançar modelos que estejam em maior concordância com os conceitos do Just in Time, cujo sistema puxado é um dos principais fundamentos.

Palavras-chave: Simulação computacional, Simulação de sistemas puxados, Otimização de *kanban*.