

LINDOLPHO OLIVEIRA DE ARAÚJO JUNIOR

**MÉTODO DE PROGRAMAÇÃO DE SISTEMAS
DE MANUFATURA DO TIPO JOB SHOP
DINÂMICO NÃO DETERMINÍSTICO**

Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia.

São Paulo

2006

LINDOLPHO OLIVEIRA DE ARAÚJO JUNIOR

**MÉTODO DE PROGRAMAÇÃO DE SISTEMAS
DE MANUFATURA DO TIPO JOB SHOP
DINÂMICO NÃO DETERMINÍSTICO**

Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia.

Programa: Engenharia Mecânica

Área de Concentração: Engenharia de
Controle e Automação Mecânica –
Engenharia Mecatrônica

Orientador: Prof. Dr. Diolino José dos
Santos Filho

São Paulo

2006

“Não é que eu procure falar difícil, é que as coisas têm nome”

(João Guimarães Rosa)

“Adesso è il momento, hai il tempo

Per capire, di avere occhi sinceri”

(John Lydon)

Obstacles are things a person sees when he takes his eyes off his goal.

(E. Joseph Cossman)

AGRADECIMENTOS

Aos meus pais minha eterna gratidão por terem me trazido à vida e me conduzido pelo caminho da educação e da sabedoria.

Pelo amor e dedicação de minha esposa Renata e minhas filhas Uliana e Letícia, minha sincera gratidão.

Ao amigo e orientador Prof. Dr. Diolino José dos Santos Filho, minha eterna gratidão por todo o incentivo e auxílio.

Ao amigo Prof. Dr. Newton Ribeiro dos Santos (*in memoriam*) que me ensinou a dar os primeiros passos como pesquisador.

Aos amigos Osvaldo, André, Cristina, Gladys e Francisco pelas valiosas contribuições e pela amizade.

A Universidade Tecnológica Federal do Paraná por me conceder licença capacitação para a realização em tempo integral do programa de doutorado junto à Escola Politécnica da Universidade de São Paulo.

A todos os professores do departamento de Engenharia Mecatrônica que fazem a grandeza desta Escola Politécnica.

Aos membros da banca avaliadora pelas contribuições valiosas.

RESUMO

Esta tese explora características de sistemas de manufatura celular virtual, de técnicas de escalonamento e despacho em tempo de execução e de algoritmos genéticos para assegurar um alto grau de flexibilidade de rotas de processo e eficiência na programação de tarefas de sistemas de manufatura do tipo (NDDJSS – *non-deterministic dynamic job shop system*). Desenvolve-se assim, um método de programação baseado em duas novas técnicas, de *clustering* e de escalonamento e despacho, respectivamente, aplicado a sistemas de manufatura do tipo NDDJSS. Apresentam-se inicialmente os conceitos fundamentais considerados na organização de sistemas de manufatura em células de trabalho e suas implicações em termos de esforço de coordenação e controle das células e das tarefas; a abordagem por organização funcional das células de manufatura virtuais; as características, definições e notação usadas para problemas de escalonamento; a descrição de técnicas de escalonamento e despacho de sistemas de manufatura; aspectos relevantes de computação evolutiva, mais especificamente, de algoritmos genéticos. Em seguida, introduz-se a nova técnica de *clustering* e a nova técnica de escalonamento e despacho, juntamente com os requisitos e as formulações associadas. A nova técnica de *clustering* forma células de manufatura virtuais balanceadas quanto ao volume de trabalho, com alto grau de flexibilidade de rotas de processo. A nova técnica de escalonamento e despacho é autônoma, estável, adaptativa, flexível, contingente, monotônica e relativamente robusta, permitindo a geração de escalas em tempo de execução, ao mesmo tempo em que mantém a medida de desempenho, o *makespan*, dentro de uma faixa de desempenho pré-estabelecida. Exemplificam-se as aplicações das duas novas técnicas, através de exemplos extraídos da literatura técnica e simulados através de um programa de computador chamado (DEVICE – *Design of virtual cells*), desenvolvido especialmente para esta pesquisa. Após simulação obtêm-se os resultados de forma estruturada para análise. Os respectivos resultados da aplicação do método de programação proposto comprovam o potencial dessa abordagem.

ABSTRACT

This thesis explores characteristics of virtual cellular manufacturing systems, of scheduling and dispatching techniques for real time requirements and genetic algorithms for assures a high level of routeing flexibility and efficiency in tasks programming for (NDDJSS – non-deterministic dynamic job shop system). It introduces a programming method based in two new clustering and scheduling and dispatching techniques, respectively, applied in NDDJSS manufacturing systems. Initially, the work presents basic concepts considered in the organization of cellular manufacturing systems and its implications for coordination and control efforts of cells and tasks, respectively; an approach of virtual manufacturing cells through the functional organization; the characteristics, definitions and notation used for scheduling problems; the description of more manufacturing systems scheduling and dispatching techniques; relevant aspects of evolutionary computation in specialty, the genetic algorithms. Subsequently, the work also introduces the new clustering and scheduling and dispatching techniques, the requirements and the formulas associated. The new clustering technique builds balanced workload with high flexibility of process routeing. Also, the new scheduling and dispatching technique is autonomous, stable, adaptative, flexible, contingent, monotonic and relatively robust, providing scales generating in running mode (real time) and mantaining the performance measures “makespan” within the stabilished range of performance. The applications of the two new techniques are exemplified based on two examples picked from technique literature and so simulated through a new software called (DEVICE – design of virtual cells). The software DEVICE was designed specialty for this research. After run some simulations, the structured form results are obtained for analisis. The respectively results obtained for the proposed method application proves the powerful of this approach.

LISTA DE FIGURAS

Figura 1.1 – Pirâmide de Automação.	25
Figura 1.2 - Hierarquia Funcional de Tomada de Decisão e Controle.	27
Figura 1.3 - Organização típica para equipamentos previsto na norma ANSI/ISA S88.01.....	30
Figura 2.1 - Exemplo de decomposição imperfeita para matriz de incidência máquinas-peças ADIL; RAJAMANI; STRONG, 1996).....	55
Figura 2.2 - Efeito dos elementos nulos e excepcionais sob a manipulação de materiais.	59
Figura 2.3 – Exemplo de <i>cell shop</i> com duas células virtuais ativas e nenhum recurso compartilhado entre as células.....	64
Figura 2.4 - a) Matriz original a_1 b) Matriz a_2 (decomposição imperfeita da matriz a_1).	69
Figura 2.5 - Melhoria após decomposição por planos alternativos (rotas alternativas).	69
Figura 2.6 - Melhoria após duplicação da máquina do tipo 2.	70
Figura 2.7 – Arquitetura física e lógica para a hierarquia de equipamentos desta tese.	73
Figura 2.8 - Desenho Esquemático da Planta Industrial Hipotética composta por uma área e oito organizações funcionais.	77
Figura 2.9 - Um local (<i>site</i>) formado por quatro áreas.	78
Figura 2.10 - Modelo hierárquico de equipamentos para o HP organizado sob a forma de células de manufatura virtual.	80
Figura 2.11 - Fluxograma simplificado do ciclo de um AG.	91
Figura 2.12 – Um exemplo de representação da população de indivíduos para o problema proposto na Tabela 2.9 da pág.98.....	94
Figura 3.1 - Classificação dos Métodos de Escalonamento (CASAVANT; KUHL, 1988).....	102
Figura 3.2 – Um exemplo de um possível grafo de precedência.....	113

Figura 3.3 - Diagrama de Gantt.	114
Figura 3.4 - Diagrama de tarefa-tempo.	114
Figura 3.5 - Tempo crítico momentâneo para um <i>job</i> em andamento.	135
Figura 3.6 – Algoritmo de BB proposto.	136
Figura 4.1 – Tela de abertura do <i>software</i> DEVICE.	140
Figura 4.2 – Análise gráfica da evolução do AG através do <i>software</i> DEVICE.	140
Figura 4.3 – Informação textual da evolução do AG.	141
Figura 4.4 – Gráfico de evolução do AG para o problema original de Chan (2003).	147
Figura 4.5 – Frequência de utilização das máquinas para o problema original.	148
Figura 4.6 - Gráfico de evolução do AG para o problema de Chan (2003) modificado.	149
Figura 4.7 - Frequência de utilização das máquinas para o problema modificado. .	150
Figura 4.8 – Desempenho do sistema para 50 cenários e variação de 0% no tempo de processamento.	154
Figura 4.9 – Gráfico de Gantt dos processos do primeiro cenário com variação de 0% no tempo de processamento.	155
Figura 4.10 – Gráfico de Gantt das máquinas do primeiro cenário com variação de 0% no tempo de processamento.	155
Figura 4.11 – Desempenho do sistema para 50 cenários com variação de 1% no tempo de processamento.	156
Figura 4.12 – Desempenho do sistema para 50 cenários com variação de 2% no tempo de processamento.	157
Figura 4.13 – Desempenho do sistema para 50 cenários com variação de 3% no tempo de processamento.	157
Figura 4.14 – Desempenho do sistema para 50 cenários com variação de 4% no tempo de processamento.	157
Figura 4.15 – Desempenho do sistema para 50 cenários com variação de 5% no tempo de processamento.	158
Figura 4.16 – Desempenho do sistema para 50 cenários com variação de 10% no tempo de processamento.	158

Figura 4.17 – Desempenho do sistema para 50 cenários com variação de 0% no tempo de processamento.	159
Figura 4.18 – Gráfico de Gantt dos processos do primeiro cenário com variação de 0% no tempo de processamento.....	160
Figura 4.19 – Gráfico de Gantt das máquinas do primeiro cenário com variação de 0% no tempo de processamento.....	160
Figura 4.20 – Desempenho do sistema para 50 cenários com variação de 5% no tempo de processamento.	161
Figura 4.21 – Desempenho do sistema para 50 cenários com variação de 6% no tempo de processamento.	161
Figura 4.22 – Desempenho do sistema para 50 cenários com variação de 7% no tempo de processamento.	161
Figura 4.23 – Desempenho do sistema para 50 cenários com variação de 10% no tempo de processamento.	162
Figura 4.24 - Gráfico de Gantt dos processos do vigésimo terceiro cenário com variação de 10% no tempo de processamento.	163
Figura 4.25 - Gráfico de Gantt das máquinas do vigésimo terceiro cenário com variação de 10% no tempo de processamento.	163

LISTA DOS SÍMBOLOS

$1-w$	Peso de um elemento <i>void</i>
$1/q_i$	Fator de quantidade
α	Fator de escala
$\alpha \beta \gamma$	Notação de Brucker para definição de um problema de escalonamento
$\beta(t)$	<i>critical ratio</i>
δ	Índice de prioridade
$\sum C_{ij}$	Somatório de todos os custos das tarefas escolhidas para um cromossomo
$\sum p_{ij}$	Somatório de todos os tempos das tarefas escolhidas para um cromossomo
$\sum_{q=j}^{mi} p_{iq}$	Tempo de processamento remanescente do <i>job i</i> após a $(j-1)$ -ésima operação ter sido completada
c	Índice de célula
c_{jk}	Tempo de comunicação entre máquinas
CMV(r)	r-ésima célula de manufatura virtual
d	Quantidade de '1' nos blocos diagonais
d_i	Data devida do <i>job i</i>
d_j	Data devida
D_j	Data de entrega da tarefa j
D_j'	Data de entrega modificada da tarefa j
d_{kj}	Data de entrega da operação k da tarefa j
e	Quantidade de elementos excepcionais na solução
E_j	Instante de chegada da tarefa j ao sistema
f	Função objetivo (aptidão ou <i>fitness</i>)
F	Número de células de um horizonte de planejamento
fc	Frequência de etapas para cada célula do HP
$F_{\text{escolhido}}$	Quantidade de células escolhida para o HP

HP(i)	Horizonte de planejamento de ordem i
Jm	Máquinas usadas em sistemas do tipo job shop
K	Tamanho da população
M	Quantidade de máquinas necessárias para realização dos <i>jobs</i> de um cromossomo
M'	Número total de operações de cada tarefa
M''	Quantidade de máquinas para um HP
M(k)	k -ésima máquina
M_c	Quantidade de máquinas alocadas na célula c
m_i	Número de operações que o <i>job</i> i irá necessitar a partir do tempo t
M_j	Subconjunto de M
$M_{máx}$	Quantidade máxima de máquinas admitida para a célula virtual
N^c	Número de pontos de crossover
O	Quantidade de '1' na matriz de incidência
OF(n)	Organização funcional de ordem n
Om	Máquinas usadas em sistemas do tipo open shop
P_c	Quantidade de peças alocadas para a célula c
P_j	Tempo requerido pelas operações restantes da tarefa j , a partir do instante t
pm	Taxa de mutação
<i>prec</i>	Precedência
<i>prpm</i>	Preempção
<i>recrc</i>	Recirculação
r_j	Data de disponibilidade
s_j	Folga da tarefa j
S_{jk}	Tempo de preparação
t	Tempo corrente (instantâneo)
T	Conjunto de n tarefas
t_j	j -ésima tarefa
v	Quantidade de <i>voids</i> na solução
w	Peso de um elemento excepcional ($0 \leq w \leq 1$)

LISTA DAS ABREVIATURAS EM PORTUGUÊS

AG	Algoritmo genético
CE	Computação evolutiva
CMV	Célula de manufatura virtual
HP	Horizonte de planejamento
IHM	Interface homem-máquina
MC	Manufatura celular
MCV	Manufatura celular virtual
NP	Não polinomial
OF	Organização funcional
RDM	Regra de despacho mista
RHC	Regra heurística combinada
RN	Redes neurais
SED	Sistemas a eventos discretos
SM	Sistemas de manufatura
SP	Sistemas produtivos
TG	Tecnologia de grupo

LISTA DAS ABREVIATURAS EM INGLÊS

ANSI	American national standards institute
ATC	Apparent tardiness cost
ATO	Assemble to order
BB	Branch and bound
CIM	Computer integrated manufacturing
COVERT	Weighted cost over time
CR	Critical ratio
CR+SPT	Critical ration plus shortest processing time
DEVICE	Design of virtual cells
DJSP	Dynamic deterministic job shop problem
EDD	Earliest due date
ERD	Earliest release date
ETO	Engineering to order
FCFS	First-come-first-serve
FIFO	First in first out
FMS	Flexible manufacturing system
ISA	Instrumentation systems and automation society
JIT	Just-in-time
JSP	Job shop problem
MDD	Modified due date
MES	Manufacturing execution system
MESA	Manufacturing execution systems association
MOD	Modified operation due date
MSLACK	Minimum slack
MST	Minimum slack time
MTO	Make to order

MTS	Make to stock
NDDJSS	Non-deterministic dynamic job shop system
NDJSP	Non-deterministic dynamic job shop problem
ODD	Operation due date
PERA	Purdue enterprise reference architecture
S/RPT	Slack per remaining processing time
S/RPT+SPT	Slack per remainig processing time plus shortest processing time
SPT	Shortest processing time
WIP	Work-in-process
WSPT	Weighted shortest processing time

LISTA DE TABELAS

Tabela 2.1 - Taxonomia de Tipos de Flexibilidade.	42
Tabela 2.2 – Relação entre metas e características de sistemas do tipo NDDJSS.	44
Tabela 2.3 – Uma possível descrição das organizações funcionais do exemplo hipotético.	81
Tabela 2.4 – Uma possível descrição de demanda para um HP.	82
Tabela 2.5 – Um exemplo de requisitos para a formação das CMV de um HP.	83
Tabela 2.6 - Frequência de Rotas por Quantidade de Etapas para o HP da Tabela 2.5.	84
Tabela 2.7 – Um exemplo de distribuição das CMV para ($F = 3$).	86
Tabela 2.8 - Matriz de Incidência de Máquinas-Peças com as três células formadas.	86
Tabela 2.9 – Exemplo de Descrição de Rotas Alternativas de Processo da Célula ($n=3$).	98
Tabela 2.10 - Totalização do número de rotas logicamente possíveis e do número de programas necessários para o exemplo da Tabela 2.9.	99
Tabela 3.1 - Variações de Máquinas do Tipo <i>Shop</i>	109
Tabela 3.2 - Outras características do campo de tarefas.	110
Tabela 3.3 - Funções de Custo mais Comuns.	112
Tabela 3.4 - Notação usada pelo algoritmo de BB.	135
Tabela 4.1 – Descrição dos Processos Originais de Chan (2003).	142
Tabela 4.2 – Demanda por produtos.	143
Tabela 4.3 – Descrição dos Processos para o problema modificado.	145
Tabela 4.4 – Representação dos tempos de processamento e das rotas alternativas de processo proposto por Kumar; Tiwari; Shankar (2003).	152

SUMÁRIO

AGRADECIMENTOS	iv
RESUMO	v
ABSTRACT	vi
LISTA DE FIGURAS.....	vii
LISTA DOS SÍMBOLOS	x
LISTA DAS ABREVIATURAS EM PORTUGUÊS.....	xii
LISTA DAS ABREVIATURAS EM INGLÊS.....	xiii
LISTA DE TABELAS.....	xv
SUMÁRIO	xvi
1 INTRODUÇÃO.....	19
1.1 Problema Abordado.....	34
1.2 Motivação	34
1.3 Objetivo do Trabalho	36
1.4 Justificativa.....	38
1.5 Organização do Trabalho.....	39
2 TÉCNICA DE <i>CLUSTERING</i> PARA FORMAÇÃO DE CÉLULAS DE MANUFATURA VIRTUAIS	40
2.1 Flexibilidade de Rotas de Processo.....	40
2.2 Flexibilidade em Sistemas de Manufatura.....	41
2.3 Critérios Estratégicos de Produção	45
2.4 Sistemas de Manufatura do Tipo <i>Job Shop</i>	46
2.4.1 O problema em SM do tipo <i>Job Shop</i> Clássico Estático	46
2.4.2 O problema do tipo NDJSP com datas de entrega	48
2.5 Manufatura Celular	49
2.5.1 Histórico da manufatura celular clássica	50
2.5.1.1 Características de Formação de Células de Manufatura.....	54
2.5.1.2 Esforço de coordenação e controle.....	56
2.5.1.3 Manipulação de material intra e intercelular.....	58

2.6 Manufatura Celular Virtual.....	62
2.7 Definição da Técnica de <i>Clustering</i> para Formação de Células de Manufatura Virtuais.....	65
2.7.1 Requisitos.....	67
2.7.2 Hipóteses para Formação de Células de Manufatura Virtuais.....	72
2.7.3 Nova Técnica de <i>Clustering</i>	73
2.7.3.1 Exemplo ilustrativo de aplicação da Técnica de <i>Clustering</i> Proposta.....	77
2.8 Comentários Finais do Capítulo.....	100
3 TÉCNICA DINÂMICA DE ESCALONAMENTO E DESPACHO DE SISTEMAS DO TIPO NDJSP.....	101
3.1 Escalonamento de Sistemas.....	101
3.1.1 Escalonamento Dinâmico.....	103
3.2 Técnicas de Escalonamento aplicadas a Sistemas de Manufatura.....	106
3.2.1 Escalonamento - Definição Formal e Notação de Brucker.....	106
3.2.1.1 Notação.....	108
3.2.1.1.1 Campo α - Recursos.....	108
3.2.1.1.1.1 Máquinas <i>Shop</i> (Especializadas).....	108
3.2.1.1.2 Campo β - Tarefas.....	109
3.2.1.1.3 Campo γ - Critério de Otimização.....	111
3.2.1.2 Representações gráficas.....	113
3.2.2 Escalonamento aplicado a <i>job-shop</i>	115
3.2.3 Técnicas de Escalonamento.....	116
3.2.3.1 Técnicas Reativas.....	119
3.2.3.2 Técnicas Pró-ativas.....	119
3.2.3.3 Técnicas Progressivas.....	119
3.2.3.4 Técnicas Mistas.....	120
3.2.3.5 Características mínimas para uma Técnica Mista aplicada ao Problema tratado nesta Pesquisa.....	120
3.3 Proposta de Técnica de Escalonamento para Sistemas do tipo NDDJSS.....	121
3.3.1 Métodos Heurísticos para Geração de Regras de Despacho.....	123
3.3.1.1 Heurística de passo único.....	123
3.3.1.2 Heurísticas de busca local.....	126

3.3.2 Regras Heurísticas Simples <i>versus</i> Regras Heurísticas Combinadas.....	127
3.3.3 Seqüenciamento com operações alternativas.....	128
3.3.4 Escalonamento de múltiplas tarefas com operações alternativas.....	129
3.3.5 Fatores Econômicos em Problemas de Escalonamento.....	130
3.3.6 Regra Heurística Combinada (RHC).....	132
3.4 Método de Análise de Resultados.....	137
3.5 Comentários Finais do Capítulo.....	138
4 EXEMPLOS DE APLICAÇÃO DAS NOVAS TÉCNICAS E ANÁLISE DOS RESULTADOS	139
4.1 Um Exemplo de Aplicação do Algoritmo Genético usado na nova Técnica de <i>Clustering</i>	141
4.1.1 Resultados da aplicação do AG ao problema proposto na Tabela 4.1.	146
4.1.2 Resultados da aplicação do AG ao problema proposto na Tabela 4.3.	148
4.2 Exemplos de Aplicação da Técnica de Escalonamento e Despacho em Tempo Real	151
4.2.1 Resultado da aplicação da nova técnica de escalonamento e despacho ao problema proposto na Tabela 4.4.....	153
4.2.2 Resultado da aplicação da nova técnica de escalonamento e despacho ao problema proposto na Tabela 4.1.....	158
4.2.3 Análise das penalidades aplicadas às escalas pelo algoritmo de escalonamento.....	162
5 CONCLUSÕES FINAIS.....	166
5.1 Trabalhos Futuros	167
REFERÊNCIA BIBLIOGRÁFICA.....	170

CAPÍTULO 1

INTRODUÇÃO

Este capítulo apresenta o contexto no qual se insere o trabalho, as justificativas que motivam o seu desenvolvimento, assim como o objetivo estabelecido. Também é exposta a forma como estão organizados os capítulos deste texto.

A classe de sistema de manufatura a ser abordada nesse trabalho é o *job shop*. De acordo com Oliveira (2002), o *job shop* tradicional é caracterizado por permitir diferentes fluxos das ordens de produção entre as máquinas, num ambiente constituído por múltiplos processos e múltiplos recursos de processamento. Pode-se dizer que se trata de uma forma de produção onde as diversas peças a serem produzidas atravessam o sistema de manufatura podendo existir ultrapassagem, ou seja, não possuem uma mesma seqüência, e ainda, existe a possibilidade de recirculação¹.

Ainda, segundo Oliveira (2002), essa organização apresenta uma característica inconveniente relativa às altas taxas de movimentação de materiais através do chão de fábrica. Neste contexto, ao caminhar no sentido de obter alternativas de rotas de processo para a execução dos processos, aumenta-se a complexidade do sistema a ser planejado e controlado. Sun et al. (2002), dizem que a produtividade nesse tipo de organização está diretamente relacionada ao uso eficaz dos recursos e à capacidade de usar rotas alternativas de processo, que por sua vez depende da capacidade de controle dos recursos, controle dos processos e de execução dos planos.

De acordo com Seixas Filho (2004), com as mudanças de perfil dos mercados homogêneos e estáveis² para mercados com características de demanda que requerem maior agilidade por parte das indústrias para atenderem aos pedidos dos clientes, surge à necessidade de se focar características de variedade e personalização dos

¹ Quando operações de um mesmo *job* (tarefa) devam ser processadas em uma mesma máquina.

² Mercados caracterizados por demandas constantes e previsíveis de produtos não personalizados.

produtos e de capacidade de atendimento dos pedidos. Então, as características essenciais em sistemas produtivos do tipo *job shop* são: utilização de rotas alternativas de processo, aumento da frequência de uso dos recursos aproveitando mais suas capacidades de produção, ciclos curtos de fabricação, baixos custos, garantia da qualidade e reposta rápida aos consumidores.

De acordo com Porter et al. (1999), um mecanismo alternativo para classificação de sistemas *job shop* é identificar e diferenciar esses sistemas entre os direcionados a estoque ou a ordens/encomendas. O primeiro grupo engloba os sistemas onde a venda do produto geralmente é feita após a sua produção, enquanto que para o segundo grupo ela é feita antes de sua produção. Essa classificação representa a forma de interação dos sistemas *job shop* com os clientes, ou seja, o nível de interferência que o cliente tem na produção do produto final.

Diante disso as indústrias podem escolher entre quatro abordagens apresentadas por Goulart (2000) conforme o ambiente da demanda:

- Produção para estoque (MTS – *Make to Stock*): caracteriza os sistemas que produzem produtos padronizados, baseados principalmente em previsões de demanda. Nesse caso, nenhum produto é personalizado, porque o pedido é feito com base no estoque de produtos acabados. Isso significa que a interação entre o cliente e o projeto do produto é muito pequena ou inexistente. Os sistemas MTS têm como principal vantagem maior rapidez na entrega dos produtos, mas os custos com estoques tendem a ser grandes e os clientes não têm como expressar suas necessidades a respeito dos produtos. Nesses sistemas os ciclos de vida dos produtos tendem a ser relativamente longos e previsíveis;
- Montagem sob encomenda (ATO – *Assemble to Order*): caracteriza os sistemas onde os subconjuntos, grandes componentes e materiais diversos são armazenados até o recebimento dos pedidos dos clientes contendo as especificações dos produtos finais. A interação entre o cliente e o projeto do produto é limitada. Nos sistemas ATO, as entregas dos produtos tendem a ser de médio prazo e as incertezas da demanda (quanto ao *mix* e volume dos

produtos) são gerenciadas com base no dimensionamento dos estoques de subconjuntos e capacidade das áreas de montagem;

- Produção sob encomenda (MTO – *Make to Order*): o projeto básico pode ser desenvolvido a partir dos contatos iniciais com o cliente, mas a etapa de produção só se inicia após o recebimento formal do pedido. As interações com os clientes são quase sempre do tipo extensiva, e o produto está sujeito a modificações mesmo durante a fase de produção. Num sistema MTO, os produtos geralmente não são um de cada tipo, porque usualmente os produtos são projetados a partir de especificações básicas. Os tempos de entrega tendem a ser entre médio e longo prazos e as listas de materiais são quase sempre únicas para cada produto;
- Engenharia sob Encomenda (ETO – *Engineering to Order*): é praticamente uma extensão do MTO, com o projeto do produto sendo feito quase que totalmente baseado nas especificações do cliente. Os produtos são altamente personalizados e o nível de interação com o cliente é muito grande.

Com estas abordagens fica clara a importância da adequação do processo de produção ao posicionamento da indústria em determinados mercados, na busca de vantagens competitivas frente a crescente concorrência. Nesta tese serão abordados apenas os sistemas orientados a ordens/encomendas, ou seja, ATO, MTO e ETO. Esta escolha deve-se a este tipo de produção ser uma tendência atual e por tratar-se de um tipo de produção, orientada ao cliente, onde a interferência dos pedidos é de extrema importância para a definição do modelo de programação e de controle do sistema produtivo. Esta é uma característica dinâmica, que requer, também, modelos dinâmicos para a programação e o controle dos sistemas produtivos.

A capacidade de produzir, podendo-se optar entre algumas alternativas de rotas de processo³, deve ser usada como uma arma competitiva, não podendo mais ser negligenciado pelas indústrias, sendo necessário, portanto, tomar decisões na hora de produzir, de forma estratégica (CHAN, 2003). Atualmente, poucas são as referências

³ O termo “rota alternativa de processo” é usado para determinar um conjunto de seqüências de uso de recursos, também alternativas.

ao uso efetivo de rotas alternativas de processo. Os modelos baseados em rotas alternativas de processo usam as rotas alternativas para a escolha, através de métodos de otimização matemática, da melhor rota de processo para cada processo. Estes métodos são considerados estáticos, pois, não permitem que mecanismos alternativos⁴ sejam usados durante a execução dos processos. A estratégia de produção que se adota nesta tese faz referência ao uso de rotas alternativas de processo⁵.

A estratégia de produção é definida nesta tese como o estabelecimento e priorização de critérios competitivos, com coerência interna e externa, procurando-se sempre conectar as atividades operacionais da manufatura com os objetivos maiores da indústria.

De acordo com Corrêa; Gianesi; Caon (2001), os critérios competitivos proporcionam as seguintes vantagens: em custos (fazer os produtos gastando menos); em qualidade (fazer produtos melhores); em velocidade de entrega (fazer o produto mais rápido); em confiabilidade de entrega (entregar os produtos no prazo) e em flexibilidade de rotas (ser capaz de mudar de modo rápido a rota de processo).

A necessidade de atender aos clientes e ao grau de personalização dos pedidos, aliada à necessidade de programação da produção de forma a usar a capacidade produtiva de maneira eficaz, faz com que se busque algo de novo, que supra a essas necessidades.

A causa da maioria dos problemas atuais de sistemas *job shop* pode ser apontada pelas arquiteturas de controle usadas, as quais não possuem características adequadamente dinâmicas para refletir no chão de fábrica suas necessidades⁶, bem como a lacuna⁷ existente entre o sistema de planejamento e o sistema de controle, a qual vem sendo preenchida atualmente por sistemas de execução da manufatura

⁴ Mecanismos alternativos refere-se ao uso de outra rota ao invés da rota preferencial apontada pelos métodos de otimização.

⁵ Rotas alternativas de processo refere-se à possibilidade de uso de seqüências de uso de recursos alternativos, para realização das etapas de um processo qualquer.

⁶ As necessidades a que se refere esta tese são características dinâmicas como a capacidade de reação a eventos como indisponibilidade de recursos, temporária ou não, atrasos na execução das tarefas, gargalos de produção, dentre outras.

⁷ Refere-se à diferença entre o ocorrido na execução das tarefas e o que foi planejado.

(MES – *Manufacturing Execution Systems*) (ARAÚJO JUNIOR et al., 2003) e (ARAÚJO JUNIOR; SANTOS FILHO, 2003).

Na realidade, a produção é uma atividade executada continuamente, no sentido que as tarefas chegam ao *job shop* em lotes (*batch*) ou bateladas. Portanto, a natureza da programação da produção é mais dinâmica do que estática. Em um *job shop* denominado dinâmico, as tarefas não estão disponíveis em lotes ou bateladas no processo de programação. Em um *job shop* dinâmico as tarefas chegam aleatoriamente ao chão de fábrica. De outra forma, cada tarefa tem um instante de chegada no sistema, que corresponde ao instante mais cedo para que a tarefa possa começar a ser processada.

Quando os instantes de chegada das tarefas são conhecidos de antemão, tem-se o problema de programação de tarefas em um *job shop* dinâmico determinístico. No entanto, em um ambiente *job shop* real, em geral, a chegada de uma tarefa ao sistema corresponde a um evento aleatório (estocástico). Assim, não se conhecem *a priori* os instantes de chegada das tarefas nem suas seqüências de operações nas máquinas. Portanto, a programação da produção é não determinística definindo um horizonte de tempo aberto (indeterminismo quanto ao tempo). Isso caracteriza o problema de programação de tarefas em um *job shop* dinâmico não determinístico, doravante abreviado por NDJSP – *non-deterministic dynamic job shop problem*. Os sistemas produtivos com esta característica serão tratados nesta tese por NDDJSS - *non-deterministic dynamic job shop system*.

O conceito de sistema de controle a ser usado nesse trabalho baseia-se no conceito dado pela (MESA - *Manufacturing Execution Systems Association*): é o sistema responsável pelas medições, monitoração, e manipulação da produção, do pessoal, dos produtos e dos processos dentro dos ambientes de processos e de chão de fábrica (MANUFACTURING EXECUTION SYSTEMS ASSOCIATION, 2000). Essa definição estabelece níveis de controle dentro dos níveis hierárquicos constituintes da “pirâmide de automação”, a ser mostrada a seguir. Mais especificamente o controle de processos e o controle de recursos.

Conceitos e definições

Neste tópico pretende-se estabelecer conceitos e definições a serem usados no restante do texto.

“Pirâmide de Automação”

Para melhor compreensão dos níveis de controle a serem abordados nesta tese, faz-se necessário definir a “pirâmide de automação”, seus níveis hierárquicos constituintes e as relações existentes entre esses níveis sob a ótica do modelo de referência PERA⁸ para CIM (*Computer Integrated Manufacturing*) estabelecido pela *Purdue University* (WILLIAMS, 2004). Os principais modelos de referência para CIM, apresentados por Goulart (2000), foram analisados e descartados, exceto o modelo PERA, por que não apresentavam uma hierarquia que favorecesse a divisão por células de manufatura virtuais⁹. A escolha baseou-se na característica principal desse modelo. O modelo PERA é fortemente hierarquizado, característica essa essencial para a atividade de controle, onde a divisão dos níveis de controle apresenta-se de forma adequada à solução pretendida. A solução pretendida envolve a formação de células de manufatura virtuais¹⁰ para horizontes de tempo definidos e o uso de regras heurísticas de despacho como forma de escalonamento em tempo real, que misture características de técnicas de escalonamento pró-ativas e progressivas.

A Fig.1.1 apresenta a “pirâmide de automação” dividida em níveis e uma tabela com a descrição dos níveis hierárquicos e a característica de tempo de resposta de cada nível hierárquico.

⁸ PERA – *Purdue Enterprise Reference Architecture*.

⁹ Célula de Manufatura Virtual é um agrupamento lógico de recursos e de processos em um controlador.

¹⁰ Organização lógica de processos e de recursos para a produção. Esse tipo de organização será explorado no capítulo 2.

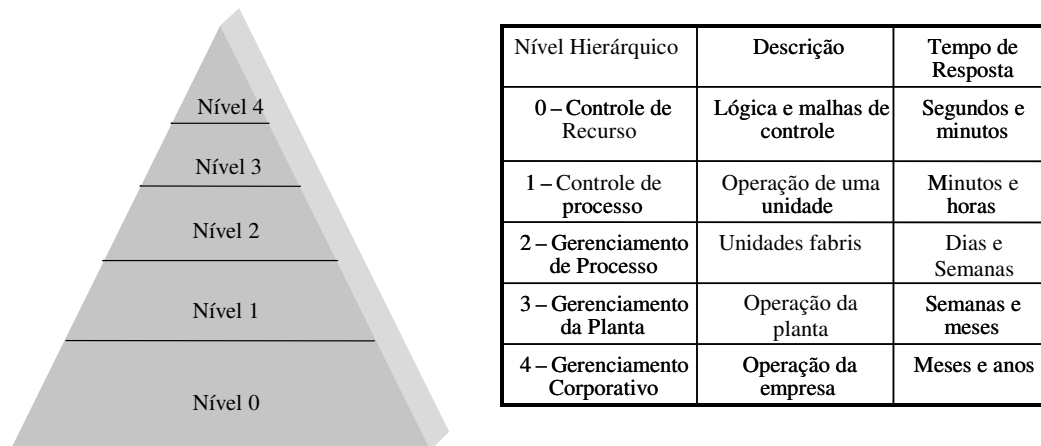


Figura 1.1 – Pirâmide de Automação.

A “pirâmide de automação” nos seus níveis hierárquicos 0, 1 e 2, que correspondem ao controle de recurso, controle de processo e gerenciamento da planta, respectivamente, será usada como referencial para a definição da configuração da arquitetura de controle do NDDJSS desta tese.

Os níveis hierárquicos da “pirâmide de automação” estão embutidos nas camadas da hierarquia do modelo PERA. Esta pesquisa basear-se-á nos níveis 0, 1 e 2 da “pirâmide de automação”, a fim de desenvolver um método de programação da produção de NDDJSS.

Moraes; Castrucci (2001) apresentaram a divisão em níveis hierárquicos da “pirâmide de automação”:

- Controle de Recurso (nível 0): é o nível das máquinas, dispositivos e componentes (chão de fábrica), onde a automação é realizada por controladores programáveis;
- Controle de Processo (nível 1): sua característica é ter algum tipo de supervisão associada ao processo. É o nível onde se encontram concentradores de informações sobre o nível 1 e as (IHM - interface homem-máquina);

- Gerenciamento de Processo (nível 2): permite o controle do processo produtivo da planta; normalmente é constituído por bancos de dados com informações dos índices de qualidade da produção, relatórios e estatísticas de processo, índice de produtividade, algoritmos de otimização da operação produtiva;
- Gerenciamento da Planta (nível 3): é o nível responsável pela programação e planejamento da produção realizando o controle e a logística dos suprimentos;
- Gerenciamento Corporativo (nível 4): é o nível responsável pela administração dos recursos da empresa, em que se encontram os softwares para gestão de vendas e gestão financeira; é também onde se realizam a decisão e o gerenciamento de todo o sistema.

Hierarquia de tomada de decisão e de controle da arquitetura PERA

A arquitetura PERA, a qual foi desenvolvida por um conjunto de representantes da indústria e do meio acadêmico, considera ao mesmo tempo requisitos de informação e funções de controle (WILLIAMS, 2004).

Originalmente, a arquitetura foi concebida para atender indústrias de processo contínuo, porém, ela pode ser usada para todos os tipos de processos dentro da indústria. As informações contidas nas próximas seções baseiam-se no esboço da norma ANSI/ISA-S95.00.01-2000 (INSTRUMENTATION SYSTEMS AND AUTOMATION SOCIETY, 2000).

A Fig.1.2 mostra como as decisões são tomadas nas três camadas do modelo hierárquico funcional PERA. As camadas podem ser entendidas como macro níveis da divisão hierárquica. Os níveis representam divisões hierárquicas dentro das camadas. As camadas do modelo PERA são:

1. Planejamento de negócios e logística (nível 4);
2. Operação da manufatura (nível 3);

3. Controle (níveis 0, 1 e 2).

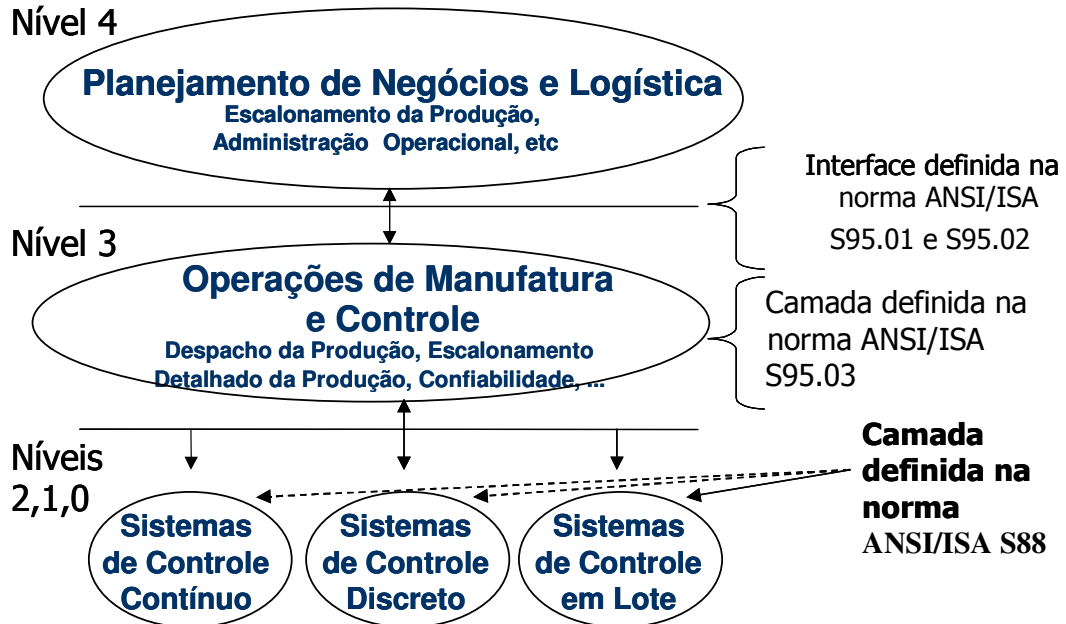


Figura 1.2 - Hierarquia Funcional de Tomada de Decisão e Controle.

Os níveis hierárquicos 3 e 4 reúnem atividades como escalonamento da produção para o sistema produtivo, administração das operações de manufatura e coordenação do chão de fábrica. Já os níveis 2, 1 e 0 envolvem atividades de controle de processo e de recurso para três tipos básicos de manufatura:

1. Batch (batelada ou lote);
2. Processos Contínuos;
3. Processos Discretos.

Essa decomposição define atividades de supervisão para células ou linhas de produção, atividades relacionadas às operações de manufatura, e atividades de controle dos processos e dos recursos. Existem diferentes métodos de execução para essas atividades baseadas na estratégia de produção utilizada, como por exemplo, a atividade de escalonamento estar contida no nível 2.

Nesta tese serão contempladas algumas atividades dos níveis 0, 1 e 2 do modelo PERA e outras atividades na interface entre os níveis 2 e 3 do modelo PERA, visando apenas os processos discretos do tipo NDDJSS. É importante salientar que os níveis 0, 1 e 2 da “pirâmide de automação” contemplam atividades dos níveis 0, 1, 2 e da interface entre os níveis 2 e 3 do modelo PERA.

As atividades dos níveis 0, 1, 2 e da interface entre os níveis 2 e 3 do modelo PERA, que serão contempladas nesta tese, serão detalhadas a seguir, sendo a escolha delas baseada na dinâmica de agrupamento lógico de processos e de recursos (formação de células virtuais) e de geração de despacho através de regras heurísticas em tempo real. A seguir são detalhadas as atividades contempladas nesta tese.

1. *Controle da alocação de recursos*: administrar os recursos diretamente associados com o controle da manufatura. Esse conjunto de recursos inclui máquinas, ferramentas, capacidade de trabalho, materiais, outros equipamentos e outras entidades que devem estar disponíveis para o início e o fim dos trabalhos. A administração desses recursos inclui a reserva de recursos locais para alcançar objetivos de escalonamento da produção, garantia de que o equipamento está adequadamente ajustado para o processamento, bem como a responsabilidade de propiciar tratamentos em tempo real, informação de *status* dos recursos e um histórico de utilização dos recursos.
2. *Despacho (dispatching)*: administrar o fluxo da produção na forma de *jobs*, lotes, quantidades e ordens de trabalho através do despacho da produção para equipamentos e pessoal específicos. O fluxo é governado pela seqüência de operações que determinam a ordem em que o trabalho deve ser feito e os momentos (tempos) de início e fim dos trabalhos. É possível alterar a seqüência ou os tempos em tempo real, na medida em que os eventos ocorram no chão de fábrica; no entanto, essas alterações são realizadas dentro de certos limites, baseado na disponibilidade local, assim como nas condições correntes. O despacho da produção inclui a habilidade de controlar a quantidade de trabalho em processo através da administração de *buffers* e da administração dos processos de recuperação e de retrabalho.

3. *Aquisição e armazenamento de dados*: administrar a produção operacional e os dados paramétricos associados aos equipamentos e processos produtivos, propiciando o *status* em tempo real de equipamentos e processos, e mantendo um histórico dos dados de produção.
4. *Planejamento e rastreamento da produção*: propiciar o *status* da produção e a disposição (andamento) do trabalho. A informação de *status* pode incluir designação de pessoal para o trabalho; material dos componentes usados na produção, condições correntes da produção, e também, alarme, retrabalho ou outra exceção relacionada ao produto;
5. *Análise de desempenho*: propiciar dados instantâneos dos resultados das operações de manufatura correntes juntamente com comparações dos dados históricos e resultados esperados. Os resultados de desempenho incluem a medida de utilização de recursos, disponibilidade de recursos, ciclo de trabalho por produto unitário, conformidade com o escalonamento, e padrões de desempenho.
6. *Operações e escalonamento detalhado*: gerar seqüências que otimizem uma função objetivo (por exemplo: minimização ou maximização) baseada em prioridades, atributos, características, regras de produção associadas a equipamentos específicos de produção e características específicas de produtos. Essa atividade é realizada usando a estimativa corrente da capacidade ociosa, bem como alternativas reconhecidas e operações paralelas e de ultrapassagem.

A seguir são definidas as áreas de responsabilidade para esta tese, que contêm as atividades para os níveis 2, 1 e 0, presentes no modelo hierárquico PERA. A Fig.1.3 mostra a divisão hierárquica para os níveis 2, 1 e 0 proposta na norma ANSI/ISA S88 e usada no modelo PERA. Trata-se da forma como os equipamentos estão divididos dentro da indústria. Informações como localização geográfica, quantidade, tipo, funções que podem desempenhar e etc. fazem parte da divisão estabelecida. A norma define áreas de responsabilidade para isso. Essa divisão proposta na norma não atende à dinâmica que se pretende impor aos sistemas do tipo NDDJSS, além disso,

o tipo de sistema produtivo tratado nesta tese é um caso particular de uma das classes de sistema contemplado pela norma. Sendo assim, a divisão hierárquica dos níveis 2, 1 e 0 foi adotada, somente para a parte das operações discretas, para atender ao caso dos sistemas do tipo NDDJSS tratado nesta tese e será apresentada no momento oportuno na fig.2.7.

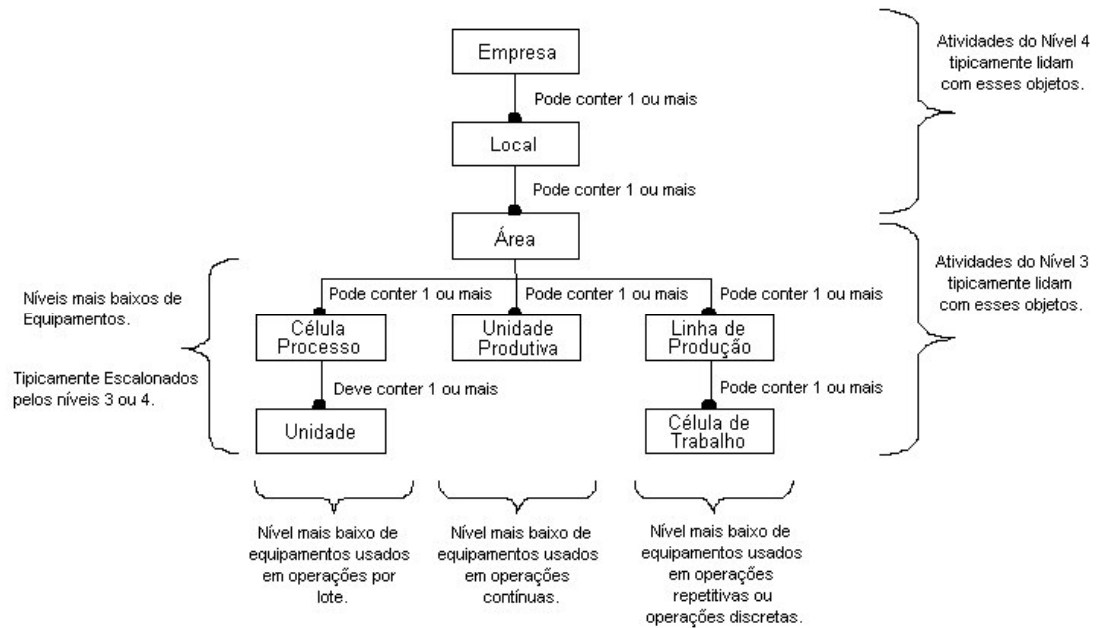


Figura 1.3 - Organização típica para equipamentos previsto na norma ANSI/ISA S88.01.

A seguir é apresentado o detalhamento das áreas de responsabilidade da organização prevista na norma ANSI/ISA S88.01 que interessam a esta tese:

1. *Empresa (enterprise)*: uma coleção de um ou mais locais (*sites*). O nível de empresa é responsável por determinar os produtos a serem manufaturados e onde serão manufaturados. Atividades do nível 4 geralmente lidam com os níveis de empresa e de local. No entanto, o planejamento e escalonamento da empresa podem envolver áreas, células, linhas, ou unidades em uma área. Funções para este nível hierárquico não serão contempladas nesta tese.

Porém, a forma celular é adotada nesta tese para os níveis 0, 1 e 2 e deve ser acatada nos níveis 3 e 4.

2. *Local (site)*: refere-se a um agrupamento físico (geográfico), que é parte constituinte da indústria. Podem conter áreas, linhas de produção, células processo, e unidades produtivas que possuam capacidades de manufatura bem definidas. A administração interna do local, bem como sua otimização, são atividades do nível 4 do modelo PERA. Os locais são geralmente usados para o planejamento e escalonamento “grosseiros”¹¹, os quais podem envolver células, linhas, ou unidades dentro de áreas. Funções para este nível hierárquico não são contempladas nesta tese, visto que a atividade de escalonamento desta tese propõe o uso de regras de despacho em tempo real, portanto, a atividade de escalonamento detalhado será completamente desempenhada no âmbito da camada de controle do modelo PERA, ou seja, nos níveis 0, 1 e 2.
3. *Área*: refere-se a um agrupamento físico (geográfico) que pode conter células processo, unidades produtivas e linhas de produção. A principal capacidade produtiva e a posição geográfica dentro de um local são geralmente usadas para identificar áreas. As áreas geralmente possuem capacidades de manufatura¹² bem definidas. Capacidades essas que são usadas para as atividades de planejamento e escalonamento. Uma área é formada por recursos dos níveis mais baixos que realizam operações de manufatura contínuas, operações de manufatura discretas (repetitivas e não-repetitivas) e operações de manufatura em batelada. Uma área pode conter vários desses elementos em variadas combinações dependendo dos requisitos de manufatura. Nesta tese, as áreas são formadas por organizações de recursos separadas pelo tipo de operação que os recursos podem desempenhar. Portanto, as áreas são vistas como agrupamentos físicos de recursos, separados em conjuntos definidos pelas funções que podem desempenhar e

¹¹ O termo “grosseiro” é usado aqui para denotar o primeiro estágio da atividade de escalonamento, ou seja, quando o nível de detalhamento é ainda superficial.

¹² Capacidade de manufatura nesta tese refere-se às características de recursos como: tipo, quantidade, localização, capacidade e etc...

pela sua localização geográfica, conjuntos esses que são denominados de organizações funcionais.

4. *Unidades Produtivas*: correspondem aos recursos individuais. São compostas de elementos de baixo nível, como módulos de equipamentos, sensores e atuadores. As unidades produtivas possuem capacidade de processamento e produção bem definidas. Essas capacidades são às vezes usadas como entrada da atividade de escalonamento. Esse tipo de organização da produção não é contemplado nesta tese.
5. *Linhas de Produção e Células de Trabalho*: correspondem a agrupamentos de recursos (equipamentos, pessoas, etc...). Geralmente, as células de trabalho são identificadas somente quando existir flexibilidade nas rotas de trabalho dentro de uma linha de produção. As linhas de produção e as células de trabalho são compostas por elementos de baixo nível. As linhas de produção e as células de trabalho possuem capacidade de manufatura e de produção bem definidas. As capacidades são freqüentemente usadas como entrada de atividades de escalonamento, assim como nas unidades produtivas, visto que as linhas de produção e as células de trabalho não são escalonadas por funções do nível 4. O escalonamento ocorre dentro da camada de controle do modelo PERA. Nesta tese, as células de trabalho serão adotadas como forma de organização dos equipamentos no chão de fábrica. Porém, não se trata de uma organização física como as células tradicionais, mas sim, de uma organização lógica em células de trabalho. A organização física será a divisão em áreas compostas por organizações funcionais de recursos.
6. *Células Processo e Unidades*: o nível mais baixo de equipamentos tipicamente escalonados por funções do nível 3 para processos de manufatura em lote (batelada). As unidades são comumente identificadas no nível 3 somente se existirem flexibilidade nas rotas dos processos dentro de uma célula processo. As definições para células processo e unidades estão contidas nas normas padrão IEC 61512 e ANSI/ISA S88.01. As células processo e as unidades possuem competências e capacidades dos lotes bem definidos, as quais são usadas pelas funções do nível 3. As competências e capacidades

podem também ser usadas como entrada de dados para escalonamento do nível 3, sempre que as células processo ou unidades não forem escalonadas por funções do nível 4. Não será abordado nesta tese por não ser adequado à dinâmica que se pretende impor aos sistemas do tipo NDDJSS.

Portanto, serão propostas modificações no modelo de hierarquia de equipamentos da norma ANSI/ISA S88 mostrado na Fig.1.3, para atender aos requisitos de escalonamento desta tese. Como anteriormente mencionado, essas modificações serão mostradas adiante.

A seguir descreve-se as cinco características desejadas para sistemas do tipo NDDJSS, as quais prevêm a adoção de células de manufatura virtuais. São elas:

- **Reconfiguração.** Através da possibilidade de formação de novos arranjos lógicos de rotas de processos¹³ e de agrupamentos de recursos para formação das células de manufatura virtuais.
- **Personificação.** Possibilidade de trabalhar com a produção de produtos personalizados, dentro dos tipos básicos de demanda a serem contemplados nesta tese, ATO, MTO e ETO.
- **Flexibilidade de rotas de processo.** Capacidade de trabalhar com rotas alternativas de processo.
- **Tomada de decisão.** Capacidade de tomar decisões¹⁴ em tempo real dentro do modelo de controle para os níveis 0, 1 e 2 da “pirâmide de automação”.
- **Estratégia.** Definição de um modelo de escalonamento em tempo real baseado em regras heurísticas de despacho e nos critérios competitivos de custo, qualidade, desempenho de entrega e reatividade a serem estabelecidos no item 2.3.

¹³ Nesta tese o termo “rotas de processo” refere-se à seqüência de uso de recursos.

¹⁴ As decisões no contexto mencionado referem-se à designação de recursos para a realização de operações de manufatura e da escolha das rotas adequadas ao momento, dentre as alternativas de rotas possíveis.

A seguir apresenta-se a identificação do problema a ser abordado, a motivação para a solução do problema e o objetivo do trabalho, em questão.

1.1 PROBLEMA ABORDADO

O problema então se refere a como conseguir que sistemas do tipo NDDJSS tenham as cinco características anteriormente apontadas, através de uma arquitetura de controle de células de manufatura virtuais, que atenda às necessidades do mercado¹⁵, dentro de uma dinâmica de despacho em tempo real, obedecendo a divisão em níveis da “pirâmide de automação” para a camada de controle da arquitetura PERA e os respectivos objetivos de controle para os níveis constituintes dessa camada.

1.2 MOTIVAÇÃO

As características básicas de sistemas do tipo NDDJSS, confrontados aos perfis de demanda dos consumidores ATO, MTO e ETO, resulta na necessidade de conceber sistemas de controle, com capacidade de despacho em tempo real (SEIXAS FILHO, 2004).

Considerando a integração entre o sistema de planejamento e o sistema de controle de sistemas de manufatura do tipo *job shop*, as dificuldades concentram-se basicamente no problema de gerar um modelo que represente efetivamente o comportamento dinâmico que se deseja impor ao *job shop*. A maior dificuldade encontrada refere-se a como conseguir que os tempos e seqüências planejados sejam efetivamente executados dentro da característica de desempenho requerida. Sem uma solução de como executar os planos de processo como o planejado, ou seja, dentro das datas devidas, dos custos planejados, com os recursos definidos, etc., não se conseguirá atender as necessidades de mercado dentro dos perfis de demanda estabelecidos, sem nenhum tipo de penalidade, seja de ordem econômica ou que venha denegrir de alguma forma a imagem de capacidade de atendimento dos pedidos da indústria.

Considerando que os sistemas produtivos, particularmente os sistemas de manufatura do tipo *job shop*, devam possuir atualmente características de programação

¹⁵ Os três tipos básicos de demanda identificados anteriormente ATO, MTO e ETO.

dinâmicas, existe a necessidade premente de flexibilidade operacional (MATSUSAKI, 2004). O foco é dirigido para arquiteturas de sistemas *job shop* com processamento simultâneo de múltiplos serviços e produtos, onde se faz presente um intenso compartilhamento de recursos no intuito de manter o desempenho desejado para o sistema.

A partir de então, o problema de controle de um *job shop* passa a caracterizar-se, não mais pela simples tarefa de estabelecer uma seqüência pré-definida para a ocorrência de eventos, mas sim pelo desafio numa dimensão, bem maior, de garantir que o seqüenciamento das operações que definem cada processo seja efetivamente realizado, a despeito do forte compartilhamento de recursos e da natureza complexa e dinâmica do *job shop* (MATSUSAKI, 2004). Além disso, é condição necessária embutir a capacidade de decisão na atividade de escalonamento e despacho, presente na camada de controle desta tese, para conseguir alcançar características como pró-atividade e progressividade¹⁶, necessárias para uma adequação a situações de conflito e imprevisibilidade presentes nos sistemas do tipo NDDJSS. A capacidade de intercâmbio de informação é outra condição necessária e indispensável para uma integração efetiva e para o uso de técnicas de programação dinâmica, programação distribuída e de técnicas dinâmicas de escalonamento e despacho para o *job shop*, pela necessidade de avaliação constante do desempenho do sistema, conforme a dinâmica que se pretende para sistemas do tipo NDDJSS.

Segundo De Paoli; Tisato (1996), o conceito de sistema dirigido por eventos abstrai o tempo como variável independente que comumente é utilizada como referência para descrever o comportamento de um sistema. Entretanto, o conceito de processo global composto por uma série de processos dirigidos por eventos, nos sistemas que são objeto de estudo deste trabalho, reflete-se em um nível de problema em que o seqüenciamento de eventos não mais pode ser utilizado como referencial para descrever o comportamento dinâmico em nível genérico, por si só. De acordo com Vidal; Beck (2003), torna-se necessário associar sistemas dirigidos por eventos a

¹⁶ Os conceitos de pró-atividade e progressividade serão expostos mais adiante no capítulo 3 quando forem tratadas as características das técnicas de escalonamento/despacho. O termo pró-atividade refere-se à capacidade de avaliação dentre várias alternativas para a escolha da melhor. Já o termo progressividade é usado no sentido de olhar à frente para avaliar possíveis mudanças dos cenários futuros para influenciar na decisão do momento atual.

restrições lógicas e temporais, através da inserção dessas restrições ao modelo de controle do NDDJSS, em outras palavras, a regra que rege a dinâmica dos processos no chão de fábrica deve possuir restrições temporais. Como a dinâmica é totalmente dirigida por eventos, as restrições temporais serão responsáveis por gerar eventos que influenciarão na dinâmica do sistema de controle.

Observando-se os trabalhos de Nakamoto (2002) e Matsusaki (2004), vê-se que com o aumento da complexidade do sistema de controle, há uma tendência em adotar como solução uma inserção progressiva de restrições ao que era antes um simples modelo de controle de seqüenciamento de um processo. Este tipo de procedimento resulta em um modelo cuja dificuldade de interpretação e manutenção é cada vez maior, além da sua capacidade de reutilização tornar-se cada vez menor, colocando em uma situação difícil todo o contexto de projeto de sistemas de controle, dentro do escopo de SED (sistemas a eventos discretos). Neste contexto, é essencial contemplar a necessidade de métodos e técnicas para a modelagem de sistemas do tipo NDDJSS minimizando a inserção de restrições e tornando-os mais fáceis de serem modificados, bem como a adoção de formalismos também adequados à modelagem. Em função dos tópicos descritos acima, foi identificado o objetivo desse trabalho, que será descrito no próximo item.

1.3 OBJETIVO DO TRABALHO

Na abordagem desse trabalho, segue-se a linha de outros trabalhos como (SANTOS FILHO, 2000); (MATSUSAKI, 2004) na síntese da operação de sistemas industriais de manufatura.

Em função do problema abordado, o objetivo desse trabalho é a conceber um método de programação de sistemas de manufatura do tipo *job shop* dinâmico não determinístico, através de uma abordagem baseada no conceito de manufatura celular virtual, com células virtuais concebidas através de técnica de computação evolutiva, mais especificamente através de um algoritmo genético, e de uma técnica de escalonamento e despacho em tempo real, que garanta um nível de desempenho estabelecido para o sistema, quanto ao uso da capacidade dos recursos e quanto à flexibilidade do uso de rotas alternativas de processo, sem que haja qualquer tipo de

penalização referente a atrasos dos pedidos e uso reduzido da capacidade dos recursos.

Em consonância com este objetivo, este trabalho baseia-se no cumprimento das metas a seguir.

- Conceber uma técnica de *clustering* para criação das células de manufatura virtuais, garantindo-lhes a característica de independência e de flexibilidade de rotas de processo (a existência de rotas alternativas). A técnica deve possibilitar a análise da frequência de uso para as máquinas (recursos). As células formadas devem possuir a capacidade produtiva necessária para atender datas devidas, com alta frequência de utilização dos recursos e baixo número de recursos dentro da célula.
- Desenvolver uma ferramenta de *software* para simulação de programações e execuções através de cenários gerados aleatoriamente e, também, para a análise de desempenho do *job shop* para os cenários simulados.
- Conceber um algoritmo de escalonamento e despacho em tempo real, baseado em uma regra heurística combinada para conferir-lhe características de escalonamento pró-ativa e progressiva. Ainda dentro desta meta, desenvolver uma ferramenta de *software* (programa de computador) para análise de sistemas do tipo NDDJSS, que permita a simulação de diversos cenários de produção, usando as técnicas desenvolvidas; analisar os resultados obtidos através de simulação¹⁷, usando para isso a ferramenta de *software* desenvolvida e exemplos retirados da literatura técnica, para atestar o desempenho das células virtuais concebidas; e finalmente, apresentar os resultados e as análises finais de maneira a confrontá-los com a hipótese inicial.

¹⁷ A análise por simulação será realizada pela dificuldade de validação da programação através de métodos matemáticos, devido à complexidade matemática inferida ao problema de escalonamento pela explosão combinatória advinda do uso de rotas alternativas de processo.

1.4 JUSTIFICATIVA

A justificativa deste trabalho baseia-se na tendência mundial de busca de melhorias nos métodos de projeto, de trabalho e de produção atuais, com ênfase nos subsistemas de programação, execução e controle. Esta busca é orientada e motivada pelas potenciais vantagens de usar os computadores como ferramentas de modelagem, análise e otimização e não apenas como base de dados.

Além dos aspectos já mencionados, apresentam-se os seguintes aspectos no sentido de completar a justificativa para este trabalho:

- Segundo Lucero (2001), a manufatura do tipo *job shop* em pequenos lotes é comparativamente a menos produtiva dentre os tipos de processos produtivos existentes, fato esse que pode ser revertido com a elaboração de métodos e técnicas mais eficazes para o escalonamento e despacho da produção;
- A necessidade estratégica de flexibilizar-se a função produção das indústrias para atingir os objetivos de demanda por produtos;
- A importância reconhecida pelos estrategistas do papel exercido pela programação dos sistemas de manufatura;
- A falta, dentro da literatura, de métodos sistêmicos que contenham em seu arcabouço o conceito de que se deve otimizar o sistema através dos resultados finais esperados (SEIXAS FILHO, 2000), opinião que se compartilha nesta tese, e não, buscando-se a otimização de cada parte do sistema em separado. Portanto, a necessidade de concepção de modelos que atendam ao perfil apontado.

Dessa forma evidencia-se o interesse por este estudo, pois abordará o problema de como fazer as coisas de forma estruturada, procurando empregar o número mínimo de recursos na transformação da matéria-prima em bens de consumo, para satisfazer as necessidades do mercado consumidor.

1.5 ORGANIZAÇÃO DO TRABALHO

1º Capítulo: Apresenta uma introdução, onde estão descritos o objetivo, a motivação, a justificativa e a organização dos capítulos.

2º Capítulo: Traz uma revisão bibliográfica sobre os assuntos relacionados a sistemas do tipo *job shop* dinâmico, a proposta de adoção do conceito de manufatura celular virtual para alcançar o objetivo traçado no capítulo 1. Então, é apresentada uma nova técnica dinâmica de *clustering* para formação das células de manufatura virtuais, baseado em um algoritmo genético.

3º Capítulo: É apresentada uma nova técnica de escalonamento e despacho para sistemas do tipo NDDJSS que atende ao conceito de manufatura celular virtual. Trata-se de uma técnica de escalonamento em tempo real que usa regras de despacho baseadas em heurísticas. O algoritmo de escalonamento é do tipo *Branch and Bound* e mistura características de técnicas de escalonamento pró-ativa e progressiva.

4º Capítulo: São apresentados exemplos de aplicações das técnicas de *clustering* e de escalonamento e despacho e as análises dos resultados obtidos.

5º Capítulo: São apresentados as conclusões finais e os trabalhos futuros.

A bibliografia encerra este trabalho.

CAPÍTULO 2

TÉCNICA DE

***CLUSTERING* PARA**

FORMAÇÃO DE

CÉLULAS DE

MANUFATURA

VIRTUAIS

Neste capítulo visa-se elucidar alguns aspectos ligados a sistemas do tipo NDDJSS, à caracterização da necessidade de uso de rotas alternativas de processo, propondo a adoção do conceito de manufatura celular virtual juntamente com uma nova técnica de *clustering* para alcançar parte do objetivo traçado no primeiro capítulo. Inicialmente, apresenta-se uma série de conceitos e definições que são condições basilares para a solução que se propõe.

2.1 FLEXIBILIDADE DE ROTAS DE PROCESSO

Segundo Rohde; Boreinstein (2004), a flexibilidade de seqüência refere-se à possibilidade de alternar a seqüência de operações em uma peça, levando em consideração as restrições e especificações de seu projeto. Flexibilidade de máquinas relaciona-se à possibilidade de uma operação poder ser executada em mais de uma máquina. Flexibilidade de roteamento (ou de rotas) refere-se à capacidade do sistema de manufatura de permitir, simultaneamente, a flexibilidade de máquinas e seqüência.

Uma das características essenciais em sistemas de manufatura do tipo NDDJSS reflete-se na capacidade do sistema de produção, como um todo, de adaptar-se da melhor forma possível à demanda por produtos.

Não é tão óbvio assim que a capacidade de lidar com incertezas do tipo restrições temporais e seqüências de operações indefinidas, através da característica de flexibilidade de rotas e políticas de escalonamento e despacho em tempo real, propicie alguma vantagem competitiva às empresas manufatureiras. Muito do sucesso associados às indústrias manufatureiras do Japão e Coréia, por exemplo, tem sido atribuído à adoção de alguns tipos de flexibilidade e especialmente à aplicação de tecnologias avançadas. No entanto, em um estudo comparativo dos sistemas (FMS - *Flexible Manufacturing Systems*) do Japão e dos Estados Unidos atribui-se ao desempenho superior dos sistemas japoneses, não à tecnologia flexível, mas sim em razão do entendimento dos princípios básicos da tecnologia (BEACH et al., 2000). A adoção de tecnologia flexível, como resposta direta a alterações do mercado, não é necessariamente a panacéia que se acredita ser. Em resumo, uma maior flexibilidade nem sempre pode representar a solução mais econômica. Porém em alguns casos, essa pode ser a solução possível, como por exemplo, no caso de sistemas de manufatura do tipo NDDJSS.

2.2 FLEXIBILIDADE EM SISTEMAS DE MANUFATURA

A flexibilidade é definida por alguns autores como “a capacidade de um sistema de manufatura de lidar, efetivamente, com mudanças internas ou externas” (GUPTA; BUZACOTT, 1989).

A definição mais abrangente para o termo flexibilidade, é dada por Upton (1994), que define flexibilidade em sistemas de manufatura como: “a habilidade de se alterar ou reagir com as menores penalidades possíveis em relação a fatores como o tempo, esforço, custo ou desempenho do sistema”.

Segundo Azzone; Beterlè (1991), por causa do grande número de tipos de mudanças (quebra de máquinas, mudança no volume de produção, etc.) é muito difícil encontrar um conceito único de flexibilidade. Conseqüentemente, é necessário

dividir o conceito de flexibilidade em conceitos elementares, que estão associados a certos tipos de elementos e distúrbios de um sistema de manufatura.

A Tabela 2.1 representa uma taxonomia para os tipos de flexibilidade associados a sistemas de manufatura apontados por (BROWNE et al., 1984 apud BEACH et al., 2000).

Tabela 2.1 - Taxonomia de Tipos de Flexibilidade.

Tipo de flexibilidade	Definição
Máquina	Facilidade em realizar as alterações requeridas para produzir um dado conjunto de tipos de peças.
Processo	A habilidade de produzir um dado conjunto de tipos de peças, considerando cada possibilidade de utilização de diferentes materiais, de diversas formas possíveis.
Produto	A habilidade de mudar para produzir um novo produto, ou um novo conjunto de produtos, da maneira mais econômica e rápida possível.
Rota	A habilidade de lidar com quebras de máquinas e condições imprevisíveis, e ainda assim, continuar produzindo um dado tipo de conjunto de peças (alteração da rota da peça).
Volume	A habilidade de operar um sistema de manufatura lucrativamente para diferentes volumes de produção.
Expansão	A capacidade de construir um sistema expansível, quando necessário, de maneira fácil e modular.
Operação	A habilidade de intercambiar a ordem de algumas operações para cada peça (alterar a seqüência de operações da peça).
Produção	Diz respeito ao universo de tipos de peças que o sistema de manufatura pode produzir.

Neste trabalho, o tipo de flexibilidade que se busca é a flexibilidade de rotas de processo da taxonomia de Browne (BROWNE et al., 1984 apud BEACH et al., 2000), para lidar com os tipos de indeterminismos quanto ao tempo e a seqüência presente nos sistemas do tipo NDDJSS.

Diversos pesquisadores têm demonstrado o potencial de melhoria no desempenho de sistemas de manufatura sob condições de flexibilidade (HUTCHINSON; PFLUGHOEFT, 1994); (ROHDE; BOREINSTEIN, 2004). Uma vez que a flexibilidade pode ser um importante atributo no que tange a melhoria dos sistemas, o desenvolvimento de modelos e algoritmos capazes de representarem a flexibilidade é um importante passo para o projeto, avaliação e controle destes sistemas.

Indústrias competentes para dar uma resposta rápida às demandas dos clientes, devem ser alicerçadas por uma infraestrutura em rede de trabalho que possa ligar times em múltiplos locais e ou unidades fabris dentro de uma corporação virtual integrada. Esse tipo de organização também requer que a produção possa responder pronta e rapidamente a alterações na definição de produtos, e às vezes, a mudanças nos processos de produção para os produtos em pequenos intervalos de tempo.

O conceito de “agilidade” é um conceito emergente e é usado nesta tese como a capacidade da indústria de atender às demandas dos clientes com o menor esforço¹⁸ possível, focalizado na pronta resposta a mercados turbulentos e dinâmicos. Nesta tese, um ambiente é caracterizado como turbulento quando possui:

1. demanda e/ou tamanho do lote de produção altamente variável;
2. tempos de processamento altamente variáveis;
3. tempos de preparação altamente variáveis;
4. demanda parcial ou totalmente estocástica;
5. alterações freqüentes do *mix* de produtos;
6. seqüências de produção variáveis;

¹⁸ O termo esforço é usado aqui para denotar os custos associados a produção.

7. competição forte por recursos.

Nos sistemas do tipo NDDJSS, as características que identificam os ambientes turbulentos e dinâmicos, são condições que devem ser atendidas pela camada de controle. Na Tabela 2.2 apresenta-se a relação das metas a serem perseguidas para alcançar as características desejadas em ambientes dinâmicos e turbulentos presentes em sistemas do tipo NDDJSS, nas quais se baseia esta tese.

Tabela 2.2 – Relação entre metas e características de sistemas do tipo NDDJSS.

Sistemas do tipo NDDJSS (metas)	Ambientes dinâmicos e turbulentos (características desejadas)
Arranjo físico distribuído – células de manufatura virtuais	Capacidade de agrupamento de conjuntos de recursos de produção em conjuntos disjuntos para atender à necessidade de produção (sistema orientado pelo tipo de demanda)
Despacho – Rotas alternativas de processo	Aumentar a flexibilidade de rotas de processos, das células de manufatura virtuais, para não perder em capacidade de utilização dos recursos e em possíveis gargalos devido ao número reduzido de recursos no interior da célula.
Despacho – Índice de prioridade para realização de tarefas, baseado em regras heurísticas combinadas.	Geração rápida da escolha dentre as rotas alternativas, visando o critério de desempenho estabelecido, para implementação em controladores em tempo real.
Medida de desempenho – Minimização dos tempos de processamento, do atraso total e confiabilidade de entrega (cumprimento de datas devidas).	Análise de desempenho do sistema.

Assim, este trabalho está focado no desenvolvimento de um método de programação dinâmico para sistemas do tipo NDDJSS em consonância com o modelo PERA.

2.3 CRITÉRIOS ESTRATÉGICOS DE PRODUÇÃO

De acordo com Corrêa; Gianesi; Caon (2001), os critérios estratégicos de produção são os listados a seguir.

1. Custo: mais baixo possível. Nesta tese o custo é a soma dos custos de preparação (*setup*), movimentação, transformação e de espera ao pé da máquina.
2. Qualidade: desempenho de qualidade melhor possível. Representa a satisfação quanto aos requisitos finais para as peças.
3. Desempenho de entrega: confiabilidade e velocidade nos prazos de entrega. Nesta tese, esse critério será garantido pela nova técnica de escalonamento e despacho que será proposta no capítulo 3.
4. Reatividade: capacidade de reação rápida à demanda por produtos. As novas técnicas de *clustering* e de escalonamento e despacho propostas nesta tese garantirão essa capacidade de reação.

Os critérios devem refletir as necessidades dos clientes, mantendo-os fiéis à indústria. Neste trabalho, os quatro critérios estratégicos serão enfocados, sendo os critérios de reatividade, desempenho de entrega e custo, sistematicamente buscados, através de novas técnicas que serão propostas. O critério de qualidade não será tratado de forma sistemática. Adotar-se-á a prerrogativa de que as máquinas escolhidas para a realização das operações de manufatura garantam a conformidade final dos produtos.

Nesta pesquisa, os critérios estão todos “amarrados”. Primeiramente, procurou-se estabelecer uma organização da produção que favoreça a programação dinâmica da produção e o controle dos processos. Baseado nisso, optou-se por trabalhar com células de manufatura virtuais independentes. Essa distribuição lógica facilita o controle dos processos e, juntamente com a nova técnica de escalonamento e

despacho que será proposta, possibilitará o grau de reatividade para o sistema do tipo NDDJSS estabelecido nesta tese. No entanto, o desempenho de entrega, a qualidade dos produtos e o custo tornam-se mais difíceis de serem alcançados, pois o tipo de célula proposto possui restrição quanto ao tamanho¹⁹. Por isso, uma nova técnica de *clustering* será proposta para a formação de células virtuais visando a adequação dessa nova organização aos critérios estratégicos estabelecidos. Para reduzir os efeitos da restrição de tamanho, a nova técnica de *clustering* dará maior flexibilidade de rotas²⁰ aos processos no interior de cada célula, ao permitir que planos alternativos (rotas alternativas de processo) façam parte da etapa de formação das células virtuais e também da etapa de execução dos processos.

Passa-se a estabelecer o elo de ligação entre os sistemas de manufatura do tipo NDDJSS, que são objeto de estudo desta tese, e a nova técnica de *clustering* proposta.

2.4 SISTEMAS DE MANUFATURA DO TIPO *JOB SHOP*

Inicialmente, faz-se necessário apresentar os problemas de *job shop* clássico estático (JSP – *job shop problem*), *job shop* dinâmico determinístico (DJSP – *dynamic deterministic job shop problem*) e *job shop* dinâmico não determinístico (NDJSP – *non-deterministic dynamic job shop problem*). Alguns conceitos e definições relacionados aos sistemas de manufatura do tipo *job shop* também serão descritos.

2.4.1 O problema em SM do tipo *Job Shop* Clássico Estático

Segundo Dos Santos; Araújo Junior (2003), o problema de manufatura em sistemas do tipo *job shop* envolve, dentre outras coisas, a alocação de recursos (humanos, máquinas e materiais), para a realização de tarefas diferenciadas ao longo de um horizonte de tempo, com o propósito de produzir diferentes produtos.

Definição 1 – Cada *job* é um conjunto ordenado de operações executadas em máquinas adequadas, associadas à confecção de um produto.

¹⁹ Refere-se à quantidade de máquinas e aos tipos de máquinas no interior de cada célula virtual de manufatura.

²⁰ Essa flexibilidade refere-se à quantidade de rotas alternativas que podem ser usadas.

Embora a programação de um *job shop* possa parecer um problema simples, no tocante a poder sempre construir uma solução (programação) alocando uma operação por vez, o grande desafio do JSP traduz-se no poder de conclusão de todas as tarefas em um tempo previsto e na otimização de alguma medida de desempenho, na presença de vários *jobs*.

O processo de alocação de recursos oferece dificuldades no tocante ao fato de envolver algumas medidas de desempenho, restrições e preferências com características não claras e conflitantes entre si.

As medidas de desempenho mais comuns aos sistemas do tipo *job shop* são: tempo total de conclusão de todas as tarefas (*makespan*²¹), atraso máximo, atraso médio, número de tarefas atrasadas, atraso total²², dentre outras menos comuns. A fixação da medida de desempenho é preponderante para a escolha do método de solução.

As restrições aos sistemas do tipo *job shop* mais comuns são as seguintes.

- De **precedência**: as operações de cada tarefa têm que ser executadas na ordem prevista;
- De **recirculação**: quando duas ou mais operações de um mesmo *job* devam ser processadas na mesma máquina;
- De **capacidade**: cada máquina pode processar somente uma tarefa de cada vez;
- De **preempção**: cada operação, uma vez iniciada o seu processamento em uma máquina, deve ser concluída sem interrupção.

Portanto, o *job shop* clássico estático é um ambiente formado por um conjunto de máquinas que devem processar um conjunto de *jobs*. Cada *job* possui uma seqüência própria de operações, cujo processamento nas respectivas máquinas demanda intervalos de tempo fixos, embora em situações reais, nem sempre esses tempos são fixos. Todos os *jobs* estão disponíveis para processamento e estão sujeitos às

²¹ O valor ideal de *makespan* corresponde à conclusão de todas as tarefas no menor tempo possível.

²² Conclusão de todas as tarefas com o menor atraso total em relação às suas datas de entrega

restrições de *precedência* e de *recirculação*, enquanto que as máquinas estão sujeitas às restrições de *capacidade* e de *preempção*.

Nesta tese, serão tratadas apenas as restrições de precedência e de capacidade. A medida de desempenho será o *makespan* e a dinâmica de despacho basear-se-á num novo índice de prioridade que será estabelecido no capítulo 3 e que leva em consideração os tempos de processamento de cada etapa dos *jobs*, as datas de entrega e o atraso.

2.4.2 O problema do tipo NDJSP com datas de entrega

O *job shop* clássico estático descreve um processo de produção bastante simplificado, onde os recursos são alocados para uma atividade por vez. No entanto, a produção é uma atividade executada continuamente e, portanto, a natureza da programação é dinâmica. Em um *job shop* denominado dinâmico não determinístico, as tarefas chegam aleatoriamente ao sistema (*shop*).

Segundo Roser; Nakano; Tanaka (2003), o controle das tarefas em um *job shop* dinâmico não determinístico é denominado de reprogramação, porque eventos estocásticos frequentemente exigem revisões na programação. Na literatura o NDJSP é geralmente tratado como se fosse uma seqüência de problemas do tipo DJSP (quando o instante de chegada das etapas é conhecido *a priori*).

Em ambientes *job shop* reais, especialmente os que produzem sob encomenda, o objetivo principal a ser satisfeito é o cumprimento da data de entrega das tarefas. Dessa forma, a minimização de uma medida de desempenho como o atraso, que leve em conta o instante de conclusão das tarefas e suas respectivas datas de entrega, possui relevância maior do que uma medida de desempenho como o *makespan*, por exemplo. Até o momento, são poucas as pesquisas orientadas no sentido de desenvolver métodos de solução para problemas de programação em *job shop* dinâmico, tendo como objetivo tanto minimizar o atraso total das tarefas quanto aumentar o seu desempenho (OLIVEIRA, 2002); (VIDAL; BECK, 2003).

O caminho para alcançar o objetivo traçado passa pelo conceito de manufatura celular virtual, aliada a métodos e técnicas dinâmicas de escalonamento e despacho

para NDJSP. Então, torna-se necessário descrever uma nova técnica de *clustering* para a formação das células de manufatura virtuais. Porém, antes de entrar na descrição da técnica proposta, será abordado o tema manufatura celular como forma de introdução ao assunto.

2.5 MANUFATURA CELULAR

Alguns temas específicos serão aqui abordados com vistas a apontar um marco teórico para esta pesquisa, bem como as relações de interdependência existente entre esses temas dentro do escopo do trabalho. O contexto em questão refere-se aos sistemas do tipo NDDJSS, sob os quais pretende-se aplicar métodos e técnicas associados ao conceito de manufatura celular virtual.

Segundo Mungwattana (2000), historicamente, os sistemas de manufatura passaram por fortes transformações no que diz respeito à organização da produção, aos avanços tecnológicos e à demanda por bens e serviços. Essas transformações, no entanto, vêm ocorrendo de maneira muito mais rápida nos dias de hoje (SEIXAS FILHO, 2004). Porém, as transformações no chão de fábrica não acontecem com a mesma velocidade e intensidade necessárias. A necessidade de métodos e técnicas que permita a indústria manufatureira acompanhar toda essa tendência é um fato indiscutível.

Segundo Drolet; Abdounour; Rheault (1996); Sharp; Irani; Desai (1999), muitos termos têm sido usados para denotar os novos sistemas de manufatura. Dentre eles, podemos destacar a manufatura enxuta, manufatura ágil, manufatura flexível e manufatura inteligente.

Para Matsusaki (2004), os sistemas produtivos (SP) são a classe de sistemas cujos elementos são independentes e interagem com o objetivo de produzir um bem material ou realizar um serviço. Segundo Moreira (2002), os sistemas de manufatura industrial, doravante tratados nesta tese apenas por sistemas de manufatura (SM), os sistemas de saúde, os sistemas de transporte, os sistemas de comunicação, etc. são alguns exemplos de SP.

Os SM são sistemas sensíveis a alterações externas e internas, possuindo um amplo número de graus de liberdade que atuam desde o planejamento e controle da produção, até chegar ao sistema de controle de processos, passando por diversos setores. Tais sistemas podem ser abordados como sistemas a eventos discretos (SED) dado que a dinâmica destes sistemas evolui através da ocorrência de eventos discretos (HO, 1989); (CASSANDRAS, 1993).

A manufatura celular (MC) pode ser apontada como caminho natural para a organização lógica e física dos processos visando uma arquitetura de controle distribuído.

2.5.1 Histórico da manufatura celular clássica

Na tentativa de simplificar o fluxo de trabalho em *job shop*, surge o conceito de manufatura celular, hoje conhecida como manufatura celular clássica (ou fixa). A manufatura celular clássica é definida por Greene; Sadowski (1982) como: “manufatura celular é a divisão física do maquinário e das facilidades em células produtivas. Cada célula é projetada para produzir uma família de componentes. Uma família de componentes é definida como um conjunto de componentes que requer maquinário, ferramentas e sistemas de fixação similares. Os componentes pertencentes à família geralmente vão desde a matéria prima até os componentes acabados produzidos dentro de uma única célula”.

Nesse tipo de organização da produção, os vários tipos de recursos necessário à produção de famílias de componentes são agrupados no chão de fábrica formando as unidades produtivas, também chamadas de células de manufatura. Esse tipo de organização também é conhecido como *cell shop*. Os sistemas clássicos de manufatura celular são o resultado da aplicação da tecnologia de grupo (TG) a *job shop*, sendo a TG um conceito desenvolvido por Mitrofanov, nos anos quarenta, (MITROFANOV, 1966). Porém, somente no início dos anos setenta é que a indústria americana começou a adotar esse conceito de manufatura.

A manufatura celular cria famílias de peças através de técnicas de *clustering* definidas como o agrupamento de peças dentro de conjuntos por similaridade entre

as partes. Essa similaridade pode ser geométrica, funcional, material, pelos requisitos de processo ou ferramental necessário, dentre outras.

O objetivo da MC é alcançar maior eficácia na produção através da exploração de similaridades entre as partes. Uma questão importante no projeto de um sistema de MC é a formação das células de manufatura. O processo de formação das células consiste na identificação das famílias de peças e grupos de máquinas. O objetivo maior no processo de formação das células é a criação de células onde as peças em cada célula sejam processadas com o mínimo de interação possível com outras células.

A formação das células é reconhecida pelos pesquisadores da área como um problema complexo e o estado da arte das técnicas de solução, requer ainda um nível substancial de intervenção humana. A maioria das abordagens propostas para a resolução desse tipo de problema não considera alterações na planta, no projeto do produto, nos métodos de processamento, ou qualquer subotimização como minimização de custos (MUNGWATTANA, 2000). Em contraposição a isso, uma boa alternativa para a formação das células visando maior flexibilidade de rotas de processo, seria a adoção de planos de processos alternativos para as peças. Segundo Wilhelm; Shin (1985), a adoção de operações alternativas pode reduzir o tempo de fluxo, e conseqüentemente o *work-in-process* (WIP), e aumentar a frequência de uso das máquinas, devido a intercambialidade de máquinas.

Segundo Gindy; Ratchev; Case (1996), os requisitos para alcançar uma solução ótima podem ser resumidos em:

1. Maximização do número de componentes na família de peças que requer um conjunto completo de recursos (máquinas) alocados para a célula, ou seja, compactação máxima para o grupo;
2. Minimização da sobreposição da capacidade das células, ou seja, divisão dos componentes em grupos onde as capacidades de processamento repetidas dentro das células sejam minimizadas.

Uma captura adequada das necessidades de processamento dos componentes e das capacidades das máquinas-ferramenta que possam ser usadas para o processamento desses componentes, bem como uma avaliação precisa da similaridade entre componentes é um dos fatores críticos e que influenciam no sucesso dos procedimentos de alocação de componentes para uma célula. Existe uma falta de exatidão em relação aos requisitos de processamento de componentes, quando esses são expressos em termos de rotas de processamento entre as máquinas para os componentes. Essa representação é usada na maioria dos métodos de formação de células. Duas hipóteses são bastante comuns quando da definição de similaridade entre componentes baseado em suas rotas de processamento (SARKER; BALAN, 1996):

1. Cada componente possui uma única (e preferida) rota de máquinas-ferramenta a ser usada para a sua manufatura;
2. Os componentes são considerados similares quanto mais eles visitarem as mesmas máquinas-ferramenta.

A primeira hipótese está se tornando seguramente inapropriada aos sistemas de manufatura atuais. Muitas plantas de manufatura modernas contêm uma variedade de máquinas com alternativas de processamento (flexibilidade de máquina), com sobreposição de capacidades e similaridade entre as máquinas, máquinas essas capazes de produzirem uma grande variedade de componentes. Além disso, as vantagens de se ter sistemas de planejamento que gerem processos alternativos para os componentes, com rotas de processo também alternativas, usando critérios de otimização multiobjetivo para selecionar dentre as alternativas geradas, tem levado à escolha de sistemas com rotas alternativas de processo em detrimento dos sistemas com rotas simples (*single routing*). A seleção da rota final para o componente pode, em muitos casos, ser determinada por outros fatores como restrições ao escalonamento e despacho e ao planejamento da produção determinadas pela disponibilidade, ou não, de máquinas. No entanto, as abordagens por rotas alternativas encontradas na literatura, resume-se à escolha de uma rota simples ao final da técnica de *clustering*, ou seja, são analisadas várias rotas alternativas para a escolha de apenas uma única rota ao final, que atenda ao critério de otimização usado

para a formação das células de manufatura (KUMAR; TIWARI; SHANKAR, 2003). Isso não traz nenhuma vantagem em se tratando de sistemas do tipo NDDJSS. Para tanto, propõe-se uma técnica de *clustering* que permita a formação de células com o uso de rotas alternativas de processamento, ao invés da escolha de apenas uma única rota por peça dentre rotas alternativas de processo analisadas. É necessário ter alternativas de rotas durante a execução dos processos.

Algumas questões devem ser levadas em conta para o agrupamento de peças e máquinas em células de manufatura virtuais, de acordo com os critérios estratégicos estabelecidos para esta tese. Essas questões são:

1. Nas células virtuais, deve-se garantir a qualidade dos produtos ali produzidos;
2. Nas células formadas deve-se reduzir a manipulação de material e melhorar a visibilidade das ações de controle das atividades;
3. Nas células formadas pode-se aumentar o número de máquinas que necessitem de duplicação, o que necessitaria de investimentos adicionais por parte dos administradores.

Portanto, é necessário encontrar uma solução conjunta para essas questões.

Segundo Sarker; Balan (1996), o tamanho das células é freqüentemente determinado pela área do chão de fábrica onde estão localizadas as máquinas. Isso faz com que seja estabelecido um limite máximo para o número de máquinas que uma célula possa ter. O tamanho²³ de uma célula não deve, de nenhuma maneira aumentar muito, de modo que possa obstruir o controle das atividades, nem ser pequena demais de forma que possa aumentar o número total de células, o que tornaria o sistema menos econômico. Quando a quantidade de células é grande, existe grande chance de que algumas operações de um *job* qualquer possam ser alocadas em mais do que uma célula. Nesses casos, as máquinas que estiverem envolvidas com essas operações devem ser duplicadas. Essas máquinas são também conhecidas como “máquinas-gargalo” (*bottleneck machines*). Elas necessitam ser duplicadas, ou então, as operações associadas a essas máquinas devem ser realizadas por terceiros

²³ O termo “tamanho da célula” faz menção ao volume de trabalho no interior da célula.

(subcontratação). Ambas as propostas podem levar a investimentos adicionais, o que freqüentemente não se deseja.

As principais questões consideradas para a criação de células de manufatura são na maioria das vezes contraditórias, no sentido de que se considera como uma “boa” característica possuir o maior número de células, o que faria com que a visibilidade do controle das atividades crescesse e o custo de manipulação de material diminuísse. No entanto, isso pode levar a uma necessidade maior de duplicação de máquinas, o que não é desejável do ponto de vista econômico.

Se o número de células cresce, a manipulação de material diminui, porém o número de operações gargalo e o investimento para administrá-los também aumentam. Então, torna-se necessário encontrar um número de células otimizado que irá minimizar o custo total associado, definido como a soma dos custos de manipulação de material intra-celular, de *setup* para a célula e o custo associado com a realização das operações gargalo.

Em sistemas dinâmicos, como no caso de NDDJSS, essa tarefa se torna ainda mais difícil. Em algumas circunstâncias relacionadas aos sistemas de manufatura, como por exemplo, um sistema de montagem, se cada submontagem é considerada como uma célula, então se torna necessário que as células tenham suas operações findadas antes de um limite de tempo particular a cada célula (data devida ou o *makespan*), de maneira que outras células não sejam afetadas devido a possíveis atrasos em algumas células.

2.5.1.1 Características de Formação de Células de Manufatura

Um dos métodos para a formação das células baseia-se na diagonalização de uma matriz de incidência de máquinas-peças. Uma matriz de máquinas-peças $a = \{a_{mp}\}$ contém a informação dos requisitos de processamento para as peças. O elemento da matriz a , a_{mp} é “1” se a peça “ p ” requer processamento na máquina “ m ”. Um exemplo de matriz de máquinas-peças, a_1 , pode ser visto na Fig.2.1.

		Máquina								Máquina							
		Peça	1	2	3	4	5	6			Peça	1	3	5	2	4	6
a) Matriz a_1	1	1	1	1	1		1		b) Matriz a'_1	1	1	1	1	1			
	2		1			1	1	1		4	1	1	1				
	3		1			1		1		5		1	1	1			
	4	1		1			1			2			1	1	1	1	1
	5		1	1			1			3				1	1	1	1
	6		1			1		1		6				1	1	1	1

Figura 2.1 - Exemplo de decomposição imperfeita para matriz de incidência máquinas-peças ADIL; RAJAMANI; STRONG, 1996).

A matriz de máquinas-peças a_1 foi rearranjada, conforme a nova matriz a'_1 na forma de blocos diagonais para a identificação das famílias de peças e grupos de máquinas, como mostrado na Fig. 2.1. A matriz a'_1 foi particionada em dois blocos diagonais para formar duas células. A qualidade do particionamento obtido pode ser medida através de diferentes cálculos como a medida de agrupamento e eficiência de agrupamento. Essas medidas serão altas quando existir um número baixo de “zeros” dentro de um bloco diagonal, também conhecidos como “voids” (ou nulos) e quando existir uma quantidade baixa de valores “um” do lado de fora dos blocos diagonais, também conhecidos como blocos excepcionais. Os elementos nulos e excepcionais têm implicações adversas em termos da operação do sistema.

Um elemento nulo dentro de um bloco (célula) implica que uma máquina designada para uma célula não é requisitada, em nenhum momento, para o processamento de uma operação de manufatura de uma peça específica, dentro da célula. A presença de elementos nulos faz com que as células formadas tenham grandes medidas de ineficiência. Isso pode contribuir, em outras palavras, para manipulação adicional intracelular de peças.

Um elemento excepcional é criado quando uma peça requer processamento em uma máquina que não está disponível para a célula, em outras palavras, não faz parte do grupo de máquinas da célula. Quando uma peça necessita visitar uma célula diferente para ser processada em parte, a manipulação intercelular aumenta. Além disso, requer-se maior esforço de coordenação entre células.

Outras implicações de elementos nulos e excepcionais podem ser observadas em termos da operação do sistema. Duas implicações mostram-se bastante importantes durante a fase de projeto dos sistemas de manufatura celular, sendo elas: o esforço de coordenação e controle e a manipulação de material intra e intercelular.

2.5.1.2 Esforço de coordenação e controle

A formação das células tem por objetivo gerar células pequenas e independentes, sempre que possível, para aumentar a visibilidade do controle das atividades. Quando as células não são independentes, alguma forma de coordenação entre células envolvendo uma camada de controle num nível hierárquico superior ao nível de controle das células faz-se necessário. Desta forma, é possível coordenar (seqüenciar e sincronizar) a passagem das peças entre células cumprindo o escalonamento local das células, bem como o escalonamento global do sistema produtivo. Os elementos nulos e excepcionais estão intimamente relacionados ao tamanho das células e suas interações (ADIL; RAJAMANI; STRONG, 1996). No sentido de ilustrar melhor essa afirmativa, considerar-se-á a matriz mostrada na Fig.2.1.

A seguinte notação será definida com relação à matriz bloco-diagonal.

c = índice de célula;

d = quantidade de '1' nos blocos diagonais;

e = quantidade de elementos excepcionais na solução;

M_c = quantidade de máquinas alocadas na célula c ;

O = quantidade de '1' na matriz de incidência;

P_c = quantidade de peças alocadas para a célula c ;

v = quantidade de *voids* na solução;

w = peso de um elemento excepcional ($0 \leq w \leq 1$);

$1 - w$ = peso de um elemento *void*.

Então, da matriz a'_i da Fig.2.1, tem-se que:

$$v = \sum_c M_c P_c - d$$

$$v = \sum_c M_c P_c - O + e$$

Considerando-se a minimização da soma dos pesos dos elementos nulos e excepcionais como o objetivo para a formação das células, tem-se que

$$\text{Min } Z = (1 - w)v + we$$

$$\text{ou } Z = (1 - w) \sum_c M_c P_c - (1 - w)O + e$$

$$\text{ou } Z = (1 - w) \sum_c M_c P_c + e$$

desde que “ O ” seja constante para uma dada matriz de incidência.

De acordo com Adil; Rajamani; Strong (1996), esse objetivo equivale a considerar o balanceamento entre o tamanho da célula (número de peças e máquinas) e o número de elementos excepcionais (interação entre as células). Células grandes tornam o planejamento da produção, o escalonamento, e em geral as ações de controle mais difíceis de serem especificados. Por outro lado, a decomposição em células pequenas pode levar à ocorrência de mais elementos excepcionais, requerendo maior esforço de coordenação entre células. Para células de manufatura fixas, esse balanceamento das células, bem como a determinação do tamanho das células e do número de células, é mais claramente vislumbrado. No entanto, para um sistema de manufatura

com células de manufatura virtuais, a quantidade de células e a ocorrência dessas pode se alterar constantemente, o que requer o balanceamento dinâmico para as células, tarefa essa mais complexa, por ser realizada em tempo de execução. Nesta tese, uma nova técnica de *clustering* é proposta para tratar essa questão de forma a garantir o balanceamento das células em relação à carga de trabalho para cada célula dentro de um horizonte de tempo de planejamento.

2.5.1.3 Manipulação de material intra e intercelular

Os elementos nulos e excepcionais podem ser interpretados em termos de manipulação de material intra e intercelulares conforme se observa na Fig.2.2.

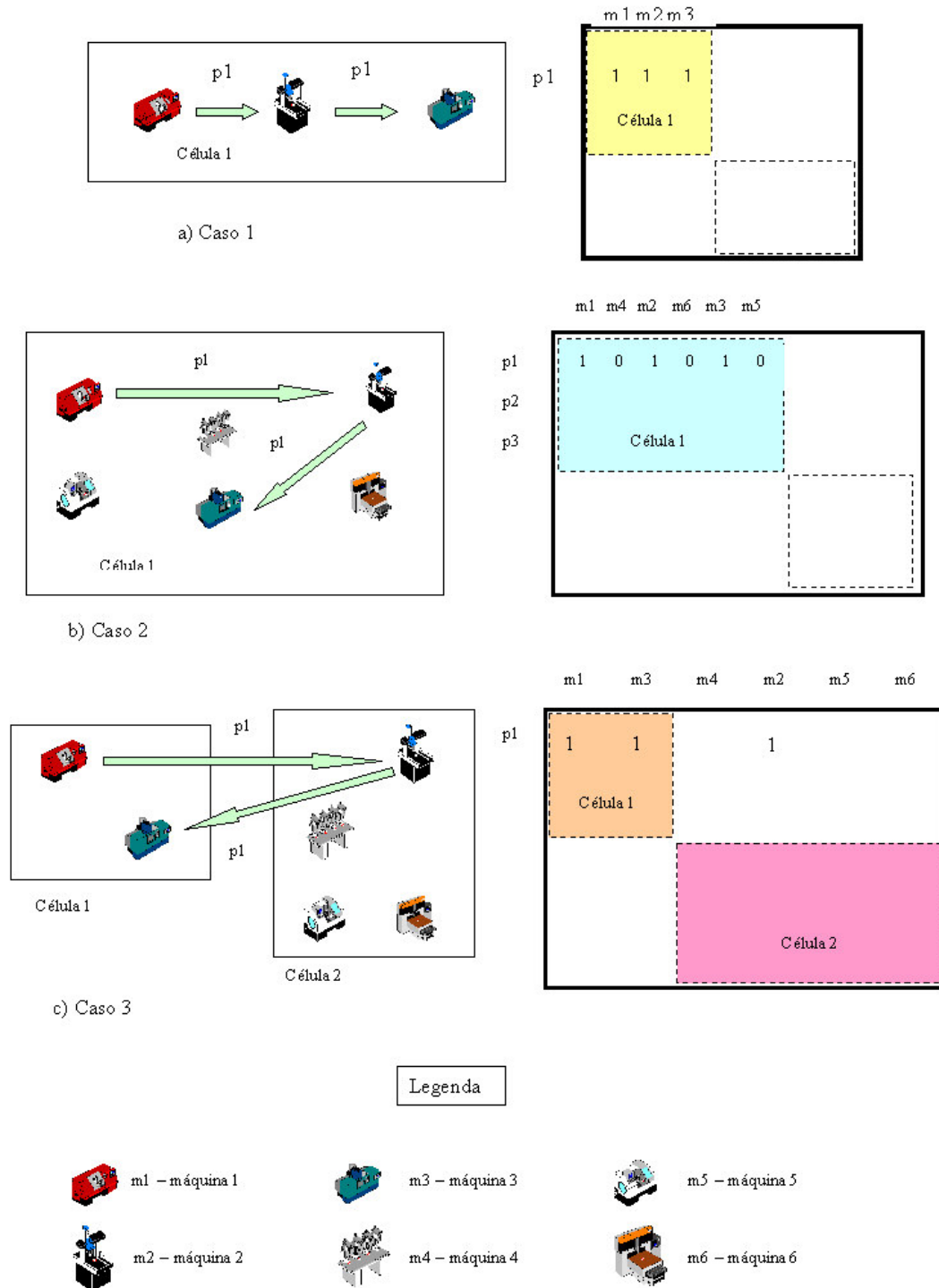


Figura 2.2 - Efeito dos elementos nulos e excepcionais sob a manipulação de materiais.

Caso 1: somente a peça P_1 , a qual requer processamento em três máquinas, m_1 , m_2 e m_3 é alocada para a célula 1. A manipulação intracelular neste caso é mínima e nenhuma movimentação de material intercelular é necessária;

Caso 2: A situação do caso 1 raramente ocorre quando mais de uma peça é produzida numa célula ao mesmo tempo. Neste caso, mais máquinas tornam-se necessárias e devem ser alocadas para a célula. Se, por exemplo, seis máquinas forem alocadas para a célula 1, e o *layout* encontrado for o melhor possível, assim sendo, a movimentação intracelular para a peça P_1 aumenta em relação ao caso 1. Em geral, é verdadeira a afirmativa de que o total de movimentação intracelular de uma peça, para o mesmo número de operações de manufatura, irá crescer junto com o número de elementos nulos na matriz bloco-diagonal (ADIL; RAJAMANI; STRONG, 1996). Então, a presença de elementos nulos pode causar o aumento de movimentação intracelular de material e, portanto, a necessidade de maior manipulação de material dentro da célula;

Caso 3: Considere-se outro cenário, onde nem todas as máquinas necessárias para a produção de uma peça estejam disponíveis em uma célula independente. Por exemplo, considerando o exemplo da peça P_1 , somente as máquinas m_1 e m_3 estejam disponíveis para a célula 1, enquanto que a máquina m_2 é designada para a célula 2. Isso faz com que se tenha movimentação de material entre células. Essa situação representa a presença de um elemento excepcional na matriz bloco-diagonal, bem como a necessidade de coordenação entre células.

Os trabalhos contemporâneos em TG advogam mais flexibilidade com menor comprometimento às características utilizadas no passado para obter o agrupamento (ROSER; NAKANO; TANAKA, 2003). Ben-Arieh; Sreenivasan (1999), analisaram o efeito que a falta de informação ou a chegada de diferentes ordens causam no agrupamento das peças.

No presente trabalho, com o objetivo de avaliar as variações na disponibilidade de informação, é proposta uma técnica de formação de células virtuais baseada em algoritmos genéticos, que permite que as peças sejam agrupadas uma a uma, de acordo com a ordem de chegada ao sistema produtivo, dentro de um horizonte de

tempo planejado. Para isso foi desenvolvido um algoritmo de *clustering*, o qual é responsável pelo projeto das células virtuais. A técnica de formação das células de manufatura virtuais é ao mesmo tempo dinâmica e distribuída, evidenciando ser um excelente caminho de solução do problema de *clustering* em sistemas do tipo NDDJSS, e, ao contrário da maioria dos trabalhos encontrados na literatura, vislumbra a adoção de rotas alternativas de processos em tempo de execução com o objetivo de minimização do esforço de coordenação e controle das células e do esforço de coordenação da manipulação de material dentro das células.

De acordo com Greene; Cleasry (1985), os sistemas de manufatura celular possuem vantagens em relação aos *job shop*, como por exemplo, a redução do sistema de controle necessário, redução da manipulação de materiais, redução dos tempos de preparação, redução do ferramental, redução dos inventários de processos, redução dos fluxos de processo e do tempo associado a esses fluxos, aumento da especialização dos operadores e o aperfeiçoamento das relações humanas no chão de fábrica. Porém, existem algumas desvantagens como a redução da flexibilidade da planta e a redução da utilização das máquinas. Entende-se que este fato deve-se à necessidade de planejar células que atendam à sua capacidade máxima²⁴ e não ao estado real da produção (processos que se encontram em produção no momento), bem como pelo fato de não se usar rotas alternativas resultando numa menor flexibilidade de rotas.

Durante a produção, nem todos os recursos da célula são usados ao mesmo tempo e nem todos os itens são manufaturados ao mesmo tempo. Por outro lado, a proximidade das estações de trabalho, seu número limitado dentro de cada célula e a similaridade dos produtos tornam o planejamento das tarefas muito mais fácil. Porém, quando o *mix* de produtos se altera, os sistemas clássicos de manufatura celular tendem a ser menos eficientes, onde se faz necessário alocar mais recursos para compensar a falta de balanceamentos adequados. Intrinsecamente, as células de manufatura resultantes da aplicação da tecnologia de grupo em *job shop* não são

²⁴ Capacidade máxima refere-se a todos os processos possíveis de serem atendidos pela célula formada sendo processados ao mesmo tempo.

totalmente independentes, aumentando a ordem de complexidade dos sistemas de controle e de planejamento (DOS SANTOS; ARAÚJO JUNIOR, 2003).

Nesta tese propõe-se uma abordagem para tornar as células independentes, facilitando o controle global da planta através da distribuição dos processos em células virtuais independentes, tornando o controle dos processos e dos recursos de uma célula, independente do controle dos processos e dos recursos de outra célula qualquer.

A manufatura celular clássica pode não ser apropriada se a seqüência de operações ou as rotas de processo se alterarem com o tempo. Isso limita seu uso em ambientes onde se necessita a personificação dos produtos e a alteração do *mix* constantes, resultando na necessidade de novos projetos de células (células novas), o que requer tempo e custo associados.

Algumas alternativas à manufatura celular clássica poderiam certamente responder melhor quanto à produção em pequenos lotes para ambientes NDDJSS como, por exemplo, os sistemas de manufatura celular virtual (MCLEAN; BLOOM; HOOP, 1982).

2.6 MANUFATURA CELULAR VIRTUAL

O conceito de célula virtual foi primeiramente introduzido por McLean no início da década de oitenta (MCLEAN; BLOOM; HOOP, 1982). De acordo com a definição dada por McLean, uma célula de manufatura virtual não pode ser identificada como um agrupamento físico e fixo de estações de trabalho no chão de fábrica, mas sim como arquivos de dados e processos em um controlador. Em outras palavras, uma célula virtual é um agrupamento lógico de recursos e de processos em um controlador. Entenda-se por estação de trabalho os recursos necessários para operações completas²⁵ de manufatura, seja esse recurso uma máquina, uma ferramenta, um agente humano, etc.

²⁵ O termo operações completas de manufatura refere-se à execução completa de uma etapa de processo que requer uma dada estação de trabalho formada por máquinas, ferramentas, pessoas e etc.

Na Manufatura Celular Virtual (MCV), quando uma ordem de trabalho é emitida e faz-se necessário o agrupamento de um conjunto de estações de trabalho, o controlador da célula virtual assume o controle dessas estações de trabalho e realiza a comunicação entre elas (RATCHEV, 2001).

De acordo com Drolet; Abdalnour; Rheault (1996), no início da década de noventa ocorreu o marco teórico que permitiu a exploração e o controle de organizações celulares virtuais de maneira a obter altos níveis de desempenho em termos de produtividade, fluxo de tempo para as ordens de trabalho, inventários de processos e flexibilidade frente ao rápido e volátil ambiente de mudanças, devido aos novos sistemas computacionais, à tecnologia de redes de computadores e os sistemas de informação.

A Fig.2.3 ilustra um exemplo de *cell shop*²⁶ com células virtuais. O *cell shop* no momento ilustrado possui duas células virtuais ativas e nenhum recurso compartilhado entre as células. Pelo fato das células virtuais possuírem somente as estações de trabalho necessárias à ordem de produção, a capacidade extra²⁷ geralmente requerida em sistemas de manufatura celular clássicos (fixos) não se faz necessária em um *cell shop* virtual. Com relação aos sistemas *cell shop* virtuais enfatizar-se-á três aspectos importantes da sua configuração física:

1. a variedade de estações de trabalho;
2. a frequência de cada tipo de estação de trabalho;
3. e finalmente, a sua distribuição através do chão de fábrica.

²⁶ Sistema de manufatura baseado em células de produção.

²⁷ Refere-se às máquinas que devem fazer parte da célula para atender a todos os processos possíveis de serem processados na célula, independentemente de possuírem ordem de serviço no momento, ou não.

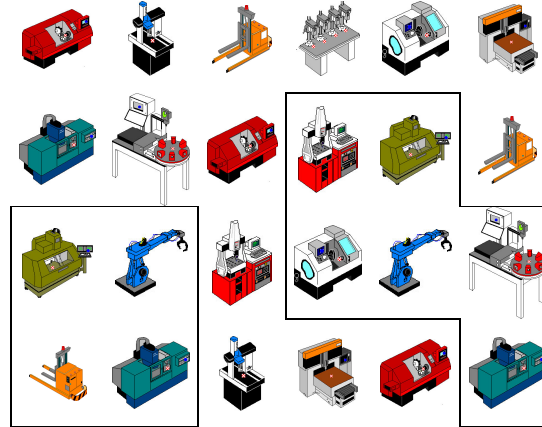


Figura 2.3 – Exemplo de *cell shop* com duas células virtuais ativas e nenhum recurso compartilhado entre as células.

Nesta tese, a variedade das estações de trabalho é função dos requisitos tecnológicos dos componentes para os quais o sistema foi projetado, do tipo e da capacidade das estações de trabalho. A frequência de um dado tipo de estação de trabalho é função da demanda, da capacidade da estação de trabalho e da capacidade para atender as rotas alternativas de processo. A distribuição física das estações de trabalho no chão de fábrica é um fator significativo para a atividade de manipulação de materiais. A organização através de células de manufatura virtuais, proposta nesta tese, apresenta estações de trabalho distribuídas através do chão de fábrica em agrupamentos de recursos com características similares, chamadas de organizações funcionais. Portanto, utiliza-se da TG para a formação das células virtuais através da associação de uma nova técnica dinâmica de *clustering*, partindo-se de um *layout* por organização funcional, o qual divide o chão de fábrica, de forma lógica, em grupos de recursos por similaridade operacional²⁸.

Desta forma, um sistema de manufatura projetado para dar a máxima eficácia a todo tempo e ser capaz de suportar produção sob encomenda a um custo mínimo pode ser obtido, com a adoção do conceito de manufatura celular virtual.

²⁸ Recursos que executam atividades similares.

Espera-se que essa abordagem mostre-se eficiente em ambientes de manufatura turbulentos como é o caso freqüente das indústrias de manufatura subcontratadas, por exemplo, as indústrias de autopeças. Em ambientes reais de manufatura, uma grande dificuldade refere-se a executar os *jobs* como planejado. Indústrias de manufatura de classe mundial estão expostas a um ambiente complexo onde necessidades, processos e produtos estão em constante mudança. Esse ambiente é ao mesmo tempo volátil e dinâmico. Porém, quando seus elementos constituintes começam a se tornar falíveis, esses podem produzir perturbações profundas e turbulências em toda a empresa.

Para um ambiente volátil e turbulento, uma organização funcional (*job shop*) poderia ser a melhor alternativa, porém, significaria trabalhar com altos níveis de inventários de processo e sistemas de rastreamento. Segundo Francis; McGinnis; White (1992), essa não seria a melhor situação especialmente em se tratando da adoção de sistemas *just-in-time* (JIT). Outro tipo de organização seria o sistema de manufatura celular clássico. No entanto, esses sistemas são fixos e rígidos demais para os requisitos de sistemas do tipo NDDJSS. Os sistemas baseados em manufatura celular virtual são sistemas onde a configuração lógica do sistema está predisposta a alterações no tempo. Assim, adota-se a MCV neste trabalho, fundamentando-se no fato da sua configuração física²⁹ ter como objetivo permitir a formação de agrupamentos lógicos ao longo de horizontes de planejamento e pelo grau de flexibilização de rotas que se é capaz de introduzir, visando facilitar o controle da alocação dos recursos aos processos.

2.7 DEFINIÇÃO DA TÉCNICA DE CLUSTERING PARA FORMAÇÃO DE CÉLULAS DE MANUFATURA VIRTUAIS

Uma célula virtual só poderá existir quando os recursos de produção (pessoas, equipamentos e material) forem dedicados à fabricação de famílias de peças ou produtos, e o fluxo de trabalho for conectado em termos de tempos e de informação. Porém, nem sempre esse agrupamento acontece em termos de espaço físico. Para Hyer; Brown (1999), as células virtuais, de fato, são referidas como “dedicação de recursos físicos à produção de conjuntos específicos de bens ou serviços, sem

²⁹ A configuração física a qual se refere o texto é a mesma proposta nesta tese, onde os recursos estão agrupados no chão de fábrica em organizações funcionais que constituem uma área.

rearranjo físico”, e também são reconhecidas como células lógicas. Por exemplo, quando não é possível realocar equipamentos, por causa de fatores como o tamanho, plantas especiais, ventilação ou outro requisito de locação, ou quando um recurso indivisível ou muito caro deve ser compartilhado por diversas células, nesses casos as células virtuais se justificam. As células virtuais ou lógicas também podem ser implementadas como um estágio evolucionário precursor de futuras células fixas. Se o conceito de célula virtual é bem trabalhado, os projetistas e mantenedores das plantas industriais podem, futuramente, optar por um investimento na evolução de células virtuais, para células fixas, principalmente em se tratando de produção do tipo MTS.

É importante criar mecanismos de comunicação eficazes para facilitar a operação e a integração de equipamentos, processos e pessoas, como fator preponderante para o bom desempenho do sistema como um todo. Uma conexão nesses termos permitirá que as células virtuais obtenham muitas das vantagens tipicamente reservadas às células fixas.

No entanto, é importante ressaltar que, quando essas afirmações de que as células virtuais não poderiam trabalhar como as células fixas foram feitas, não se dispunha de muitos recursos de comunicação capazes de oferecer todos os requisitos para a formação de células virtuais. Porém, segundo Seixas Filho (2004), isso é plenamente possível nos dias de hoje.

Em situações onde os equipamentos não podem ser movidos, as células virtuais oferecem algumas vantagens por serem dedicadas. No entanto, elas são geralmente menos eficazes do que as células fisicamente ligadas (agrupadas), por causa dos tempos necessários para a transferência entre operações em equipamentos não próximos, fazendo com que o tempo de produção se prolongue, e que a ausência de contato entre as operações contribua para uma falta de entrosamento físico entre o pessoal envolvido nas operações dentro de cada célula. Por essa razão, Hyer; Brown (1999), observaram que essas células virtuais devem ser vistas como soluções temporárias. No entanto, essa afirmativa se tornou obsoleta devido às novas tecnologias empregadas nas máquinas-ferramenta, que as tornaram parametrizáveis, diminuindo bastante os tempos de preparação (*setup*).

No caso de sistemas do tipo NDDJSS com demandas do tipo ATO, MTO e ETO, essa característica temporária é bastante forte e, portanto, esse tipo de solução temporária é necessário e desejado. Nesses casos, a manufatura celular virtual com suporte em técnicas de *clustering* dinâmicas, algoritmos de escalonamento e despacho dinâmicos baseados em regras heurísticas com prioridade no atendimento às datas devidas, no aumento do fluxo de trabalho e na minimização dos atrasos, faz-se necessário.

2.7.1 Requisitos

Alguns métodos de projeto de células de manufatura têm assumido que cada operação para um tipo específico de peça só pode ser processada num tipo de máquina específica. Segundo Mungwattana (2000), isso não é verdade em se tratando do emprego de máquinas parametrizáveis com capacidade de processamento de múltiplos processos e operações.

O uso dessas máquinas resulta na capacidade de uso de rotas alternativas de processo para cada operação de um *job*. Isso é, cada uma das operações de um determinado tipo de peça tem a possibilidade de ser processada em um conjunto de máquinas da célula com diferentes custos e tempos. Quando um tipo de peça pode ser processado por rotas alternativas através de um sistema de manufatura, está se referindo à chamada flexibilidade de rotas. A flexibilidade de rotas é uma função da flexibilidade de máquinas e da flexibilidade de operações da taxonomia proposta por Browne (BROWNE et al., 1984 apud BEACH et al., 2000), lembrando que a flexibilidade de máquinas refere-se aos vários tipos de operações que uma máquina pode realizar sem requerer um esforço proibitivo na comutação de uma operação para outra e a flexibilidade de operação de uma peça é a sua habilidade de ser produzida (ou processada) de diferentes formas.

A flexibilidade de rotas de um tipo de peça implica que o tipo de peça possui planos de processos alternativos. É importante reconhecer que num ambiente de manufatura do tipo NDDJSS, cada tipo de peça terá mais que um plano de processo se uma ou mais operações puderem ser processadas em máquinas alternativas. De acordo com Rajmani et al. (1990); Oliveira (2002), a designação de uma única máquina para uma

operação de um processo, não propicia necessariamente uma rota ótima, além disso, aumenta o custo e diminui a utilização das máquinas, refletindo-se nas duas maiores desvantagens da manufatura celular clássica. Portanto, nesta tese, o processo de formação das células de manufatura virtuais prevê a análise da frequência de utilização das máquinas, para evitar que máquinas com baixa frequência de utilização sejam introduzidas na célula e, também, aumentar a flexibilidade de rotas. A consideração de planos alternativos de processos pode melhorar o agrupamento dos elementos nos blocos diagonais das matrizes de incidência de máquinas-peças, através da duplicação de tipos de máquinas e ou rotas de processo para as peças.

Os exemplos das Fig.2.4, Fig.2.5 e Fig.2.6 mostram como se pode melhorar o agrupamento dos elementos da matriz através das duas soluções apontadas. Na Fig.2.4a tem-se a matriz de incidência de máquinas-peça original. Após o agrupamento, é obtida a matriz bloco-diagonalizada da Fig.2.4b, a qual apresenta imperfeições no que se refere ao agrupamento, ou seja, a não formação de células de manufatura independentes. Uma solução seria a adoção de planos de processo alternativos, e, após a análise das alternativas, fazer a escolha dos planos mais adequados, dentre as alternativas, visando o objetivo de minimização de movimentação inter e intracelular. Essa solução é feita em duas etapas. Na primeira etapa reorganiza-se a matriz original e encontram-se os elementos excepcionais através de métodos de *clustering*. Após encontrar os elementos excepcionais, elegem-se os planos alternativos (rotas alternativas) factíveis para os planos que contêm esses elementos excepcionais. Numa segunda etapa, todas as combinações das outras rotas com os planos alternativos são testadas e através de medidas de eficiência de agrupamento, escolhe-se a melhor combinação de planos. A Fig.2.5 mostra a matriz reorganizada após escolha das rotas alternativas e a Fig.2.6 mostra a matriz reorganizada após a duplicação de máquinas. Essa solução compreende a adoção de heurística e de técnicas de otimização, como na maioria das soluções encontradas na literatura para utilização de rotas alternativas. Essa é uma “boa” solução, mas não uma solução ideal para sistemas do tipo NDDJSS. Não basta eleger alternativas de rotas e depois trabalhar apenas com a eleita, pois esta prática faz com que a rota eleita passe a ser a rota única quando se passa à execução dos processos. É preciso trabalhar com o conceito “verdadeiro” de rotas alternativas de processo para

a escolha de soluções em tempo real, isto é, deve-se manter as alternativas disponíveis para que durante a execução do processo sejam realizadas as decisões de escolha entre as rotas possíveis.

No entanto, devido à explosão combinatória de soluções e a dificuldade de escalonamento de tais sistemas, será adotada uma solução que, com algumas restrições de quantidade de células, quantidade de recursos e processos no interior de cada célula e de determinação da quantidade máxima de rotas alternativas de processos para cada processo, limita o espaço de soluções, tornando o problema tratável.

		Tipo de Máquina						
		Peça	1	2	3	4	5	6
a) Matriz a_1	1	1	1	1		1		
	2		1		1	1	1	
	3		1		1			1
	4	1		1		1		
	5		1	1		1		
	6		1		1		1	

		Tipo de Máquina						
		Peça	1	3	5	2	4	6
b) Matriz a_2	1	1	1	1	1			
	4	1	1	1				
	5		1	1	1			
	2			1	1	1	1	1
	3				1	1	1	1
	6				1	1	1	1

Figura 2.4 - a) Matriz original a_1 b) Matriz a_2 (decomposição imperfeita da matriz a_1).

		Tipo de Máquina						
		Peça	1	2	3	4	5	6
a) Matriz b_1	1	1	1	1		1		
	2		1		1	1	1	
	3		1		1			1
	4	1		1		1		
	5 (1)		1	1		1		
	5 (2)	1		1		1		
	6		1		1		1	

		Tipo de Máquina						
		Peça	1	3	5	2	4	6
b) Matriz b_2	1	1	1	1	1			
	4	1	1	1				
	5 (2)	1	1	1				
	2			1	1	1	1	1
	3				1	1	1	1
	6				1	1	1	1

Figura 2.5 - Melhoria após decomposição por planos alternativos (rotas alternativas).

		Tipo de Máquina								Tipo de Máquina								
		Peça	1	2	3	4	5	6			Peça	1	3	5	2	2	4	6
Matriz c_1	1	1	1	1		1			Matriz c_2	1	1	1	1	1				
	2		1		1	1	1	4		1	1	1						
	3		1		1		1	5			1	1	1					
	4	1		1		1		2				1		1	1	1		
	5		1	1		1		3						1	1	1	1	
	6		1		1		1	6						1	1	1	1	

Figura 2.6 - Melhoria após duplicação da máquina do tipo 2.

A consideração de flexibilidade de rotas para as peças durante o projeto das células pode melhorar a capacidade de agrupamento das células bem como aumentar a utilização das máquinas como mostrado. Assim, adotando uma posição contrária ao uso de rotas simples, deseja-se agora nesta pesquisa focar na perspectiva de utilização de rotas alternativas e de até onde essa flexibilidade de rotas deve ser levada em consideração no projeto de células de manufatura virtuais.

Durante o estágio de operação das células, é possível que o processamento dos tipos de peça não possa ser realizado como planejado durante a fase de projeto, seja por quebra de máquinas, paradas previstas para manutenção e não consideradas durante a fase de escalonamento, indeterminismos quanto ao tempo (como no caso de uma peça esperando junto à máquina para ser processada) ou por alterações imprevistas no *mix* de produtos e/ou na demanda por produtos. Com a introdução de flexibilidade de rotas, é possível³⁰ reorganizar as rotas para as peças, em tempo real, de maneira que elas ainda possam ser processadas dentro da programação feita para as células de manufatura projetadas, sem que para isso o sistema tenha que ser novamente programado ou ter seus elementos constituintes modificados. No entanto, a performance do sistema pode ser reduzida em relação ao planejamento original. Torna-se necessário então, durante a fase operacional do sistema, estabelecer medidas de desempenho para determinar se o comportamento do sistema se encontra dentro da faixa de desempenho estabelecida. Caso alterações do desempenho

³⁰ Essa possibilidade será mostrada no próximo capítulo. Trata-se da técnica de despacho em tempo real proposta nesta tese.

aconteçam, fazendo com que a medida de desempenho saia da faixa estabelecida, novas alternativas devem ser ativadas com base em regras de despacho. Do contrário, alterações devem ser conduzidas nessa célula, como por exemplo, a introdução de novas máquinas em substituição a máquinas inoperantes ou a restrição de capacidade da célula no caso de impossibilidade de introdução de novas máquinas, ou ainda a transferência de parte do trabalho da célula para outra célula virtual.

O modo de formação das células de manufatura virtuais deve levar em conta uma diversidade de fatores importantes, simultaneamente. Esses fatores traduzem-se na natureza dinâmica e estocástica dos requisitos de produção e na disponibilidade de rotas alternativas (flexibilidade de rotas). Para considerar esses fatores, a solução proposta mescla a complexidade combinatória de modelos matemáticos, aliada a métodos heurísticos para obtenção da solução e a computação evolutiva (algoritmos genéticos).

Portanto, os requisitos básicos para o projeto de um sistema de manufatura celular virtual nessa pesquisa são:

1. formação de células de manufatura virtuais independentes para reduzir a zero o esforço de coordenação entre células de manufatura;
2. adoção de rotas alternativas e tipos de máquinas duplicados³¹ para formação de células independentes e balanceadas (minimização de movimentação intracelular e melhor distribuição da carga de trabalho entre as células virtuais ativas no sistema para um horizonte de tempo pré-fixado);
3. formação de células de manufatura ajustadas ao objetivo de formação de células independentes e com o menor número possível de movimentação intra e intercelular, introduzindo restrições de tamanho das células com o objetivo de obtenção de células balanceadas;
4. introdução de restrições de alocação de recursos, como por exemplo, levar em consideração os escalonamentos previstos para manutenção de equipamentos,

³¹ O termo tipos de máquinas duplicados não se refere à duplicação de um recurso (máquina), mas sim a adoção de máquinas com múltiplas capacidades.

a disponibilidade de equipamentos e de material, a disponibilidade de transporte, etc.;

5. definição e escolha dos recursos e atributos para as células;
6. formação das células virtuais visando o escalonamento global da produção (no nível de células), considerando-se medidas de desempenho como *makespan*, *due date* e *deadline* para as células;
7. formação das células virtuais visando o seu escalonamento interno, levando em consideração medidas como atraso efetivo, *due date* e *slack time* dos processos internos.

2.7.2 Hipóteses para Formação de Células de Manufatura Virtuais

Seguindo-se os requisitos básicos para o projeto de um sistema de manufatura celular virtual, a nova técnica de *clustering* levará em consideração as seguintes hipóteses para a formação de células de manufatura virtuais:

1. formação de conjuntos lógicos (células virtuais) que apresentem o maior número possível de rotas alternativas, com baixo compartilhamento de recursos, priorizando a formação de células virtuais independentes (sem movimentação intercelular) com baixa movimentação intracelular;
2. a escolha do melhor conjunto de rotas alternativas de processo em consonância com os objetivos de minimização dos atrasos, cumprimento das datas devidas, maximização do fluxo de trabalho e o aumento da frequência de utilização das máquinas;
3. a aplicação da técnica de *clustering* deve capacitar o projetista a gerar futuras soluções de controle para as células virtuais (algoritmos de controle), de maneira a diminuir o esforço de coordenação entre as células, ou seja, a solução deve ser modular, distribuída e dirigida a eventos, de forma a possibilitar o uso de técnicas de modelagem de sistemas a eventos discretos.

2.7.3 Nova Técnica de Clustering

A seguir, serão apresentadas as etapas da técnica de *clustering* para a formação das células de manufatura virtuais desta tese. Primeiramente, é necessário fazer algumas observações com relação à adoção do conceito de manufatura celular virtual. Para que seja possível a implementação física, é importante partir-se de um *layout* que propicie a integração lógica e física das estações de trabalho dessas células de manufatura virtuais. É importante ter-se um *layout* onde os recursos (pessoal e equipamentos) estejam distribuídos pelo chão de fábrica em unidades ou organizações funcionais que executem atividades similares. As organizações funcionais propostas nesta tese compõem uma área. Essa organização do *layout* baseou-se na hierarquia de equipamentos da norma ANSI/ISA S88.01 (INSTRUMENTATION, SYSTEMS AND AUTOMATION SOCIETY, 2000). Uma fábrica pode conter mais de uma área geograficamente distribuídas em um mesmo local. A distribuição dos recursos caracteriza-se por ser lógica e simétrica ao mesmo tempo. A Fig.2.7 mostra a organização proposta para esta tese.

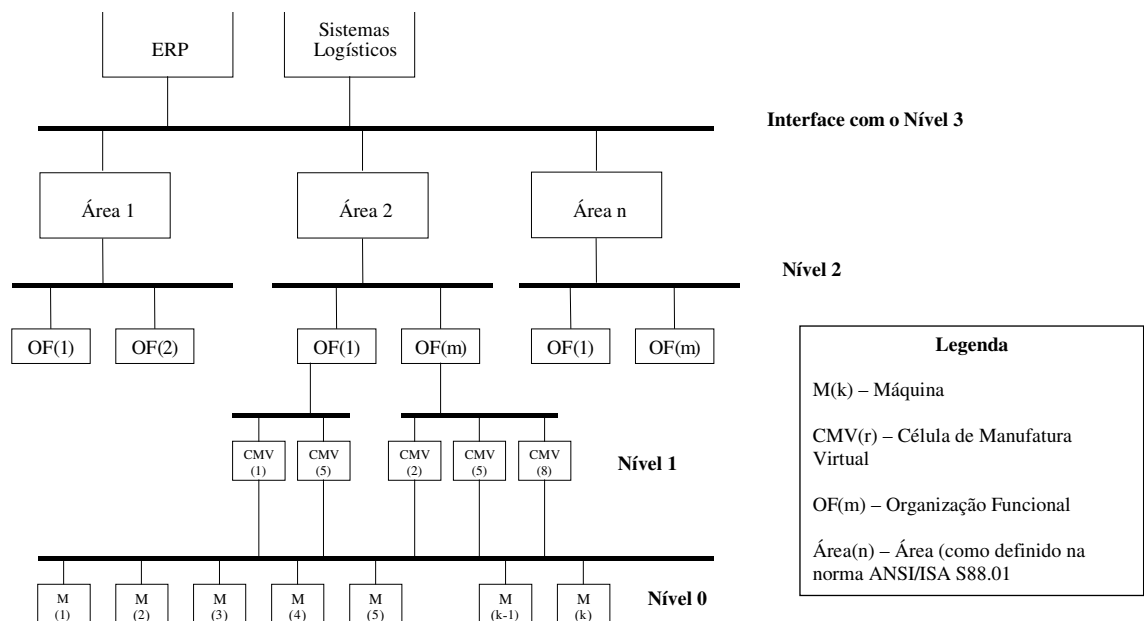


Figura 2.7 – Arquitetura física e lógica para a hierarquia de equipamentos desta tese.

É interessante notar que a formação de qualquer combinação (arranjos lógicos) para as células é inteiramente possível, desde que aconteçam dentro da mesma área.

O método de formação das células de manufatura virtuais que se propõe é decomposto nos seguintes passos:

1. enumeração de todos os recursos existentes no chão de fábrica, bem como as restrições associadas a máquinas e a tarefas, levantamento dos tempos e dos custos associados a esses recursos para a realização das atividades relacionadas aos processos e, por fim, levantamento das restrições de tempo;
2. organização funcional³² do maquinário como proposta de organização lógica e física dos recursos no chão de fábrica;
3. determinação do número de rotas alternativas para cada etapa (operação) de cada processo, e também, do número de programas para o cumprimento das rotas alternativas das etapas de cada processo. O número de rotas para cada etapa de uma peça qualquer é calculado com base no número de etapas para a fabricação de cada peça. O número máximo de rotas por etapa é obtido pela eq.(2.1).

$$Rotas_i = \left\lceil \frac{Etapas}{2} \right\rceil \quad (2.1)$$

onde: $\lceil x \rceil$ significa o menor inteiro maior do que x , i é o índice da peça e *Etapas* corresponde ao somatório de etapas (atividades) para a peça i . Sempre que o número máximo de rotas for igual a 1 (um), adotar-se-á, por convenção, que este número³³ seja igual a 2 (dois), para garantir a flexibilidade de rotas mínima para cada etapa da peça. Essa restrição serve para evitar a explosão combinatória de rotas alternativas e garantir um número pequeno e factível de programações (solução econômica) por etapa para cada peça. A eq.(2.2)

³² O termo organização funcional denota o agrupamento do maquinário no chão de fábrica por similaridade funcional

³³ Cada etapa de um processo deverá conter no mínimo duas alternativas de uso de recursos para que se garanta uma boa flexibilidade de rotas e a garantia de execução da etapa.

estabelece o produto de rotas logicamente possíveis sem nenhuma restrição. Já o número de programas é obtido pela eq.(2.3).

$$\prod_{i=1}^{etapas} r_n \quad n = 1 \dots \text{rotas para a peça} \quad (2.2)$$

$$\sum_{i=1}^{etapas} r_n \quad n = 1 \dots \text{rotas para a peça} \quad (2.3)$$

onde: i corresponde ao índice das etapas, r o número de rotas possíveis por etapa e n é o índice dos processos (peças).

4. determinação do intervalo para a quantidade de máquinas possível de ser usado é dado por:

$$\begin{aligned} & \text{Min}\{\text{Qtd. Peça}, \text{Qtd. Etapas}, \text{Qtd. Tipo Máq.}\} \leq \text{N}^\circ \text{ de Máq.} \\ & \leq \text{Máx}\{\text{Qtd. Peças}, \text{Qtd. Etapas}, \text{Qtd Tipo Máq.}\} \end{aligned} \quad (2.4)$$

5. o número ideal de máquinas é na maioria das vezes impraticável, pela quantidade elevada de máquinas a serem colocadas em disponibilidade para as células virtuais. O número mínimo de máquinas torna-se impraticável, pois não permite que planos alternativos sejam concebidos, usando-se o critério de duplicação de máquinas. É importante lembrar que as células que estão sendo projetadas são independentes. Dessa forma, todos os recursos necessários a todos os planos de todas as peças a serem fabricadas dentro da célula virtual devem fazer parte dela. Dessa maneira, um novo intervalo deve ser usado.

$$\text{Máquinas}_{\text{Min}} \leq \text{N}^\circ \text{ de Máquinas} \leq \text{Máquinas}_{\text{Etapas}} \quad (2.5)$$

6. determinação das células de manufatura virtuais para um horizonte de planejamento (HP). Essa etapa compreende a determinação do número de células para o HP, a quantidade de máquinas (grupo de máquinas) e a determinação dos tipos de máquinas para compor as células de manufatura virtuais, assim como as famílias de peças a serem processadas dentro de cada célula virtual. A partir do processo com maior número de etapas, passa-se a estabelecer a melhor distribuição de frequências de rotas de processo (volume de trabalho). Para isso, é necessário levantar o número de células do HP. O número de células é limitado pela maior frequência de rotas para o HP. Dessa maneira, o número de células virtuais a ser formado para o HP é dado por:

$$F \leq \left\lceil \frac{\text{Frequência Total}}{\max(\text{Frequência de Rotas})} \right\rceil \quad (2.6)$$

$$F \geq \left\lfloor \frac{\text{Frequência Total}}{\max(\text{Frequência de Rotas})} \right\rfloor \quad (2.7)$$

onde, F é o número de células para o HP, $\lfloor y \rfloor$ significa o maior inteiro menor do que y , $\lceil y \rceil$ significa o menor inteiro maior do que y , *Frequência Total* é a frequência de uso para as máquinas considerando-se a soma de todas as frequências individuais para as peças a serem processadas e suas rotas alternativas e *Frequência de Rotas* é a frequência para as rotas alternativas considerando-as individualmente. Para ilustrar esses conceitos, um exemplo é mostrado na seção 2.7.3.1;

7. aplicação de um algoritmo genético para definir quais as máquinas, e rotas de processo farão parte das células de manufatura virtuais;
8. geração da informação necessária ao escalonamento global (referente ao HP para as células de manufatura virtuais) e ao escalonamento local (para as tarefas dentro de cada célula), segundo flexibilidade de rotas, de máquinas e de pessoal previstos para as alternativas pré-selecionadas;

A Fig.2.7 mostra a organização do chão de fábrica adotada nesta tese. Essa organização privilegia a formação de células de manufatura virtuais por permitir o agrupamento lógico de recursos e de processos.

2.7.3.1 Exemplo ilustrativo de aplicação da Técnica de Clustering Proposta

Considere o exemplo de aplicação ilustrado pela Fig.2.8 para mostrar o conceito de MCV aplicado a um sistema fabril hipotético.

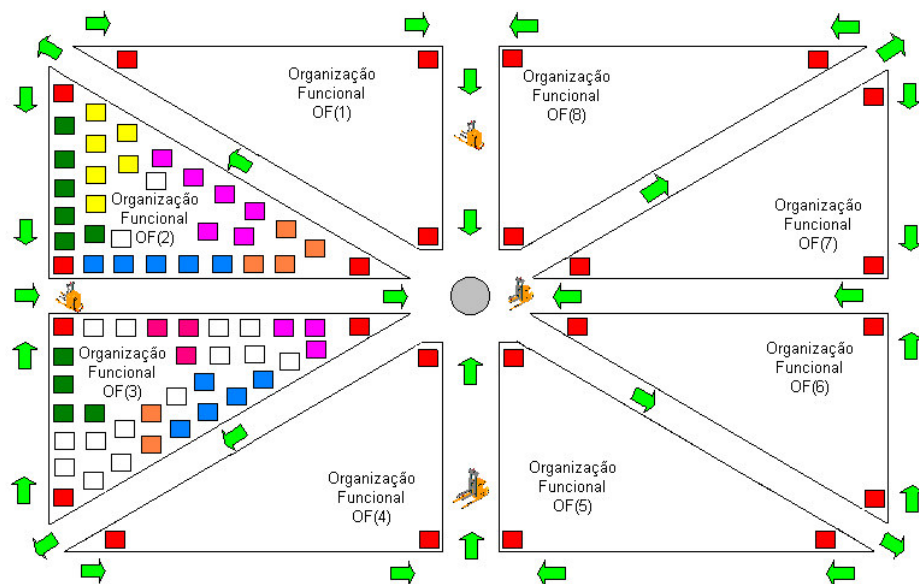


Figura 2.8 - Desenho Esquemático da Planta Industrial Hipotética composta por uma área e oito organizações funcionais.

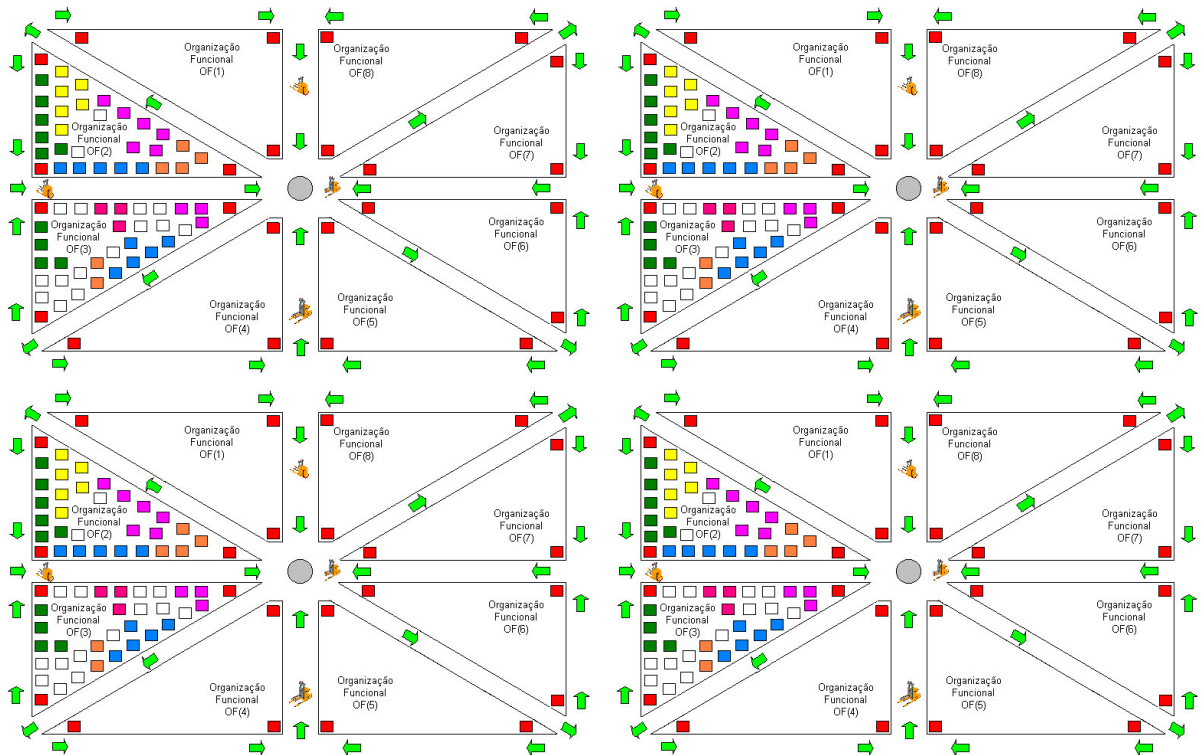


Figura 2.9 - Um local (*site*) formado por quatro áreas.

Descrição do Sistema

A planta esquemática desse exemplo, Fig.2.8, é uma representação hipotética de uma planta industrial, com um tipo de organização que visa facilitar a operacionalização das células de manufatura virtuais, objetivando a organização lógica e física dos elementos constituintes do sistema produtivo.

A planta é formada por uma área contendo 08 organizações funcionais (OF). Essas organizações funcionais representam a separação do maquinário por tipo de operação de manufatura (por exemplo: operações de usinagem, operações de estampagem, operações de soldagem, pintura, transporte, manipulação, etc.). Segundo Rais; Sugimura e Kokubun (2002), essa separação pode ser conseguida através da análise das *features* para os componentes a serem produzidos pela planta industrial, bem como através da capacidade de realização de operações de manufatura pelas

máquinas-ferramenta da planta, não cabendo aqui um estudo sobre essas possibilidades. A quantidade de máquinas-ferramenta de cada tipo, dentro de cada OF, dependerá dos produtos fabricados para estoque (*make to stock*), dos produtos especiais (*make to engineering*) e dos produtos sob encomenda (*make to order*), além, é claro, do número de equipamentos disponíveis para cada tipo.

O modelo hierárquico de equipamentos a ser usado nesse trabalho está representado na Fig.2.10. São apresentados três horizontes de planejamento. Na parte sublinhada está representada a proposta de organização celular virtual desta pesquisa. O HP representado é o HP(1). Essa figura reflete uma fotografia instantânea do sistema para um HP constituído de três horizontes de planejamento: HP(1), HP(2) e HP(3). As células virtuais, bem como sua distribuição ao longo das OF, pode e deve se alterar a todo o momento. Observa-se que as OF representam uma instância superior de coordenação das células de manufatura virtual. Essas OF possuem função de coordenação das células (ou parte das células em seu interior). As questões relativas aos tempos, seqüenciamento e sincronismo, encontram-se já escalonadas mediante o escalonamento *master* da produção. No entanto, imprevisibilidades podem ocorrer, fazendo com que mecanismos controladores orientados por regras de despacho entrem em ação conforme condições e restrições pré-fixadas no escalonamento *master*.

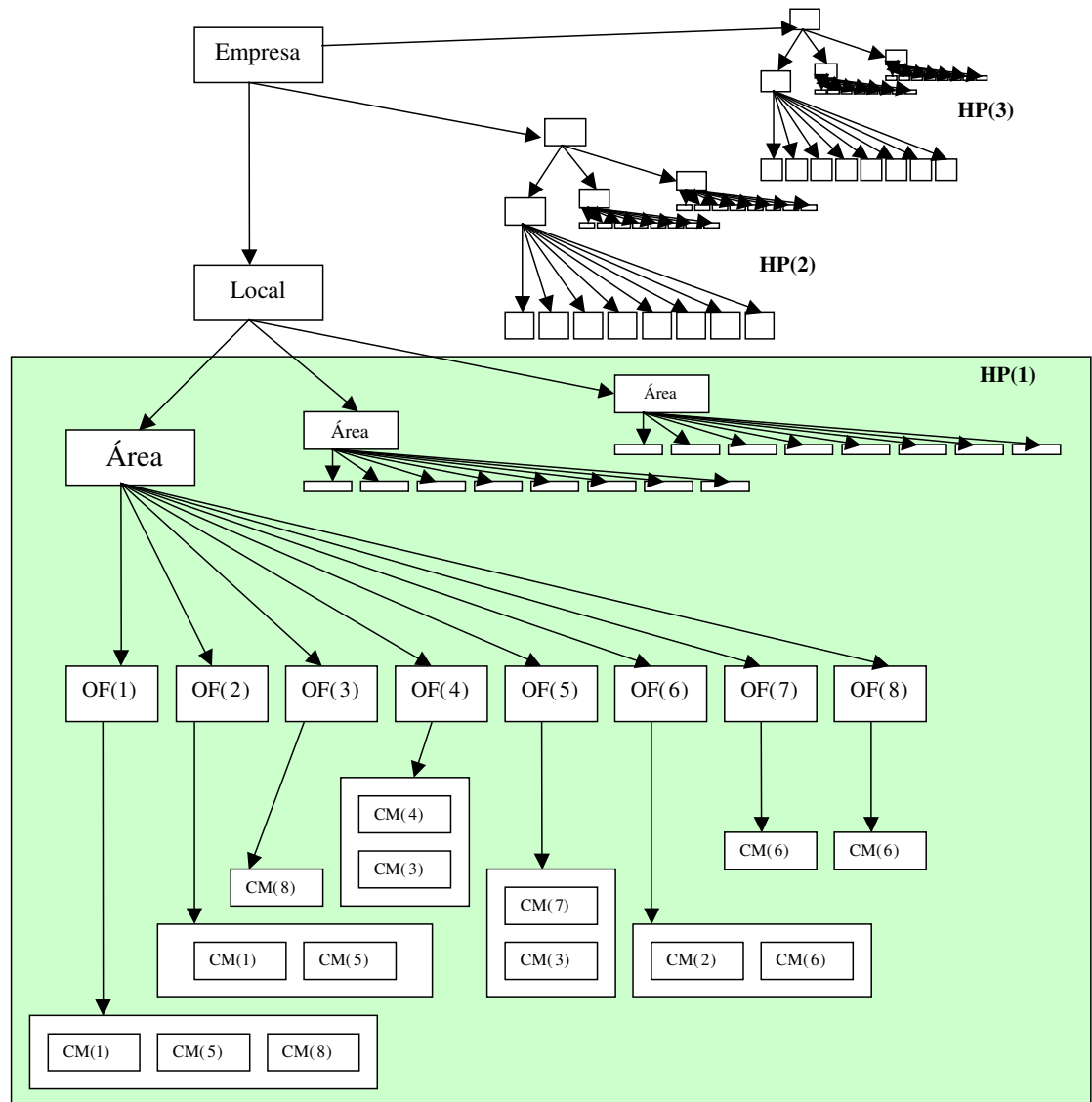


Figura 2.10 - Modelo hierárquico de equipamentos para o HP organizado sob a forma de células de manufatura virtual.

Descrição das Organizações Funcionais do Exemplo

Um exemplo de organização funcional típico é mostrado na Tabela 2.3:

Tabela 2.3 – Uma possível descrição das organizações funcionais do exemplo hipotético.

OF(n)	Descrição	Total de Máquinas na OF(n)
OF(1)	Composta por tornos e fresadoras. Pode realizar operações de usinagem de peças prismáticas e cilíndricas.	80
OF(2)	Composta por fresadoras. Pode realizar operações de usinagem de peças prismáticas.	65
OF(3)	Composta por tornos. Pode realizar operações de usinagem de peças cilíndricas.	32
OF(4)	Estações de Montagem	25
OF(5)	Estações de Inspeção.	30
OF(6)	Estações de Furação.	25
OF(7)	Prensas.	36
OF(8)	Estação de Pintura e Banhos.	20

Descrição dos Processos para um horizonte de planejamento

Os processos são definidos em termos das etapas de transformação e de seus grafos de precedência correspondentes. Os processos de transporte e de manipulação são definidos e controlados separadamente. Para esse exemplo, será considerado o

projeto de um novo HP. O HP é constituído de uma demanda por produtos como o exemplo especificado na Tabela 2.4.

Tabela 2.4 – Uma possível descrição de demanda para um HP.

Horizonte de Planejamento			
Tipo de Peça	Data Devida	Tamanho do Lote	Quantidade de Etapas
“a”	10 dias	5000 unidades	05
“b”	02 semanas	15000 unidades	03
“c”	48 horas	10000 unidades	04
“d”	03 meses	20000 unidades	07

Uma possível demanda por produtos, como o exemplo da Tabela 2.4, deve constituir-se de informações sobre o tipo do produto a ser produzido; a data para entrega do produto; o tamanho do lote de peças a ser produzido a quantidade de etapas (atividades) para a produção, etc...

Os requisitos para a formação das CMV de um HP podem ser vistos na Tabela 2.5. Tomando-se a primeira linha da Tabela 2.5 como exemplo, a primeira e a segunda colunas descrevem treze tipos de peças diferentes com comprimento de tamanho dez. A terceira coluna representa a quantidade de etapas total para as treze peças de comprimento dez. A quarta coluna descreve a quantidade de recursos necessário para processar uma etapa por vez de cada uma das treze peças. A quinta coluna descreve a quantidade de recursos estipulada pelo comprimento das peças. A última coluna descreve a quantidade de recursos necessária para realizar todas as operações de manufatura (por tipo de máquina necessário).

Tabela 2.5 – Um exemplo de requisitos para a formação das CMV de um HP.

Nº de Peças	Etapas por Peça	Total de Etapas	Quantidade Ideal de Máquinas	Quantidade de Máquinas por etapa	Quantidade de Tipos de Máquinas
13	10	130	13	10	06
5	09	45	05	09	08
17	08	136	17	08	05
05	07	35	05	07	04
11	06	66	11	06	05
10	05	50	10	05	05
08	03	24	08	03	03
03	01	03	03	01	01
			Total: 72 Máquinas (Uma máquina para cada etapa de cada peça)	Total: 49 Máquinas (com duplicação)	Total: 37 Máquinas (sem duplicação)

Projeto das Células de Manufatura Virtual

O número de rotas para cada etapa de uma peça qualquer será calculado com base no número de etapas para a fabricação de cada peça. O número máximo de rotas por etapa será dado pela eq.(2.1). O valor obtido é sempre inteiro.

Assim, a Tabela 2.6 mostra o número de rotas alternativas, por etapa, estabelecido para cada peça, bem como a frequência de rotas por quantidade de etapas.

Tabela 2.6 - Frequência de Rotas por Quantidade de Etapas para o HP da Tabela 2.5.

Quantidade de Peças	Quantidade de Etapas/Peça	Quantidade de Rotas/Etapa/Peça	Frequência de Rotas/Etapa
13	10	06	13x10x06=780
05	09	05	05x09x05=225
17	08	05	17x08x05=680
05	07	04	05x07x04=140
11	06	04	11x06x04=264
10	05	03	10x05x03=150
08	03	02	08x03x02=48
03	01	01	03x01x01=03
Total = 72	Total = 49		Total = 2.290

O intervalo para a quantidade de máquinas possível de ser usado é dado pela eq.(2.4).

$$37 \leq N^{\circ} \text{ de Máquinas} \leq 72 \quad (2.8)$$

O número ideal de máquinas é dado pela eq.(2.5). Portanto, um novo intervalo deve ser usado.

Assim, o intervalo a ser usado é:

$$37 \leq \text{N}^\circ \text{ de Máquinas} \leq 49 \quad (2.10)$$

Dessa maneira, o número de células virtuais a ser formado para o HP é dado pelas eq.(2.6) e eq.(2.7).

Portanto,

$$F \leq \left\lceil \frac{2290}{\max(780,225,680,140,264,150,48,03)} \right\rceil \quad (2.11)$$

$$F \geq \left\lfloor \frac{2290}{\max(780,225,680,140,264,150,48,03)} \right\rfloor \quad (2.12)$$

O número ideal de células é o resultado da divisão de 2290 por 780:

$$F = 2,9358 \quad (2.13)$$

assim sendo, o intervalo ideal é:

$$2 \leq F \leq 3 \quad (2.14)$$

A frequência ideal para cada célula pode ser calculada por:

$$fc = \frac{2290}{F_{\text{escolhido}}} \quad (2.15)$$

portanto,

$$fc \approx 763,33 \quad (2.16)$$

De acordo com a faixa estabelecida para F eq.(2.14), optou-se pelo valor de (F = 3) para que se tenha o maior número de células independentes. Para F = 3, a melhor distribuição de frequência³⁴ encontrada é mostrada na Tabela 2.7.

³⁴ Melhor distribuição do volume de trabalho.

Tabela 2.7 – Um exemplo de distribuição das CMV para (F = 3).

Célula(i)	Quantidade de Etapas das Peças	Freqüência Total para a Célula (i)	Quantidade de Máquinas
01	09,07,06,05	779	$10 \leq Total\ de\ Máquinas \leq 19$
02	08,03,01	731	$08 \leq Total\ de\ Máquinas \leq 11$
03	10	780	$07 \leq Total\ de\ Máquinas \leq 13$

Usando a quantidade máxima de máquinas para as etapas das células 01 e 02, tem-se que $M''=30$. Portanto, a opção pela quantidade de máquinas a serem usadas depende de uma análise econômica de momento e da freqüência de uso do maquinário disponível. Para efeito de prosseguimento do exemplo, optou-se pelo limite máximo de máquinas. Isso possibilitará a duplicação de máquinas para geração de planos de processo alternativos (flexibilidade de processos), bem como, possibilitará a adoção de rotas de processo alternativas (flexibilidade de rotas) para adoção de planos alternativos. Já a célula 03, por possuir apenas peças com 10 etapas, optou-se por atendê-las com uma máquina por etapa por peça, ou seja, $M''=13$. Portanto, o número final de máquinas para o HP ficou em $M''=43$, lembrando-se que este número deve respeitar tanto a faixa estabelecida quanto à disponibilidade de maquinário. A Tabela 2.8 mostra a composição das três células quanto aos processos.

Tabela 2.8 - Matriz de Incidência de Máquinas-Peças com as três células formadas.

		Máquinas		
		Célula(1)	Célula(2)	Célula(3)
Peças	p(1) até p(31)	Célula(1)		
	p(32) até p(59)		Célula(2)	
	p(60) até p(72)			Célula(3)

As máquinas a serem escolhidas para a produção das peças do HP dependem das datas de disponibilidade dos recursos. As máquinas devem ser escolhidas para garantir o melhor conjunto de rotas alternativas de processo possível no momento, conforme os requisitos de projeto para as peças.

Os critérios estabelecidos na técnica de *clustering* proposta são:

1. frequência de rotas por etapa para os processos do HP;
2. disponibilidade de máquinas tecnologicamente preparadas para atender as etapas.

Para continuar a descrever a técnica de *clustering* proposta, torna-se necessário introduzir alguns fundamentos de algoritmos genéticos.

Fundamentos de Algoritmos genéticos

Charles Darwin, estudando as espécies e suas evoluções, coletou durante anos uma grande quantidade de material que demonstrou, principalmente, a existência de inúmeras variações em cada espécie. Seus estudos, associados às pesquisas de outros cientistas do assunto, tornaram evidentes que as espécies animais efetivamente se modificam. Um dos principais pontos dos estudos de Darwin foi sem dúvida o aspecto das variações apresentadas entre indivíduos da mesma espécie. Segundo Darwin, todas as novas espécies são produzidas por meio de uma seleção natural (DARWIN, 2003).

No planeta terra, desde há quase cinco bilhões de anos, a natureza vem resolvendo problemas com sucesso. Cada organismo possui *cromossomas*, *genes*, *exons*, *íntrons* e *códons*, constituindo um sistema genético. Um determinado grupo de indivíduos vive junto, constituindo uma população. Nesta população há os organismos melhores, que são os que têm mais chance de casar e gerar bons descendentes. Estes descendentes são mais bem adaptados do que a média da população, pois receberam melhores *genes*. Ao final, vence a lógica, de sobreviver o mais adaptado ao nicho ecológico da população, o que garante que este sistema de escolha com sucesso seja

passado às gerações subseqüentes, melhorando cada vez mais as populações envolvidas.

Segundo Serrada (2004), a Computação Evolutiva (CE) é um ramo da ciência da computação que se embasa em um novo paradigma para a resolução de problemas, que não exige o conhecimento de uma sistemática prévia de resolução, e baseia-se nos mecanismos encontrados na natureza, à luz da teoria da evolução natural de Darwin. A Computação Evolutiva constitui um ramo da Computação Natural, e envolve tópicos de vida artificial, geometria fractal, sistemas complexos e inteligência computacional. Fazem parte dos estudos deste campo os Algoritmos Genéticos, as Redes Neurais e os Sistemas Especialistas.

Os Algoritmos Genéticos representam um paradigma de aprendizado pela máquina. Seu funcionamento encontra inspiração em um dos mecanismos básicos da evolução da natureza, chamada seleção dura. Estes algoritmos foram inicialmente desenvolvidos pelo professor John Holland, da Universidade de Michigan, nos Estados Unidos da América, em suas explorações dos processos adaptativos de sistemas naturais e suas possíveis aplicabilidades em projetos de *softwares* de sistemas artificiais. Eles foram formalmente introduzidos no livro de (HOLLAND, 1975).

A idéia dos Algoritmos Genéticos é mais antiga, como reconhece o próprio Holland, referindo-se a trabalhos anteriores e ainda outras abordagens semelhantes³⁵. As pesquisas realizadas por Holland e seus discípulos na Universidade de Michigan tinham as seguintes metas:

- explicar de forma rigorosa e abstrata o processo evolutivo dos sistemas naturais;
- desenvolver um programa computacional que reproduzisse o importante mecanismo de solução de problemas empregado pelos sistemas biológicos.

³⁵ Para melhor entendimento desses outros trabalhos anteriores, aconselha-se reportar-se diretamente a eles através da leitura do livro do prof. John Holland (HOLLAND, 1975).

Para se entender, em sua plenitude, este algoritmo que Holland resolveu chamar de Algoritmo Genético, é conveniente que se conheça um pouco dos processos biológicos observados na evolução natural.

Algoritmos genéticos básicos são os que realizam, segundo Whitley (2004), as seguintes funções:

- inicializa a população de cromossomos (soluções);
- avalia cada cromossomo (indivíduo) da população;
- cria novos cromossomos a partir da população atual (aplica mutação e cruzamento, substituindo os ascendentes pelos descendentes);
- termina, se o critério de fim for alcançado, se não, reinicializa.

Um processo de otimização consiste em melhorar o desempenho, com o objetivo de alcançar um ou vários pontos ótimos. É desta forma que funcionam os AGs. Eles combinam a sobrevivência do mais adaptado, com uma troca de informações ao mesmo tempo aleatória e estruturada. Os AGs trabalham sobre uma população de palavras ou cromossomos (*strings*), que são seqüências de códigos, geralmente de forma binária podendo possuir outro formato, que representam determinados parâmetros. Para se criar uma população de palavras, são aplicadas sucessivas operações de reprodução, cruzamento e/ou mutação.

A seguir são apresentadas as três operações básicas de um AG: reprodução, cruzamento e mutação.

Reprodução

A reprodução em um AG simples é o processo no qual as palavras ou cromossomos individuais são copiados de acordo com os valores dados pela função objetivo "*f*", que na Biologia se denomina de função aptidão (*fitness*). Esta função "*f*" é a medida de utilidade que se deseja maximizar. Os cromossomos com altos valores de *fitness* possuem maior probabilidade de formar a próxima geração. Este operador é um modelo artificial da seleção natural. Nas populações naturais a aptidão (*fitness*) é

determinada pela capacidade que os organismos têm de lutar pela sobrevivência, resistindo a predadores, doenças, intempéries, falta de alimentos e outros obstáculos. No meio artificial, a função de avaliação é que decide quais os cromossomos que sobreviverão e quais morrerão.

Cruzamento

O cruzamento ou corte (*crossover*) ocorre em duas etapas, sendo a primeira onde se determina, aleatoriamente, quais os cromossomos que casarão e, em seguida, também aleatoriamente, em que posição do cromossomo se processará o cruzamento (*crossover*).

Mutação

A mutação é uma alteração aleatória e ocasional do valor de uma posição qualquer do cromossomo. Esta alteração ocorre de acordo com uma probabilidade prefixada e, por exemplo, no caso de um cromossomo binário, poderia significar a mudança de "1" para "0" ou de "0" para "1". Esta probabilidade de mutação, como a probabilidade de cruzamento, e também o tamanho da população, são elementos muito importantes nos AGs. Eles controlam todo o processo de busca, influenciando diretamente a velocidade de convergência, e evitando que aconteça a supremacia de uma determinada subpopulação, o que geraria o chamado elitismo.

Segundo Goldberg (1989), os AGs têm quatro aspectos que os diferem dos outros métodos tradicionais de otimização, e que são:

- trabalham sobre uma codificação de parâmetros e não diretamente sobre os parâmetros do problema;
- operam em uma população e não em pontos isolados, reduzindo assim o risco de busca a falsos pontos;
- usam informações de uma função objetivo e não derivadas de outros conhecimentos auxiliares;

- procedem a busca utilizando operadores escolásticos³⁶ e sua regra de transição é probabilística e não determinística.

A Fig.2.11 representa um fluxograma simplificado das operações sequenciadas nos AGs mais simples.

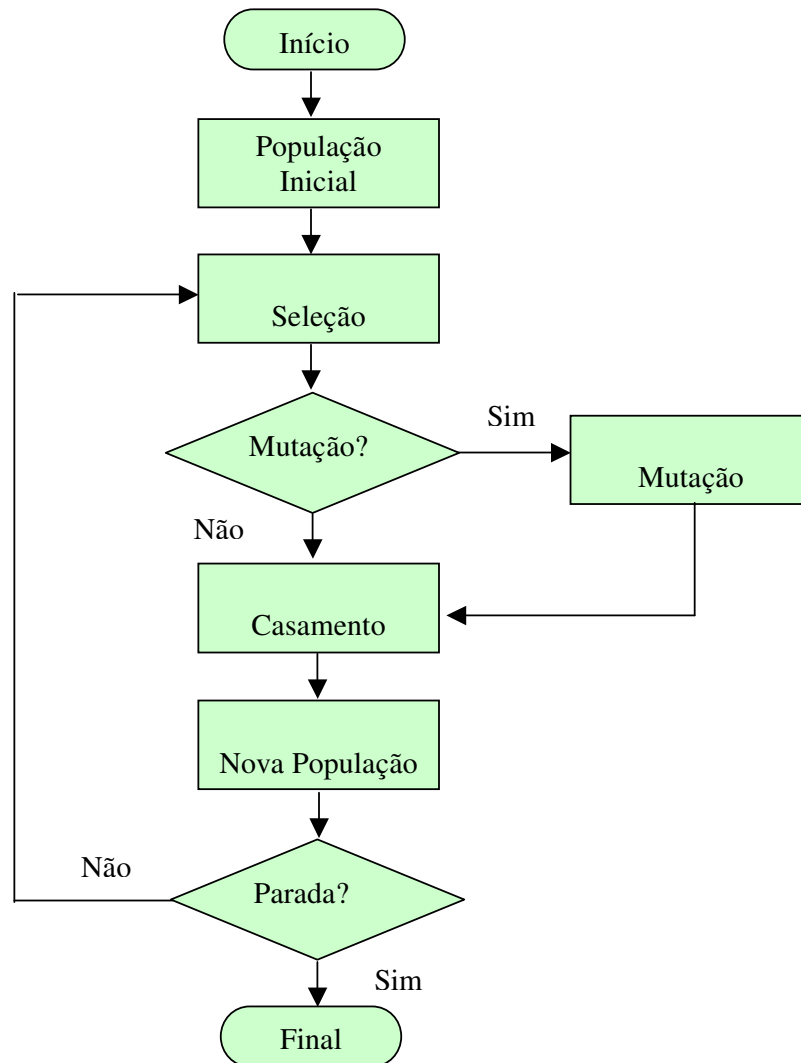


Figura 2.11 - Fluxograma simplificado do ciclo de um AG.

Os AGs, ao usarem ao mesmo tempo operações aleatórias e estruturadas, operam de forma singular uma troca de informações entre os cromossomos, o que emula de certa maneira a forma de descobrir e pesquisar do ser humano.

³⁶ Uso de regras lógicas e disputas.

Na busca pela diminuição do esforço computacional e de modelos matemáticos mais simplificados, os AGs, segundo os estudiosos, simplificam a solução, porque não há a necessidade de se trabalhar sobre todos os dados do problema; basta que se conheça o que deve ser maximizado e quais são as variáveis que devem ser julgadas. Além disso, este operador também permite que se interrompa o processamento para ver as soluções já atingidas, e ainda permite que nesta altura possam ser feitas mudanças para otimizar ainda mais o processo.

Proposta de AG para o problema de formação das células virtuais desta tese

A seguir, passa-se a apresentar um algoritmo que, com o uso da programação genética, promove a busca de aproximações ótimas para o problema do agrupamento de máquinas-peças em células com o uso de rotas alternativas, considerando-se como entrada uma tabela similar à Tabela 2.9 da pág.98.

O AG proposto resultará numa saída quase ótima para as rotas alternativas de processo através de um arranjo de máquinas-peças. A Tabela 2.9 inicial prevê todos os casos possíveis de uso de máquinas para realizar as operações específicas de todas as peças previstas para um horizonte de planejamento³⁷. Uma outra tabela, no mesmo formato, será usada para representar o resultado da aplicação do AG desenvolvido nessa tese para a escolha das melhores rotas, as quais serão usadas pelo escalonador do sistema para escolha em tempo real das rotas mediante um índice de prioridade estabelecido pela regra de despacho, a qual é estabelecida no capítulo 3.

Aplicação dos algoritmos genéticos

Como descrito anteriormente, de acordo com Goldberg; Luna (2000), os AGs diferem dos diversos métodos heurísticos por apresentar um grupo de características distintas: operar em um conjunto de pontos (população) e não a partir de pontos isolados; operar em um espaço de soluções codificadas e não diretamente no espaço de busca; necessitam como informação, somente o valor de uma função objetivo (função de adaptabilidade, ou *fitness*); usarem transições probabilísticas e não regras determinísticas.

³⁷ Refere-se a uma janela de tempo planejada para a execução dos planos de processos.

Em linhas gerais, uma população inicial é criada (o cromossomo de cada indivíduo é criado aleatoriamente) e o valor de adaptação (*fitness*) é calculado para cada indivíduo. Operadores genéticos são aplicados a indivíduos selecionados probabilisticamente, com base nas suas aptidões, e uma nova geração de indivíduos é criada. Contudo, a evolução de novas gerações também é conduzida pela inserção de pares de cromossomos na população corrente usando *crossover* e *mutação*. Apenas os cromossomos que apresentarem melhores *fitness* terão mais chances na seleção. Este procedimento irá se repetir até que o critério de finalização seja atingido. O algoritmo proposto para esta tese será mostrado mais adiante de forma elucidativa.

Na aplicação de AG ao problema específico desta tese, considera-se uma função de aptidão que tem valor proporcional ao inverso do produto do somatório dos tempos de processamento das tarefas dos *jobs* pelo somatório dos custos associados às tarefas e pelo número total de máquinas envolvidas nos processos, sendo que o número total de máquinas não pode ultrapassar a um número máximo de máquinas estabelecido para a célula de manufatura virtual.

$$f = \left(\frac{1}{M \cdot \sum p_{ij} \cdot \sum C_{ij}} \right) \quad \{ \forall : M \leq M_{m\acute{a}x} \} \quad (2.17)$$

onde:

f = função objetivo (aptidão ou *fitness*).

M = quantidade de máquinas necessárias para realização dos *jobs* de um cromossomo.

$M_{m\acute{a}x}$ = quantidade máxima de máquinas admitida para a célula virtual³⁸.

$\sum p_{ij}$ = somatório de todos os tempos das tarefas escolhidas para um cromossomo.

³⁸ M não poderá ser nunca maior que $M_{M\acute{a}x}$. Isso faz com que o indivíduo (cromossomo) não possa fazer parte da população final, pois, o número de recursos empregados é maior que a quantidade máxima de recursos admitida para a célula de manufatura virtual.

$\sum C_{ij}$ = somatório de todos os custos das tarefas escolhidas para um cromossomo.

Os cromossomos denotam soluções factíveis para o problema, e seu comprimento será igual ao produto da quantidade de peças da célula pelo número de etapas do processo de maior comprimento da célula.

A partir da Tabela 2.9 da pág.98, extrai-se aleatoriamente os cromossomos (indivíduos) da população inicial. Os cromossomos do problema da tese são representados pela Fig.2.12. Cada *gene* do cromossomo equivale a um *job* e cada *locus* denota a etapa, respectivamente. Assim, ficam duas informações contidas em cada posição da *string* do indivíduo. Dentro de cada *locus* fica registrada a máquina escolhida para a etapa correspondente.

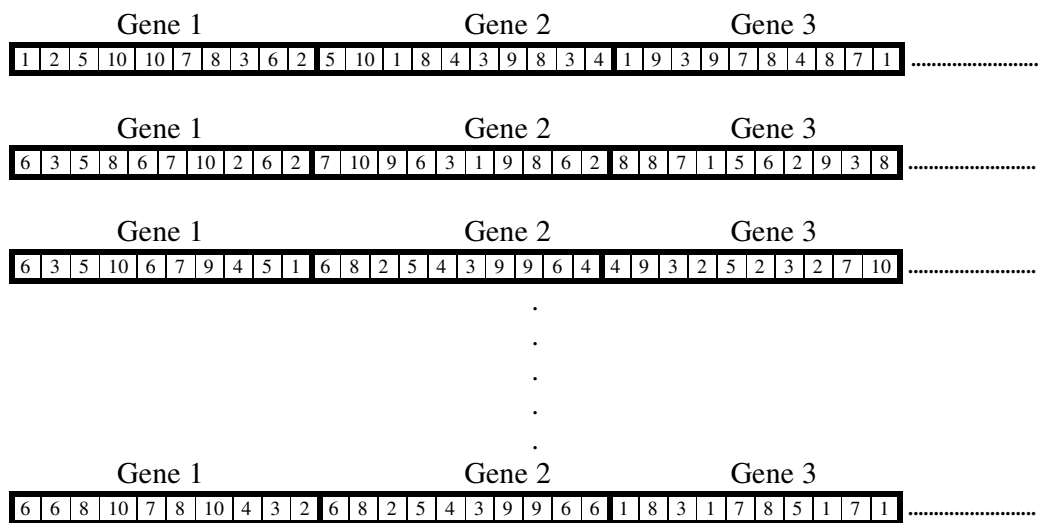


Figura 2.12 – Um exemplo de representação da população de indivíduos para o problema proposto na Tabela 2.9 da pág.98.

Na Fig.2.12, os cromossomos são representados pelas possíveis rotas de processo para cada peça. Os *genes* representam uma possível rota. Os cromossomos completos representam um conjunto de possíveis rotas para o conjunto de peças. A população inteira de indivíduos representa um conjunto de indivíduos que satisfaz a restrição de $M_{máx.}$ para a célula virtual.

Uma vez que se dispõe de uma tabela que organiza as máquinas envolvidas em cada etapa, já ponderadas em seu número máximo, cada posição do cromossomo indicará portanto o índice das máquinas escolhidas para a execução das tarefas. Torna-se evidente que o tempo de processamento para cada máquina estará associado com o futuro cálculo do *fitness*, outrora descrito na eq.(2.17).

É importante associar que cada peça terá reservado no cromossomo o tamanho Q , onde Q representa o comprimento do maior processo da célula, mesmo que a peça tenha um número de etapas inferior a Q . Nos casos onde o comprimento dos processos for menor que o comprimento da palavra que compõe cada *gene*, será arbitrado um valor numérico (zero) que será desconsiderado no cálculo do *fitness*. Importante ressaltar também que este procedimento não interfere nos *crossovers* ou mutações subsequentes.

Conforme mencionado, dois operadores são aplicados no AG: *crossover* e mutação. O *crossover* determina o mecanismo de combinação de dois cromossomos existentes e cria dois filhos. Já a mutação, refere-se a mudanças aleatórias nos *genes*, para escapar de máximos locais e garantir acesso a qualquer solução do espaço de soluções.

Importante ressaltar que existirá um *crossover* para cada grupo Q de *genes* do cromossomo, garantindo que para cada rota de fabricação possível, de cada peça, ter-se-á um cruzamento que permitirá excursionar pelas várias respostas factíveis de rotas para o problema, na busca de um ou mais indivíduos otimizados para o problema.

Zolfaghari; Liang (2002) investigaram os efeitos do tamanho da população, taxas de mutação e números de pontos de *crossovers* na qualidade das soluções. Observaram que grandes populações e pequenas taxas de mutação tendem a melhorar a performance dos AGs na solução de problemas, em detrimento ao fato de que o número de pontos para *crossovers* não afetou significativamente tal qualidade. Ainda segundo os referidos autores, este trabalho considera a sugestão a seguir:

- tamanho da população³⁹, $K = 5Q$;
- taxa de mutação, $pm = 2\%$;
- número de pontos de *crossovers*, $N^c = 1$.

Rodrigues; Branco; Rosa (2003) consideraram as mesmas taxas apresentadas acima na busca de soluções do seu problema que envolvia o uso de AGs para compor agrupamentos celulares fixos de máquinas-peças, com base em matrizes de incidência, porém, sem o uso de rotas alternativas, alcançando bons resultados quanto ao agrupamento de recursos e processos em células fixas de manufatura.

O algoritmo genético relativo ao problema do agrupamento de máquinas para esta tese é descrito a seguir, conforme Zolfaghari; Liang (2002) e Goldberg (1989):

Passo 1: Inicialização

Ler o tamanho da população K e taxa de mutação pm .

Inicializar cromossomos pela geração de tantas soluções factíveis quanto o tamanho da população.

Passo 2: Cálculo do *fitness*

Calcular os valores de *fitness* de cada indivíduo da população.

Passo 3: Seleção dos pais

Selecionar aleatoriamente dois⁴⁰ cromossomos da população, considerando a probabilidade de escolha associada ao *fitness* de cada um.

Passo 4: Geração de descendência

³⁹ Para o AG desta tese, considerar-se-á para efeito de cálculo do tamanho da população, o número de rotas logicamente possíveis, o qual poderá ser dividido por um fator de escala (α).

⁴⁰ Esses dois cromossomos são os cromossomos que irão cruzar para gerar os cromossomos filhos.

Empregando o operador *crossover*, gerar dois cromossomos a partir dos pais selecionados no passo 3.

Passo 5: Fim da geração de descendência

Repetir os passos 3 e 4 se o tamanho da geração de descendentes for $< K$; caso contrário, ir para o passo 6.

Passo 6: *Mutação*

Para cada indivíduo da população, varrer os elementos de cada cromossomo, modificando-os aleatoriamente, com probabilidade p_m .

Passo 7: Cálculo do *fitness*

Calcular o *fitness* para os cromossomos descendentes.

Passo 8: Finalização

Caso o critério de finalização tenha sido alcançado, parar; caso contrário, dirigir-se ao passo 3. O critério usado para parada, refere-se ao número de gerações estabelecidas.

Ao final da aplicação do AG, obtém-se os indivíduos da população (rotas escolhidas) e as máquinas necessárias para a realização das rotas escolhidas, dentro da restrição de número de máquinas estabelecida para a célula. Ainda assim, é necessário fazer um estudo da frequência de utilização para as máquinas arbitradas pelas rotas escolhidas para a melhor geração de indivíduos, para saber se é necessário a retirada de máquinas e uma nova busca pelas melhores rotas sem essas máquinas. Passa-se então a descrever a tabela com as informações necessárias à aplicação da técnica de *clustering* desta tese.

A descrição das tarefas básicas para um HP é dada na forma da Tabela 2.9, onde n é o índice da célula virtual, m é o índice da máquina, p_{ij} é o tempo de execução da etapa i na máquina j (onde o tempo p_{ij} representa o somatório dos tempos de

preparação, movimentação, transformação) e o índice r corresponde ao número de rotas alternativas para a etapa indicada.

Tabela 2.9 – Exemplo de Descrição de Rotas Alternativas de Processo da Célula (n=3).

Célula (n=3)	Etapa (1)			Etapa (2)			Etapa (3)			Etapa (4)			Etapa (5)			Etapa (6)			Etapa (7)			Etapa (8)			Etapa (9)			Etapa (10)		
	m	p _{ij}	r	m	p _{ij}	r	m	p _{ij}	R	m	p _{ij}	r	m	p _{ij}	r	m	p _{ij}	r	m	p _{ij}	r	m	p _{ij}	r	m	p _{ij}	r	m	p _{ij}	r
Peça (1)	1	30	3	3	50	3	5	42	3	1	88	3	10	20	3	7	54	2	8	15	3	1	78	4	1	12	4	1	45	2
	2	40	3	2	50	3	7	30	3	8	55	3	6	30	3	8	50	2	9	44	3	2	89	4	6	23	4	2	48	2
	6	60	-	6	20	-	8	25	-	10	60	-	7	25	-	-	45	-	10	38	-	-	3	56	3	21	-	-	-	-
Peça (2)	5	10	3	10	35	3	1	20	3	8	15	3	3	64	3	1	12	2	5	90	2	8	45	3	1	88	3	2	75	3
	6	12	3	8	35	3	2	40	3	6	15	3	4	58	3	3	15	2	9	80	2	9	35	3	6	55	3	4	70	3
	7	20	-	4	35	-	9	20	-	5	20	-	10	89	-	-	-	-	-	-	-	2	10	30	3	60	6	6	80	-
Peça (3)	8	45	3	8	10	2	7	55	2	1	32	3	5	30	3	8	05	4	2	88	4	1	45	6	3	66	2	1	73	3
	4	35	3	9	14	2	3	33	2	2	33	3	6	30	3	6	05	4	3	87	4	2	46	7	7	60	8	8	70	-
	1	30	-	-	-	-	-	-	-	9	31	-	7	30	-	4	09	-	4	60	-	3	40	6	-	-	2	10	80	3
Peça (4)	9	02	3	4	32	3	10	45	3	1	50	3	8	10	2	1	15	3	7	66	3	4	88	3	5	23	4	4	44	2
	10	05	3	3	40	3	6	35	3	2	50	3	4	10	2	10	15	3	5	30	3	8	55	3	6	20	4	5	40	-
	7	08	-	2	45	-	3	30	-	4	20	-	-	-	-	9	12	-	6	52	-	9	60	-	7	20	-	-	-	-
Peça (5)	5	68	3	7	15	2	10	40	2	9	40	3	4	79	3	2	45	3	1	32	3	4	02	3	1	20	10	12	14	2
	8	56	3	2	15	2	1	40	2	6	40	3	7	80	3	5	35	3	2	36	3	6	05	3	7	25	2	4	14	-
	3	49	-	-	-	-	-	-	-	5	40	-	10	85	-	9	30	-	10	45	-	8	08	-	8	08	-	8	20	-
Peça (6)	1	10	4	1	20	4	6	12	2	1	56	3	7	18	4	1	15	2	8	36	2	2	90	3	4	12	8	46	48	3
	2	10	4	3	20	4	7	18	2	4	55	3	9	32	4	7	16	2	6	41	2	3	98	3	6	15	7	7	48	-
	6	10	-	4	20	-	-	-	-	6	50	-	5	30	-	-	-	-	-	95	-	4	95	-	7	31	10	49	-	-
Peça (7)	3	23	3	1	80	2	6	50	2	9	80	3	5	12	3	10	10	3	4	12	2	8	20	2	6	02	2	2	89	3
	3	20	3	7	65	2	9	23	2	3	50	3	7	19	3	1	23	3	7	54	2	7	25	2	5	05	3	4	65	-
	4	20	-	-	-	-	-	-	-	6	61	-	10	41	-	2	45	-	-	-	-	-	-	-	3	08	8	78	-	-
Peça (8)	5	10	2	1	32	2	9	66	2	8	73	3	7	44	3	1	50	3	9	88	3	9	35	2	7	45	4	54	36	2
	7	25	2	2	45	2	5	45	2	9	84	3	5	45	3	8	50	3	7	55	3	10	27	2	8	69	2	5	36	-
	-	-	-	-	-	-	-	-	-	10	91	-	3	44	-	4	20	-	5	60	-	-	-	-	-	-	-	-	-	-
Peça (9)	3	79	3	1	52	2	5	32	2	1	02	3	5	63	2	8	99	2	7	33	2	9	12	2	5	15	3	3	50	3
	2	80	3	2	46	2	4	30	2	2	05	3	6	45	2	6	95	2	9	36	2	10	23	2	6	12	3	5	50	-
	1	85	-	-	-	-	-	-	-	3	08	-	-	-	-	-	-	-	-	-	-	-	-	-	8	19	9	20	-	-
Peça (10)	8	97	3	3	10	6	79	3	23	3	12	5	45	7	56	8	23	8	50	8	23	8	50	4	30	30	3			
	10	85	2	4	45	2	7	80	3	7	20	4	9	36	2	8	78	2	10	63	3	9	20	3	5	50	3	2	23	-
	-	-	-	-	-	-	8	85	-	8	20	-	-	-	-	-	-	-	9	78	-	10	20	-	9	50	1	15	-	-
Peça (11)	4	33	2	5	45	2	1	18	2	4	76	2	1	90	3	9	02	3	6	12	2	3	79	3	1	20	8	32	36	2
	8	35	2	8	42	2	7	19	2	6	23	2	9	52	3	2	05	3	7	36	2	4	80	3	6	25	3	3	36	-
	-	-	-	-	-	-	-	-	-	-	-	-	10	87	-	5	09	-	-	-	-	10	85	-	9	20	-	-	-	-
Peça (12)	9	40	2	9	40	2	2	56	2	8	23	2	4	90	2	5	15	2	8	32	2	4	50	2	2	40	10	20	30	2
	6	63	2	10	45	2	7	35	2	9	45	2	6	78	2	8	36	2	9	30	2	6	50	2	5	42	2	1	30	-
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Peça (13)	6	79	3	7	88	3	3	88	3	5	50	3	3	20	2	10	20	2	3	50	3	8	74	2	6	64	6	80	85	2
	5	80	3	8	55	3	7	55	3	6	50	3	6	30	2	7	35	2	2	40	3	9	56	2	7	62	2	9	85	-
	3	85	9	6	67	10	10	60	10	10	20	-	-	-	-	-	-	-	1	60	-	-	-	-	-	-	-	-	-	-

Tabela 2.10 - Totalização do número de rotas logicamente possíveis e do número de programas necessários para o exemplo da Tabela 2.9.

Célula (n=3)	Produto de rotas logicamente possíveis (sem nenhum tipo de restrição)	Somatório de programas para cumprimento de todas as etapas e suas respectivas rotas
	$\prod_{n=1}^{etapas} r_n$	$\sum_{n=1}^{etapas} r_n$
Peça (1)	= 3 x 3 x 3 x 3 x 3 x 3 x 2 x 3 x 4 x 4 x 2 = 46.656	= 3 + 3 + 3 + 3 + 3 + 3 + 2 + 3 + 4 + 4 + 2 = 30
Peça (2)	= 3 x 3 x 3 x 3 x 3 x 3 x 2 x 2 x 3 x 3 x 3 = 26.244	= 3 + 3 + 3 + 3 + 3 + 3 + 2 + 2 + 3 + 3 + 3 = 28
Peça (3)	= 3 x 2 x 2 x 3 x 3 x 4 x 4 x 6 x 2 x 3 = 62.208	= 3 + 2 + 2 + 3 + 3 + 4 + 4 + 6 + 2 + 3 = 32
Peça (4)	= 3 x 3 x 3 x 3 x 2 x 3 x 3 x 3 x 4 x 2 = 34.992	= 3 + 3 + 3 + 3 + 2 + 3 + 3 + 3 + 4 + 2 = 29
Peça (5)	= 3 x 2 x 2 x 3 x 3 x 3 x 3 x 3 x 2 x 2 = 11.664	= 3 + 2 + 2 + 3 + 3 + 3 + 3 + 3 + 2 + 2 = 26
Peça (6)	= 4 x 4 x 2 x 3 x 4 x 2 x 2 x 3 x 3 x 3 = 414.720	= 4 + 4 + 2 + 3 + 4 + 2 + 2 + 3 + 3 + 3 = 30
Peça (7)	= 3 x 2 x 2 x 3 x 3 x 3 x 2 x 2 x 3 x 3 = 11.664	= 3 + 2 + 2 + 3 + 3 + 3 + 2 + 2 + 3 + 3 = 26
Peça (8)	= 2 x 2 x 2 x 3 x 3 x 3 x 3 x 2 x 2 x 2 = 5.184	= 2 + 2 + 2 + 3 + 3 + 3 + 3 + 2 + 2 + 2 = 24
Peça (9)	= 3 x 2 x 2 x 3 x 2 x 2 x 2 x 2 x 3 x 3 = 5.184	= 3 + 2 + 2 + 3 + 2 + 2 + 2 + 2 + 3 + 3 = 24
Peça (10)	= 2 x 2 x 3 x 4 x 2 x 2 x 3 x 3 x 3 x 3 = 15.552	= 2 + 2 + 3 + 4 + 2 + 2 + 3 + 3 + 3 + 3 = 27
Peça (11)	= 2 x 2 x 2 x 2 x 3 x 3 x 2 x 3 x 3 x 2 = 5.184	= 2 + 2 + 2 + 2 + 3 + 3 + 2 + 3 + 3 + 2 = 24
Peça (12)	= 2 x 2 x 2 x 2 x 2 x 2 x 2 x 2 x 2 x 2 = 1.024	= 2 + 2 + 2 + 2 + 2 + 2 + 2 + 2 + 2 + 2 = 20
Peça (13)	= 3 x 3 x 3 x 3 x 2 x 2 x 3 x 2 x 2 x 2 = 7.776	= 3 + 3 + 3 + 3 + 2 + 2 + 3 + 2 + 2 + 2 = 25
Totais	= 648.052 Rotas Logicamente Possíveis	= 345 programas para o total de etapas

Apesar do número de programas para as etapas, num total de 345, parecer ser muito alto, é necessário lembrar que para rotas únicas, esse número seria de 130 programas, representando um acréscimo de 215 programas, porém, com o número de rotas de processo possíveis subindo de 13 para 648.052 rotas. Além do mais, esses programas para as etapas, são muito semelhantes por serem executados em máquinas também

muito semelhantes, ou seja, existe um reaproveitamento muito grande de código para as etapas.

A medida em que as etapas forem sendo executadas, haverá uma redução drástica no número de rotas possíveis remanescentes, o que tornará o algoritmo de despacho cada vez mais leve.

2.8 COMENTÁRIOS FINAIS DO CAPÍTULO

Uma vez definidas as células virtuais parte-se para o escalonamento do sistema, onde informações referentes aos tempos, seqüenciamento e sincronismo entre processos devem ser levadas em consideração para uma melhor distribuição das tarefas dentro das células, de forma a garantir uma faixa de desempenho para o fluxo de trabalho e para a capacidade de produção, minimizando aspectos como atrasos e gargalos de produção. Os métodos e técnicas empregadas, bem como as descrições dos algoritmos serão mostradas no capítulo 3.

CAPÍTULO 3

TÉCNICA

DINÂMICA DE

ESCALONAMENTO

E DESPACHO DE

SISTEMAS DO TIPO

NDJSP

O presente capítulo tem por objetivo enumerar os requisitos necessários para a aplicação de uma nova técnica de escalonamento e despacho dinâmica a ser usada em sistemas do tipo NDDJSS, além de definir o problema a ser escalonado identificando-o através da notação definida por (BRUCKER, 1998). A técnica de escalonamento em tempo real desta tese buscará a solução através de modelos matemáticos (algoritmos) e de heurísticas. Finalmente, será definido o procedimento para a geração de regras de despacho para a produção.

3.1 ESCALONAMENTO DE SISTEMAS

Avanços tecnológicos referentes ao *hardware* e ao *software* usados em sistemas computacionais têm levado ao aumento do interesse pelo uso em larga escala de sistemas paralelos e distribuídos no que se refere às aplicações em tempo real, banco de dados, defesa, sistemas de manufatura e etc. (PINEDO; CHAO, 1999). Uma das principais questões relativas a esses tipos de sistema é o desenvolvimento de técnicas efetivas para a distribuição dos processos e dos programas paralelos em múltiplos processadores. Segundo Pinto (2004), o problema se refere a como distribuir (escalonar ou escalar) os processos entre os elementos processadores para atingir o

objetivo de performance do sistema, por exemplo: a minimização do tempo de execução, a minimização dos atrasos de comunicação, a maximização da utilização dos recursos, bem como atingir os objetivos de controle de uma planta industrial no que tange os aspectos de indeterminismo quanto ao seqüenciamento das atividades, à designação dos recursos e aos tempos relacionados aos processos. Do ponto de vista de um sistema *job shop*, a escolha dessa distribuição se torna um problema de administração de recursos e pode ser considerado um fator importante durante a fase de projeto dos sistemas de controle em ambientes com múltiplos processadores.

Métodos de escalonamento de processos são tipicamente classificados dentro de categorias como mostrado na taxonomia proposta por Casavant; Kuhl (1988) como mostrado na Fig.3.1. Esta classificação, apesar de antiga, é ainda bastante usada por pesquisadores que trabalham com escalonamento de sistemas. Outras áreas de conhecimento como: planejamento e controle podem usar classificações diferentes. A classificação do método de escalonamento a ser usado encontra-se sombreada.

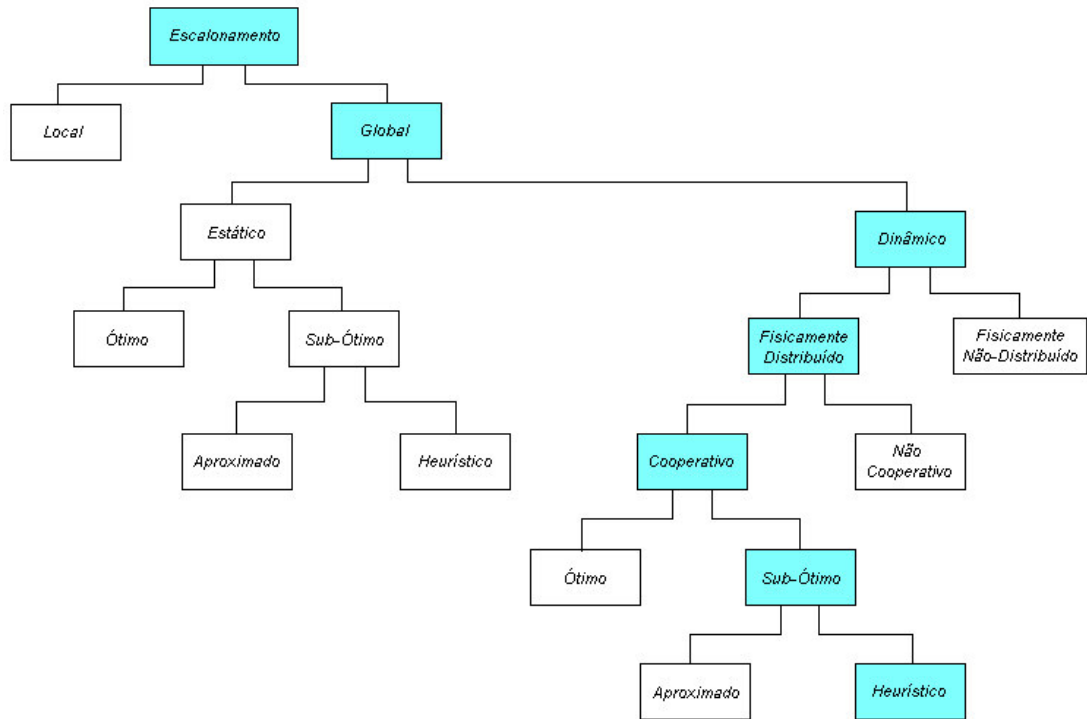


Figura 3.1 - Classificação dos Métodos de Escalonamento (CASAVANT; KUHL, 1988).

Nesta taxonomia, o escalonamento local consiste em uma designação de processos para as fatias de tempo de um sistema constituído por um único processador e não faz parte do escopo deste trabalho. O escalonamento global, por outro lado, é o processo de decisão de onde executar um processo em um sistema com múltiplos processadores, podendo ser conduzido por uma autoridade simples e centralizada, ou estar distribuído entre os elementos processadores.

Os métodos de escalonamento global podem ser divididos em duas classes, métodos estáticos ou métodos dinâmicos. Os métodos estáticos são caracterizados principalmente pela obtenção do escalonamento antes do início da execução das tarefas, e conseqüentemente, impossível de adaptar-se às mudanças nas hipóteses em tempo de execução. Por essa razão, os métodos dinâmicos são mais adequados a aplicação em sistemas de manufatura flexíveis, como o tipo de sistema de interesse nesta tese.

3.1.1 Escalonamento Dinâmico

Rabelo; Klen (2000) definem escalonamento *dinâmico* como sendo "a ação de adaptar o escalonamento corrente, simultaneamente com a sua execução, em função da ocorrência de eventos não planejados, tanto os provenientes do chão-de-fábrica como os do nível de planejamento, de tal forma que o escalonamento se mantenha... *realístico*: reflita o real estado do chão-de-fábrica; *viável*: realizável, de acordo com as restrições temporais, de capacidade e tecnológicas existentes; e *coerente*: sem desvios em relação aos objetivos e metas traçadas".

O escalonamento dinâmico baseia-se na distribuição dos processos entre os processadores durante o tempo de execução. Essa distribuição é realizada através da transferência de tarefas de processadores sobrecarregados para processadores menos sobrecarregados ou livres. Este processo é conhecido como balanceamento de carga e tem o objetivo de melhorar a performance da aplicação (SHIVARATRI; KREUGER; SINGHAL, 1992). Um algoritmo de balanceamento de carga típico pode ser definido através de três características particulares:

1. característica de informação: a qual especifica a informação de quantidade de carga que se faz necessário para o trabalho aos responsáveis pela tomada de decisão;
2. característica de transferência: determina as condições sob as quais uma tarefa pode ser transferida, isto é, a carga corrente da estação de trabalho e o tamanho da tarefa que está sendo considerada, e ainda, pode ou não incluir a migração de tarefas, ou seja, suspender uma execução de tarefa e a transferir para outra estação de trabalho para que a execute (preempção);
3. característica de alocação: identifica os elementos processadores para os quais uma tarefa possa ser transferida.

As operações de balanceamento de carga podem ser centralizadas em um único processador ou distribuídas entre todos os elementos processadores que participam do processo de balanceamento de carga. Podem também existir algumas políticas de escalonamento combinadas, como por exemplo, o de uma política de informação que pode ser centralizada, porém as políticas de transferência e alocação das tarefas podem ser distribuídas. Nesse caso, todos os elementos processadores enviam suas respectivas informações de carga para um processador central e recebem informação de carga do sistema através desse processador central. No entanto, as decisões relativas a quando e para onde uma tarefa poderia ser transferida são tomadas localmente por cada processador. Se uma política de informação distribuída for empregada, cada elemento processador mantém sua própria imagem local da carga do sistema. Essa política cooperativa é freqüentemente alcançada através da informação de um gradiente de distribuição de carga entre os elementos processadores. Cada processador repassa sua informação de carga corrente para seus pares em intervalos de tempo, resultando na dispersão da informação de carga entre todos os elementos processadores em um curto período de tempo. Políticas de informação distribuída podem também ser não cooperativas. Técnicas de escalonamento aleatório são um exemplo de escalonamento não cooperativo, em que um processador sobrecarregado escolhe aleatoriamente outro processador para o qual irá transferir uma parte da carga de trabalho. O balanceamento aleatório de carga funciona razoavelmente bem quando as cargas de todos os processadores são

relativamente altas, isto é, quando não faz mais diferença onde a tarefa será executada.

A vantagem do balanceamento dinâmico de cargas sobre os métodos estáticos de escalonamento é que o sistema não necessita estar atento em relação ao comportamento do tempo de execução das aplicações antes da execução. A flexibilidade inerente ao balanceamento dinâmico de carga possibilita a adaptação quanto à característica de imprevisibilidade dos requisitos da aplicação em tempo de execução. O balanceamento dinâmico de carga é particularmente proveitoso em sistemas que consistem de uma rede de trabalho, composta de estações de trabalho interligadas, na qual a meta primária de desempenho é maximizar a utilização dos processadores em vez de minimizar os tempos de execução das aplicações. A maior desvantagem dos métodos de balanceamento de carga dinâmicos é o custo relativo ao tempo de execução, devido principalmente aos seguintes fatores:

1. transferência de informação de carga entre os processadores;
2. processos de tomada de decisão quanto à seleção dos processos e dos processadores para a transferência de tarefas;
3. atrasos de comunicação com relação às tarefas de auto-alocação.

Apresentada a classe (escalonamento dinâmico) que caracteriza a nova técnica de escalonamento e, antes de apresentar a solução buscada neste trabalho, explanam-se na próxima seção particularidades do escalonamento em sistemas de manufatura e também a notação a ser usada para a descrição formal do problema de escalonamento.

3.2 TÉCNICAS DE ESCALONAMENTO APLICADAS A SISTEMAS DE MANUFATURA

No atual ambiente competitivo em que as indústrias manufatureiras encontram-se inseridas, o escalonamento eficaz tornou-se uma necessidade para sobrevivência no mercado. Indústrias esforçam-se ao máximo para cumprir prazos firmados com os clientes, onde o fracasso deste comprometimento pode resultar em perda da imagem da empresa perante os clientes, dentre outras coisas (PINEDO; CHAO, 1999).

O escalonamento das atividades é uma das etapas que compõem o planejamento da produção, podendo ainda estar inserida no contexto de sistemas de controle, num nível hierárquico superior. A segunda opção será a adotada nesta tese. No escalonamento, é levada em consideração uma série de “*jobs*” que disputam vários recursos por um período de tempo, recursos esses que possuem capacidade limitada. Os elementos a serem processados são chamados de ordens de fabricação, tarefas ou *jobs* e são compostos de partes elementares chamadas atividades ou operações. Os principais objetivos tratados no problema de escalonamento podem ser resumidos no atendimento de prazos (ou datas de entrega), na minimização do tempo de fluxo dos estoques intermediários e na maximização da utilização da capacidade disponível, ou mesmo na combinação destes objetivos (WALTER, 1999).

3.2.1 Escalonamento - Definição Formal e Notação de Brucker

Antes de definir formalmente escalonamento, é importante definir algumas notações relativas aos recursos e às tarefas.

Considerar-se-á os recursos, representados por um conjunto de m máquinas $M = \{m_1, \dots, m_m\}$ e por um conjunto de n tarefas $T = \{t_1, \dots, t_n\}$. É necessário lembrar que o conceito de recurso usado nesta pesquisa refere-se a recursos materiais, humanos e de equipamentos. No entanto, neste trabalho, abordar-se-á apenas o tipo de recurso (máquina), partindo-se do pressuposto que as necessidades de recursos materiais e humanos serão sempre satisfeitas.

Algumas características importantes estão associadas às tarefas, dentre as quais podemos citar:

1. Tempo de Execução (p_{ij}): tempo necessário para a execução completa da tarefa t_j na máquina m_i . Caso as máquinas sejam idênticas este tempo é denotado apenas por p_j . No caso de máquinas especializadas, isto é, quando a tarefa t_j só pode ser executada em um sub-conjunto M_j de M . O tempo p_{ij} só é definido para $i \in M_j$;
2. Data de Disponibilidade (r_j): corresponde ao instante em que a tarefa t_j está disponível, isto é, é o primeiro instante em que a tarefa t_j pode começar a ser executada;
3. Data Devida (d_j): instante de tempo no qual a tarefa t_j deve estar pronta. Conforme o problema, o término da execução da tarefa t_j após d_j é permitido, mas existe uma penalidade associada junto ao critério de otimização.

Definição 1: Um escalonamento é uma atribuição de tarefas a recursos, no tempo, que obedece as seguintes propriedades:

1. cada tarefa é atribuída a uma máquina por vez, isto é, uma tarefa não pode ser executada por mais do que uma máquina simultaneamente;
2. cada máquina executa no máximo uma tarefa por vez;
3. a tarefa t_j não é executada antes da sua data de disponibilidade (r_j);
4. todas as tarefas são executadas por completo.

A definição que se apresenta é bem geral e é válida para a maioria dos problemas clássicos de escalonamento. No entanto, conforme as alterações nas condições do problema, propriedades adicionais poderão ser atribuídas ao problema de escalonamento.

3.2.1.1 Notação

Será adotada a nomenclatura básica introduzida por Brucker (1998), a qual tem o objetivo de criar uma classificação para os diversos tipos de problemas de escalonamento. A notação é composta por três campos ordenados $\alpha | \beta | \gamma$, onde o primeiro campo descreve os recursos, o segundo as tarefas e o último o critério de otimização.

3.2.1.1.1 Campo α - Recursos

Os tipos mais comuns de recursos em sistemas de manufatura são as máquinas paralelas e as máquinas tipo “*shop*” (especializadas). No caso das máquinas paralelas as tarefas podem ser completadas por uma única máquina. Para o caso de máquinas do tipo *shop*, existem tarefas que podem precisar ser executadas em máquinas distintas. As máquinas paralelas não constituem o escopo deste trabalho.

3.2.1.1.1.1 Máquinas *Shop* (Especializadas)

No sistema de manufatura a ser abordado nesta pesquisa, as máquinas são consideradas como dedicadas, ao invés de paralelas. A organização dessas máquinas nos sistemas produtivos dá-se segundo os três principais modelos descritos na Tabela 3.1 e suas possíveis variações.

Tabela 3.1 - Variações de Máquinas do Tipo *Shop*.

Tipo	Descrição
<i>Flow-Shop</i> (F_m)	No modelo <i>flow-shop</i> (<i>line-transfer</i>), todas as tarefas têm o mesmo número de operações as quais são executadas na mesma ordem, por todas as m máquinas em série. Após a execução em uma máquina, a tarefa entra na fila para execução na próxima máquina. Geralmente as filas seguem a disciplina FIFO (<i>first in first out</i>);
<i>Open-Shop</i> (O_m)	Neste modelo as tarefas também devem ser executadas por todas as m máquinas, entretanto, alguns tempos de execução podem ser nulos. Não existem restrições quanto à ordem das máquinas onde cada uma das tarefas deve passar;
<i>Job-Shop</i> (J_m)	Neste modelo cada uma das tarefas possui um roteiro pré-determinado. O número de máquinas, como nos modelos anteriores também é m . Um tipo especial de <i>job-shop</i> onde uma tarefa pode visitar uma máquina mais do que uma vez, é denominado por recirculação (com a palavra <i>recrc</i> no segundo campo).

3.2.1.1.2 Campo β - Tarefas

Além das características das tarefas mostradas anteriormente como tamanho (tempo de execução), data de disponibilidade e data de término, serão introduzidas aqui outras características bastante comuns, conforme a Tabela 3.2.

Tabela 3.2 - Outras características do campo de tarefas.

Tempo de preparação (<i>setup</i>) da máquina (s_{jk})	Dadas duas tarefas t_j e t_k , s_{jk} corresponde ao tempo de preparação de uma máquina que executou uma tarefa t_j e vai executar uma tarefa t_k . Caso este tempo também dependa da máquina m_i em questão, usa-se a notação s_{ijk} . No caso de t_j ser a primeira (ou última) tarefa executada na máquina, o tempo de preparação é denotado por s_{0j} e s_{j0} , respectivamente.
Relações de precedência (<i>prec</i>)	Quando existir, a ordem para a execução das tarefas geralmente é representada por um grafo orientado, onde os vértices correspondem às tarefas e os arcos às relações diretas de precedência. Um arco de uma tarefa t_j a uma tarefa t_k faz com que a tarefa t_k possa iniciar sua execução apenas após o término da tarefa t_j . Caso o grafo de precedência pertença a uma família específica, ao invés de <i>prec</i> , coloca-se o nome da família no segundo campo. Os mais comuns são: <i>intree</i> , <i>outree</i> , <i>chains</i> , <i>split-compute-merge</i> , dentre outros.
Tempo de comunicação entre máquinas (c_{jk})	No caso de precedência entre duas tarefas t_j e t_k , se a tarefa t_k finaliza a sua execução no instante t na máquina m_i , a tarefa t_k não pode iniciar a sua execução antes do instante t , na mesma máquina m_i , ou do instante $t + c_{jk}$ em qualquer outra máquina.
Interrupção ou preempção (<i>prmp</i>)	Quando é possível interromper uma tarefa, ela não precisa necessariamente terminar na mesma máquina em que começou a sua execução. O tempo de execução utilizado até a interrupção não é perdido. Quando a tarefa volta a uma máquina qualquer, ela só precisa ser executada durante o tempo restante após a última interrupção.
Restrições de escolha de máquina (M_j)	Este item aparece no segundo campo quando a tarefa t_j só puder ser executada em um sub-conjunto $M_j \subset P$.

No caso de problemas de escalonamento que não se adaptem as variações apresentadas na tabela, pode-se definir novos parâmetros para os campos, desde que a notação de Brucker para os campos recurso, tarefa e critério de otimização sejam respeitados (BRUCKER, 1998).

3.2.1.1.3 Campo γ - Critério de Otimização

Para a definição dos principais critérios de otimização usados em problemas de escalonamento, torna-se necessário denotar o instante em que cada tarefa t_j termina a sua execução, sendo este instante normalmente chamado de C_j . Para cada tarefa é associada uma função de custo $f_j(C_j)$. A função de custo total para a maioria dos critérios de otimização corresponde ao máximo,

$$\max\{f_j(C_j) \mid 1 \leq j \leq n\}, \quad (3.1)$$

ou a soma de todos os custos,

$$\sum_{j=1}^n f_j(C_j) \quad (3.2)$$

O problema de escalonamento consiste em encontrar uma solução que minimize a função de custo total. As funções de custo mais comuns são definidas na Tabela 3.3.

Tabela 3.3 - Funções de Custo mais Comuns.

Tipo	Descrição
<i>makespan</i> (C_{max})	Definido como o instante de tempo em que a última tarefa é finalizada, $C_{max} = \max\{C_1, \dots, C_n\}$. Neste caso o critério corresponde ao máximo e as funções $f_j(\cdot)$ são funções identidade.
Tempo total de término ($\sum C_j$)	Definido como a soma dos instantes de término de cada tarefa, $\sum C_j = \sum_{j=1}^n C_j$
Tempo total de término ponderado ($\sum \omega_j C_j$)	Definido como a soma dos instantes de término de cada tarefa com peso ω_j , $\sum \omega_j C_j = \sum_{j=1}^n \omega_j C_j$
<i>Lateness</i> (grau de atraso)	$L_j = C_j - d_j$, pode assumir valores negativos
<i>Earliness</i> (grau de adiantamento)	$E_j = \max\{0, d_j - C_j\}$
<i>Tardiness</i> (grau de atraso efetivo)	$T_j = \max\{0, C_j - d_j\}$
Penalidade unitária por atraso	$U_j = \begin{cases} 0 & \text{se } C_j \leq d_j \\ 1 & \text{se } C_j > d_j \end{cases}$

Da mesma forma que o *makespan*, definido por $\max\{C_j\}$, é denotado por C_{max} , os critérios de maximização também são denotados por L_{max} , E_{max} , T_{max} e U_{max} .

Dessa maneira, para o problema de escalonamento de tarefas para as células de manufatura virtuais independentes adotadas neste trabalho, com tempo de execução p_{ij} , relações de precedência arbitradas por grafos de precedência em m máquinas especializadas do tipo *job shop*, de forma que o atraso seja minimizado, se usará da

notação dos três campos $\alpha | \beta | \gamma$, em que a representação para o problema de escalonamento para a produção proposta para essa pesquisa é dada a seguir, na eq.(3.3), onde o critério de otimização segue uma heurística simples⁴¹ resultante da combinação de duas outras regras heurísticas também simples e bem conhecidas dos pesquisadores da área:

$$Jm | prec;recrc;P_{ij};C_{ij};t_i;d_i;S_{ijk};M_i \subset P | Regra Heurística \tag{3.3}$$

Portanto, devido ao indeterminismo quanto aos instantes de chegada das tarefas, usar-se-á uma regra heurística combinada para o terceiro campo do problema descrito pela notação da eq.(3.3), e o sistema se baseará em regras de despacho.

3.2.1.2 Representações gráficas

Uma instância do problema corresponde a um grafo de precedência e uma quantidade de máquinas. A Fig.3.2 mostra um exemplo de grafo de precedência.

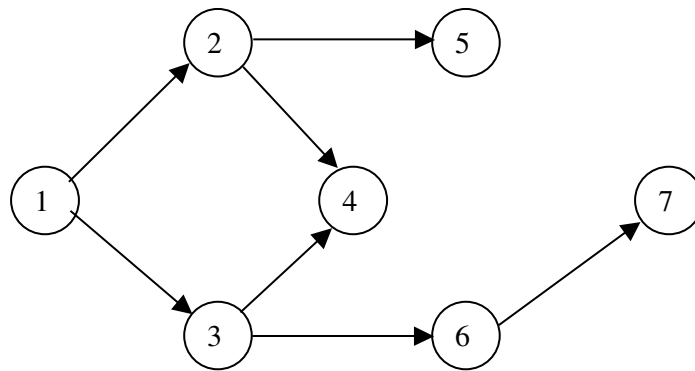


Figura 3.2 – Um exemplo de um possível grafo de precedência.

A representação gráfica de um escalonamento possível pode ser dada pelo seguinte diagrama de espaço-tempo (diagrama de Gannt) da Fig.3.3. O tempo é representado no eixo horizontal e as diferentes máquinas são representadas no eixo vertical.

⁴¹ O conceito de regra heurística simples faz menção a heurísticas com um único objetivo. Regras combinadas levam a múltiplos objetivos.

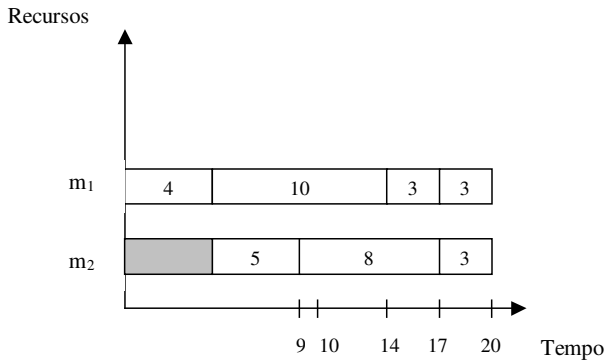


Figura 3.3 - Diagrama de Gantt.

Uma alternativa para a representação espaço-tempo é a representação tarefa-tempo mostrada na Fig.3.4. Existem problemas de escalonamento onde este segundo tipo de diagrama pode ser mais útil do que o primeiro.

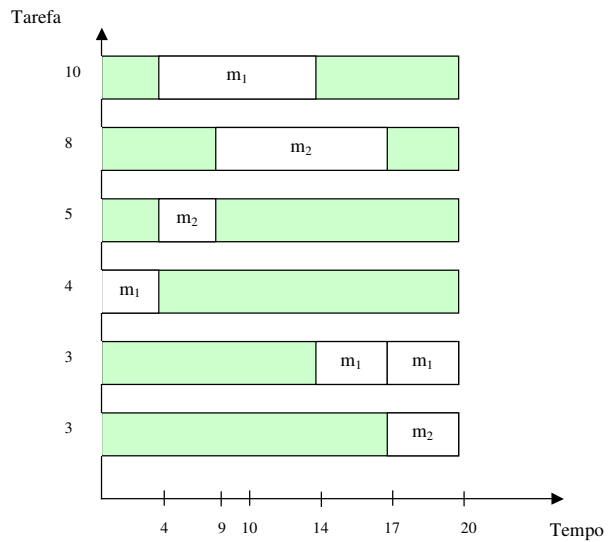


Figura 3.4 - Diagrama de tarefa-tempo.

Os retângulos em negrito correspondem aos instantes em que as respectivas tarefas não estão alocadas em nenhuma das máquinas disponíveis.

3.2.2 Escalonamento aplicado a *job-shop*

O problema de escalonamento aplicado a um *job-shop* é um problema de otimização combinatória, na maioria das vezes NP-completo⁴², ou seja, exige a enumeração explícita ou implícita de todas as alternativas possíveis para garantir a solução ótima. Dessa forma, algoritmos de otimização são computacionalmente viáveis quando aplicados a problemas pequenos, com objetivos limitados, onde a solução ótima pode ser encontrada sem muito esforço de computação. Para problemas de porte similar aos encontrados no ambiente industrial, costuma-se sacrificar a obtenção de uma solução ótima por métodos heurísticos, que resultem em uma solução subótima, com tempo computacional aceitável. Dessa maneira, os modelos podem ser agrupados em duas classes dependendo dos métodos de escalonamento aplicados: os modelos matemáticos (solução ótima) e os modelos heurísticos, descartando-se os modelos aproximados. Porém, segundo Pacheco; Santoro (1999), a escolha do modelo mais adequado à solução de problemas reais deve levar em consideração alguns aspectos como:

1. o número de máquinas envolvidas;
2. os roteiros das ordens;
3. o regime de chegada das ordens;
4. a variabilidade dos tempos de processamento, entre outras características.

Modelos Matemáticos

Os modelos matemáticos empregam abordagens algorítmicas ou de otimização para realizar os escalonamentos. Seus métodos enfatizam a obtenção de resultados ótimos em função de algum parâmetro de desempenho, como, por exemplo, a minimização dos tempos de produção ou a maximização do uso dos recursos. Devido ao grande número de soluções possíveis e à complexidade do problema de escalonamento, torna-se difícil e até impossível modelar todas as variáveis envolvidas no processo. Além disso, o tempo de resposta destes sistemas eleva-se consideravelmente,

⁴² NP – Não polinomial.

tornando impraticável a obtenção de soluções ótimas em tempos satisfatórios. Esta abordagem tende a ser aplicada em problemas onde simplificações podem ser introduzidas. Como exemplo pode-se citar os métodos de *Johnson e Branch and Bound*.

Modelos Heurísticos

Os modelos heurísticos, que consistem em regras práticas (indutivas) de escalonamento, permitem resolver problemas grandes em tempos de computação viáveis. Tais métodos são ditos satisfatórios, pois não garantem que a solução encontrada seja a ótima, mas sim alguma solução que satisfaça a um critério de desempenho. De acordo com Chan (2003), nos modelos heurísticos quando há mais de uma atividade disputando um recurso, cabe a uma regra selecionar a atividade a atender primeiro, de forma a otimizar o programa de produção de acordo com algum critério. Usando conhecimentos ou experiências heurísticas previamente obtidas de especialistas humanos é possível restringir o número de soluções alternativas até que seja encontrada uma única solução viável para o momento.

3.2.3 Técnicas de Escalonamento

Nesta pesquisa, a programação da execução será orientada a tarefas, onde as tarefas deverão ser realizadas para alcançar um objetivo fixo, ao contrário de programações orientadas a processos, onde um processo deve ser controlado sob certas condições fixas que devem ser mantidas.

Não serão usadas técnicas do tipo “*temporal planning*”, porque esse tipo de programação não atende plenamente à classe sistemas a eventos discretos (SED), pois necessita de representação explícita de tempos para as tarefas e os atrasos relacionados à execução das tarefas. Isso corresponde a um tipo de indeterminismo em SED que deve ser tratado por técnicas mais adequadas a esse problema.

Os sistemas dinâmicos a eventos discretos possuem uma característica de incerteza que deve ser levada em conta para o escalonamento das tarefas, sendo elas:

1. as durações das tarefas não podem ser conhecidas com exatidão, o que constitui um indeterminismo quanto ao tempo de execução das tarefas. Uma saída para esse tipo de incerteza seria usar uma distribuição probabilística, o que não resolveria o problema de indeterminismo quanto ao tempo do presente caso, pois esses tempos continuariam não exatos;
2. os recursos podem possuir baixa capacidade temporária como, por exemplo, no caso de: quebra de máquinas e a matéria-prima não chegar a tempo;
3. a necessidade de se levar em conta novas tarefas como a chegada de novas ordens de manufatura e a necessidade de refazer uma tarefa que falhou.

Definição 2: O tempo de *setup* é definido como o tempo necessário para configurar um recurso, para que esse seja capaz de processar uma atividade. A medida do tempo de *setup* pode depender das atividades que precedem e das atividades posteriores à configuração.

Um exemplo interessante é o *mix* de tintas usado numa estação de pintura. Se uma troca de cores, de preto para branco for necessária, é preciso limpar a máquina de pintura completamente antes de usá-la com a tinta branca. Porém, se a necessidade fosse a troca de branco para cinza, isso necessitaria um esforço menor e, portanto, menos tempo para a preparação. Com o crescente número de sofisticadas máquinas parametrizáveis, o escalonamento de tarefas visando a minimização dos tempos de *setup* tornou-se um critério de otimização comum.

A representação de incertezas caracteriza-se da seguinte forma:

1. as variáveis podem assumir valores distintos;
2. apresentarem tipos básicos, como por exemplo, uma lista com valores possíveis;
3. serem do tipo probabilística como, por exemplo, redes bayesianas;
4. do tipo possibilidades, como os conjuntos *fuzzy*;

5. usar certos modelos de resolução baseados em decisões possíveis, como por exemplo, a teoria de decisão.

Para as conceituações seguintes, torna-se necessário estabelecer um conjunto de definições.

Definição 3: Um sistema é dito autônomo quando nenhuma intervenção humana é necessária para a sua execução.

Definição 4: Uma escala é dita estável quando as decisões tomadas de maneira *off-line* não são revisadas durante a sua execução.

Definição 5: Uma escala é dita robusta quando características como otimalidade e qualidade não se degradam durante a sua execução.

Definição 6: Uma escala é dita adaptativa quando qualquer comportamento reativo for tratável durante sua execução (*on-line*).

Definição 7: Uma escala é dita flexível quando não apresenta nenhum conjunto completo para a tomada de decisão. Decisões *off-line* incompletas ou não consideradas devem ser tomadas no modo *on-line*⁴³.

Definição 8: Uma escala contingente/condicional é aquela na qual algumas alternativas são modeladas, porém somente uma é executada (disjunção).

Definição 9: Uma técnica é dita monotônica quando decisões de escalonamento nunca são questionadas posteriormente.

Definição 10: Uma técnica é dita não-monotônica quando as decisões de escalonamento podem ser alteradas posteriormente.

Definição 11: Um par (evento, decisão) é dito síncrono quando podem ser alocados num estágio preciso da escala.

Definição 12: Um par (evento, decisão) é dito assíncrono quando puder ocorrer em qualquer momento.

⁴³ On-line no contexto em questão refere-se a em tempo de execução (tempo real).

Portanto, a técnica de escalonamento necessária a esta tese é dita autônoma, estável, adaptativa, flexível, contingente e monotônica.

As técnicas mais usadas em problemas de escalonamento podem ser classificadas como Reativas, Pró-ativas, Progressivas e Mistas.

3.2.3.1 Técnicas Reativas

As técnicas reativas caracterizam-se pela execução da escala nominal enquanto nenhum evento inesperado acontece. De outra forma, a escala nominal deve ser totalmente revisada. Pode possuir característica não-monotônica, porém limitada, onde um re-escalonamento deve ser realizado para ajuste da escala nominal. Nos casos extremos, toda a escala preditiva deve ser feita novamente. O custo de resolução é alto e geralmente do tipo subótimo, onde situações reativas são limitadas em razão do tempo (VIDAL; BECK, 2003).

3.2.3.2 Técnicas Pró-ativas

Existem duas abordagens distintas para as técnicas do tipo pró-ativa. A primeira que usa uma representação probabilística/*fuzzy* da incerteza para gerar uma escala que cubra muitos casos. A segunda baseia-se na construção de escalas que levem em consideração casos de desvio em relação a uma escala nominal. A representação é do tipo flexível ou condicional. As características principais da segunda opção são a resolução *off-line* aliada à tomada de decisão *on-line*, onde todos os casos devem estar previstos. Uma desvantagem para essa técnica reside no fato de que o modelo resultante pode se tornar extenso demais (VIDAL; BECK, 2003).

3.2.3.3 Técnicas Progressivas

Essas técnicas baseiam-se na intercalação do escalonamento com a execução, para intervalos de tempo pequenos (curto prazo) durante o tempo de execução das escalas. Possui resolução *on-line*, porém como uma tarefa de apoio, podendo dispor de mais tempo para a procura da solução. Realiza as decisões de escalonamento quando informações novas sobre a execução dos planos chegam. Caracteriza-se como monotônica e (possivelmente) ótima, porém somente para uma visão de curto prazo.

Apresenta a necessidade de ser rápida, porém a característica de antecipação deixa tempo para a resolução. Também pode levar em conta eventos síncronos e ou assíncronos e requer pouca memória, possivelmente uma memória constante (VIDAL; BECK, 2003).

3.2.3.4 Técnicas Mistas

Essas técnicas combinam técnicas de resolução pró-ativa para levar em conta muitas possibilidades (muitos casos), mais a capacidade de lidar com a tomada de decisão *on-line* e eventos inesperados. Vidal; Beck (2003) atestam que não existem muitos exemplos na literatura para abordagens mistas, as quais representam atualmente um grande desafio.

3.2.3.5 Características mínimas para uma Técnica Mista aplicada ao Problema tratado nesta Pesquisa

A técnica de escalonamento e despacho a ser desenvolvida para esta pesquisa, classifica-se como uma técnica de escalonamento e despacho mista, a qual mescla características de técnicas pró-ativas (*off-line*) e de técnicas progressivas (*on-line*).

A idéia principal das técnicas progressivas (*on-line*) é intercalar o escalonamento e a execução, sendo no caso dessa pesquisa considerada a intercalação do escalonamento e do controle da execução das tarefas. A geração dos planos é monotônica com uma visão à frente limitada, porém, incrementalmente atualizada. Uma característica importante desse tipo de técnica refere-se ao fato de poder incluir incertezas quanto ao tempo de duração das atividades. A alocação dos recursos para as tarefas é realizada no curto-prazo quando novos tempos efetivos são tomados durante a execução.

O escalonamento pode ser realizado através da técnica chamada de “*telescoping time-horizon*” (VIDAL; BECK,2003). Essa técnica representa a geração de um plano global, porém esse plano só é detalhado durante a execução, para janelas de tempo pequenas, mantendo um plano abstrato ou incompleto para janelas mais longas de tempo. O plano é futuramente detalhado durante o progresso da execução.

A técnica de escalonamento *master* da produção proposta nesta tese, mescla características de modelos matemáticos (algorítmicos) e modelos heurísticos.

3.3 PROPOSTA DE TÉCNICA DE ESCALONAMENTO PARA SISTEMAS DO TIPO NDDJSS

Como mencionado anteriormente, o principal objetivo de um escalonador de um *job shop* consiste em escalonar (ou escalar) seus conjuntos de tarefas de maneira que esses conjuntos sejam executados na menor quantidade de tempo possível, segundo algum critério de otimização. Geralmente, restrições relacionadas a tarefas individuais não são levadas em consideração, e o conjunto de tarefas, como um todo, é o único interesse.

Para a maior parte dos problemas de escalonamento solucionáveis em tempo polinomial, os algoritmos usados são do tipo guloso. Mais especificamente, algoritmos de lista, onde as tarefas são ordenadas segundo algum critério, para em seguida serem alocadas às máquinas. O caso específico de uso de rotas de processo alternativas requer a adoção de técnica de programação dinâmica, onde a solução é construída por etapas, a partir de soluções para subproblemas.

Com relação aos problemas considerados difíceis, têm-se algumas opções, tais como usar algoritmos simples como os de lista, usar heurísticas mais sofisticadas ou enumerar as soluções possíveis de uma forma mais conveniente. A vantagem das duas primeiras opções refere-se ao fato de serem mais rápidas, apesar de nem sempre encontrarem a solução ótima, sendo nesse caso, por um abuso de notação, aceitos os escalonamentos “válidos” como soluções para o problema. A última opção mesmo que bem feita, nem sempre é praticável devido ao fato de possíveis explosões combinatórias de estados.

Um dos métodos mais usado é o método de enumeração *Branch-and-Bound* (BB). É um dos principais métodos de enumeração encontrado na literatura. Este método é semelhante à divisão e conquista, onde segundo um critério dado, algumas das conquistas não precisam ser executadas, sem prejuízo ao resultado final.

O método BB pode ser descrito da seguinte forma. Inicialmente, o espaço de soluções S é dividido em subconjuntos disjuntos S_1, \dots, S_k . Estes são subdivididos em subconjuntos até o ponto em que os subconjuntos sejam unitários. O espaço inicial de soluções S pode ser visto como a raiz de uma árvore, onde os filhos de um nó correspondem a subdivisões de conjunto correspondentes ao nó. A idéia básica do BB é eliminar os subconjuntos que não tem a possibilidade de conter soluções menores o quanto antes.

Um nó da árvore pode deixar de ser analisado nas seguintes condições:

1. o subconjunto S_i não contém nenhuma solução melhor do que a melhor solução conhecida até o momento;
2. o subconjunto S_i não contém soluções válidas;
3. todas as soluções contidas em S_i não são melhores que as soluções contidas em outro subconjunto S_j . Isto é, a melhor solução em S_i é no máximo tão boa quanto a melhor solução em S_j .

O mecanismo básico usado em BB consiste em conhecer alguma estimativa, ou limite na função objetivo das soluções em S_i , de forma a compará-lo com estimativas de outros conjuntos, ou de alguma solução. A solução inicial pode ser obtida por um método heurístico qualquer. Logo, um método de BB consiste de:

1. *branching*: escolha do método de divisão e busca nos subconjuntos;
2. *bounding*: alguma forma de estimar a função-objetivo das soluções de um subconjunto, de maneira a eliminar os nós na árvore de busca.

Limites para o *bound* podem ser obtidos com heurísticas como, por exemplo, a EDD (*earliest due date*) onde as tarefas são ordenadas por ordem crescente de datas devidas, ou qualquer outro tipo de heurística adequada ao problema tratado.

Nesta tese, um algoritmo de BB é proposto para a condução da designação dos recursos (máquinas) às tarefas. Porém, ele trabalhará com uma visão à frente de curto prazo⁴⁴.

3.3.1 Métodos Heurísticos para Geração de Regras de Despacho

Devido à dificuldade de determinação de uma solução ótima para o NDJSP com o objetivo de minimização do atraso total das tarefas, é certo que se desenvolvam métodos aproximados ou heurísticos que produzam soluções de boa qualidade a um esforço computacional aceitável.

Segundo Oliveira (2002), os métodos heurísticos podem ser divididos em duas classes, a saber:

- heurísticas de passo único;
- heurísticas de busca local.

3.3.1.1 Heurística de passo único

Uma heurística de passo único é aquela onde se constrói uma solução completa, fixando uma operação por vez na programação. Estes passos são executados baseados em regras de despacho. Esta classe de heurísticas é geralmente a mais usada para resolver problemas práticos em ambiente *job shop*. Isto se deve a sua facilidade de implementação e o pequeno esforço computacional necessário.

A seguir, será mostrada a notação utilizada para as formulações matemáticas e citadas as regras de despacho mais comuns.

t – instante em que se considera a alocação de uma operação em uma máquina;

k – k -ésima operação da tarefa j ;

P_j – tempo requerido pelas operações restantes da tarefa j , a partir do instante t ;

s_j – folga da tarefa j ;

⁴⁴ Visão de curto prazo denota a visão de um passo à frente e não até o final da árvore de decisão.

M' – número total de operações de cada tarefa;

p_{ij} – tempo de processamento da operação i da tarefa j ;

E_j – instante de chegada da tarefa j ao sistema;

D_j – data de entrega da tarefa j ;

d_{kj} – data de entrega da operação k da tarefa j .

De acordo com Oliveira (2002), as regras de despacho mais eficazes, e provavelmente as mais usadas na prática para o JSP com o objetivo de minimizar o atraso total das tarefas (*jobs*) são:

- ERD (*earliest release date*): seleciona a operação associada com a tarefa que chegou mais cedo ao sistema;
- EDD (*earliest due date*): seleciona a operação associada com a tarefa que possui a data de entrega mais cedo;
- MST (*minimum slack time*): seleciona a operação associada com a tarefa que possui a folga mínima. A folga mínima de uma tarefa é definida da seguinte maneira:

$$s_j = D_j - (t + P_j)$$

a expressão mostra que a folga de uma tarefa s_j é a diferença entre sua data de entrega D_j e o tempo de término mais cedo da tarefa $(t + P_j)$;

- ODD (*operation due date*): seleciona a operação que possui a data de entrega da operação mais cedo. A data de entrega de uma operação é definida da seguinte maneira:

$$\text{seja o subintervalo} = (D_j - E_j) / M'; \quad \text{então faça}$$

$$d_{kj} = k \cdot \text{subintervalo} \quad k = 1, \dots, M$$

a expressão mostra que a data de entrega de uma operação é determinada dividindo o intervalo entre a data de entrega da tarefa e seu instante de chegada ao sistema, em M' subintervalos. O final de cada subintervalo representa uma data de entrega para a correspondente operação;

- MDD (*modified due date*): seleciona a operação associada com a tarefa que possui a data de entrega modificada mais cedo. Define-se a data de entrega modificada da tarefa j da seguinte maneira:

$$D_j' = \max \{D_j, t + P_j\}$$

A expressão mostra que se a tarefa j atrasar, a nova data de entrega será seu instante mais cedo possível de término. Caso contrário, a data de entrega permanece a original;

- MOD (*modified operation due date*): é a versão da MDD considerando-se como base às operações e não a tarefa. A data de entrega da k -ésima operação da tarefa j é definida da seguinte maneira:

$$d_{kj}' = \max \{d_{kj}, t + p_{kj}\}$$

a data de entrega da operação k , é calculada através da seguinte fórmula recursiva:

$$d_{kj} = d_{k-1,j} + (D_j - E_j) \cdot p_{kj} / \sum p_{ij}$$

- CR + SPT: combinação de *critical ratio* (CR) com *shortest processing time* (SPT). Nesta regra a data de entrega da k -ésima operação da tarefa j é definida como:

$$d_{kj} = \max \{t + \beta(t) \cdot p_{kj}, t + p_{kj}\} \text{ com } \beta(t) = a_j(t)/P_j \text{ e } a_j(t) = D_j - t$$

O termo $\beta(t)$ representa a *critical ratio* que é a razão entre o tempo para atingir a data de entrega e o tempo de processamento restante da tarefa. Quando $\beta(t)$ for menor que 1, d_{kj} assume o valor do segundo termo da expressão, dado que a tarefa vai atrasar;

- S/RPT + SPT: combinação de *slack per remaining processing time* (S/RPT) com SPT. Esta regra difere da anterior, pois no lugar de CR usa-se S/RPT e, neste caso, a data de entrega da k-ésima operação é definida como:

$$d_{kj} = \max \{ t + \gamma(t) \cdot p_{jk}, t + p_{kj} \} \quad \text{com} \quad \gamma(t) = (D_j - (t + P_j)) / P_j = \beta(t) - 1$$

O termo $\gamma(t)$ representa a *slack per remaining processing time* que é a razão entre a data de entrega menos o tempo de término da tarefa e o tempo de processamento restante da tarefa.

3.3.1.2 Heurísticas de busca local

Como dito anteriormente, as regras de despacho embora eficazes do ponto de vista computacional, não garantem a obtenção de soluções de boa qualidade⁴⁵, quando o objetivo é encontrar a solução ótima, enquanto que as técnicas de programação inteira determinam uma solução ótima, mas não são viáveis em problemas de médio e grande porte, devido ao esforço computacional requerido. Diante disso, observa-se na literatura científica a proliferação de heurísticas de busca local que procuram superar as deficiências das regras de despacho e das técnicas de programação inteira. Isto é, as heurísticas de busca local freqüentemente determinam uma solução de boa qualidade a um esforço computacional aceitável.

Heurísticas de busca local podem ser encontradas na literatura, com atenção especial para a heurística desenvolvida por Scrich; Armentano; Laguna (2004), a qual é baseada em busca tabu e usa regras de despacho para obter uma solução inicial, buscando novas soluções em uma vizinhança baseada nos caminhos críticos das tarefas. Esta heurística é importante como referência⁴⁶ para esse trabalho pelas seguintes razões: a) experimentos computacionais mostraram que esta heurística possui uma alta eficácia e b) os fundamentos dessa heurística servirão de base para o método de escalonamento e despacho dessa tese.

⁴⁵ Soluções próximas da solução ótima.

⁴⁶ O termo referência é aqui usado para denotar inspiração para a solução do problema e não que esta heurística será a heurística usada.

Portanto, a heurística desenvolvida por Scrich; Armentano; Laguna (2004) consiste em:

Fase I: Geração de uma solução inicial (viável);

Fase II: Busca de melhores e novas soluções na vizinhança.

3.3.2 Regras Heurísticas Simples *versus* Regras Heurísticas Combinadas

As regras heurísticas para o despacho das tarefas em um *job shop* constituem-se funções matemáticas simples, as quais são derivadas do conhecimento de especialistas e de observações dos sistemas do tipo *job shop*.

Partindo-se da premissa de que nenhuma regra simples pode perfazer consistentemente todas as outras características de outras regras, Ishii; Talavage (1994) usaram uma abordagem baseada em regras de despacho mista (RDM) para o escalonamento de um sistema flexível de manufatura (FMS) através da fusão de quatro regras:

1. *next-in-next-out*;
2. *shortest processing time first* (SPT);
3. *largest slack first*;
4. *first-in-first-out*.

O critério usado para analisar a performance do sistema incluía medidas do tempo de fluxo, do atraso, do tempo de fluxo ponderado, do atraso ponderado, e ainda a combinação deles. Os autores constataram que a RDM melhorou a performance do sistema analisado, se comparado aos melhores resultados usando apenas uma única regra em separado, devido a uma estratégia de busca que se focava nas máquinas gargalo. No entanto, uma constatação desse trabalho foi que a abordagem por RDM não trabalhou eficientemente em escalonamentos multiobjetivo como ela o fez para casos com um único objetivo.

Vários pesquisadores usam lógica *fuzzy* para solucionar problemas multiobjetivo. Watanabe; Tokumaru; Nakajima (1992) combinaram um critério de aproveitamento e folga através de uma função definida entre o aproveitamento e o tempo de completude da ordem de manufatura. Hatono; Suzuka; Umano (1992) desenvolveram um modelo mais generalista para alcançar múltiplos objetivos de performance. Grabot; Geneste (1994) introduziram uma regra de importância que, freqüentemente é negligenciada por pesquisadores, a qual protege a completude tardia das ordens em particular. Combinando a regra SPT, a regra *slack time* e uma regra de importância, eles demonstraram que a junção das regras pode propiciar uma performance comprometida com os objetivos particulares das regras elementares que compuseram essa combinação. Com a mesma suposição, ao misturar-se ou combinar-se regras de prioridade, trabalha-se melhor do que com regras simples com relação a sistemas de manufatura multiobjetivo.

Nesta pesquisa, propõe-se a combinação de regras heurísticas para garantir que características de objetivo de regras individuais possam ser alcançadas de forma balanceada por uma única regra heurística combinada, gerando índices de prioridade para o escalonamento dinâmico das atividades aos recursos, possibilitando a abordagem através de técnica mista de escalonamento com características pró-ativa e progressiva.

3.3.3 Seqüenciamento com operações alternativas

Um dos problemas a ser resolvido pela técnica de escalonamento dinâmico refere-se ao seqüenciamento das atividades. Em sistemas do tipo *job shop* esse seqüenciamento deve obedecer à ordem de precedência para as atividades, estabelecida através de grafos de precedência para os *jobs*, caracterizando-se numa restrição ao escalonamento dinâmico.

A dinâmica de acompanhamento da execução da produção e do escalonamento das atividades, dentro do escopo dessa pesquisa, é implementada por um algoritmo de escalonamento do tipo BB, através de uma regra heurística combinada e de uma fila de tarefas em ordem crescente de prioridade a ser estabelecida pela regra heurística combinada.

3.3.4 Escalonamento de múltiplas tarefas com operações alternativas

Um dos objetivos mais comuns das indústrias é a redução do tempo de ciclo para as ordens de trabalho ou *lead time*. Esse objetivo pode ser alcançado não apenas atribuindo-se rotas para os múltiplos tipos de peças concorrentemente, mas também, disponibilizando operações alternativas (em outras máquinas) para as peças e, com isso, evitando o congestionamento de certos recursos devido a gargalos de produção.

Shmilovici; Maimon (1992) propuseram a adoção de uma estratégia baseada em três regras heurísticas e concluíram que:

1. a flexibilidade de operação aumenta o fluxo da produção enquanto conserva o *lead time*, o que é sempre desejável. No entanto, requer-se com isso um amplo espaço para os *buffers* de processo;
2. aumentando a capacidade dos *buffers*, aumenta-se o fluxo no chão de fábrica, porém até um limite viável;
3. a heurística associada com a capacidade de visão futura dos estados possíveis se tornou cara devido à alta utilização de *buffers* de processo e longos *lead times*, além de impor um alto custo computacional;
4. o trabalho também concluiu que o critério de prioridade deveria ser dado por heurísticas simples, de fácil compreensão e implementação, como o caso da heurística combinada que se proporá nesta tese.

Uma observação importante a fazer é que o estabelecimento de várias ordens de produção concorrentemente, em células de trabalho, nem sempre é benéfica por motivos como:

1. o aumento da frequência de trocas de dispositivos de fixação e de ferramentas, causando conflitos de seqüência e, conseqüentemente, o aumento do tempo de fluxo;
2. requer espaço de estocagem temporário, isto é, um *buffer* interno.

No entanto, o aspecto do primeiro item pode ser contornado pela tecnologia avançada, em que as máquinas atuais permitem que os dispositivos de fixação e as ferramentas possam ser trocados mais rapidamente. Os dispositivos de troca rápida de ferramenta e os *magazines* de ferramenta nos centros de máquinas reduziram significativamente os tempos de *setup*. Por isso, regras heurísticas combinadas podem trabalhar muito bem na resolução de problemas de escalonamento visando o seqüenciamento de múltiplas ordens.

3.3.5 Fatores Econômicos em Problemas de Escalonamento

O atendimento das datas devidas é talvez, um dos mais importantes objetivos dos escalonamentos dos sistemas de produção industrial, especificamente os sistemas com características de demanda orientada a encomendas/pedidos. As conseqüências de entregas tardias, como as penalidades por atraso, o descontentamento dos clientes e os custos de remessas apressadas podem variar significativamente de uma ordem para outra e de um cliente para outro. Por essa razão, um custo ponderado por atraso deveria ser associado à medida de prioridade para as ordens. Examinando seis regras de prioridade bastante encontradas na literatura (VIDAL; BECK, 2003), conclui-se que:

1. *first-come-first-serve* (FCFS): essa regra não considera datas devidas, portanto não pode ser usada nessa pesquisa;
2. *earliest due date* (EDD): essa regra assume que todas as ordens possuem o mesmo custo por atraso, o que não pode ser levado em conta nessa pesquisa por motivos óbvios;
3. *slack per remaining processing time* (S/RPT): Essa regra deriva-se da regra de folga mínima (MSLACK – *minimum slack*). Um índice de folga mínima é calculado a partir da data devida para uma ordem *i* menos o seu tempo de processamento remanescente menos a data corrente. Ela se contrapõe à regra SPT no sentido de que para todas as regras com datas devidas iguais, a ordem que possuir o maior tempo de processamento recebe a prioridade máxima. Por essa razão, a regra S/RPT é usada para compensar essa tendência anti

SPT. Sua fórmula de cálculo é dada pelo índice MSLACK da ordem i dividido pelo tempo de processamento remanescente para a ordem. São considerados custos de atraso iguais para todas as ordens;

4. *weighted shortest processing time* (WSPT): é bem conhecido que a regra SPT propicia um tempo de fluxo mínimo para o escalonamento das máquinas, o que equivale a minimizar o atraso médio. Por isso, a regra SPT poderia afetar o objetivo de minimização do número de ordens atrasadas. A regra WSPT é ponderada, o que representa que ordens diferentes podem ter custos de atraso diferentes;
5. *weighted cost over time* (COVERT): através da característica dinâmica de visão à frente, essa regra considera o tempo esperado para as operações remanescentes. A regra COVERT tem se mostrado superior a outras regras em relação à performance dos atrasos médios. Essa regra também é ponderada;
6. *apparent tardiness cost* (ATC): Também é uma regra dinâmica. A regra segue uma função de folga exponencial que envolve um parâmetro de visão à frente medido em unidade de tempo de processamento médio. Também é uma regra ponderada.

Para atender aos objetivos de sistemas de manufatura do tipo NDDJSS, existe a necessidade de especificação de regras heurísticas simples de despacho que combinem características importantes de outras regras.

Neste trabalho, o algoritmo de escalonamento a ser usado para o escalonamento local das células basear-se-á num algoritmo do tipo BB e numa regra heurística de busca local resultante da combinação das regras SPT e S/RPT (fase I) e da adoção de uma segunda etapa (fase II) para a escolha da melhor da solução⁴⁷ dentre as alternativas geradas pelas rotas de processo alternativas, para as próximas operações possíveis dentre as tarefas.

⁴⁷ O caminho mais distante possível do caminho crítico (pior caso) no momento da escolha.

3.3.6 Regra Heurística Combinada (RHC)

O problema de escalonamento dessa pesquisa, eq.(3.3), refere-se ao problema demonstrado através da notação de Brucker. Para melhor caracterizar o problema de escalonamento e a regra heurística combinada adotados nesta tese, algumas características importantes serão aqui colocadas:

1. chegada contínua de *jobs* e tarefas: as ordens chegam continuamente, dessa forma o escalonamento escolhido é dinâmico;
2. não preempção de atividades: quando uma atividade é processada, as outras atividades não podem interrompê-la;
3. disponibilidade de rotas: rotas alternativas de processo podem ser usadas;
4. tempo de processamento estimado: o tempo de processamento de cada atividade para um *job* qualquer é estimado, sendo que máquinas alternativas podem requerer tempos de processamento diferentes para a mesma operação;
5. os tempos de *setup* são pequenos em relação ao tempo de processamento. Assume-se que o sistema possui troca rápida de ferramentas;
6. o sistema não sofre bloqueio: a execução do sistema possui uma dinâmica que dificulta o bloqueio, pois o algoritmo trabalha com janelas de tempo pequenas para a observação do comportamento futuro baseando-se numa lista de prioridades, e ainda, pode-se escolher dentre as rotas alternativas de processo;
7. a ausência total de movimentação intercelular;
8. eventos inesperados, como a quebra de máquinas e a falta de matéria-prima no momento da execução, devem ser levados em consideração.

A regra heurística mista a ser definida combinará as características das regras heurísticas SPT e S/RPT. Contribuiu para essa escolha um fator importante, onde as regras a serem combinadas foram as que possuíam objetivos mais adequados ao

problema de escalonamento a ser resolvido. Para compor a regra heurística combinada para o problema de escalonamento desta tese, as regras SPT e S/RPT apresentam-se como boas candidatas, pelo fato de priorizarem o aumento do fluxo de produção e o cumprimento das datas devidas, respectivamente. Usando-se a regra SPT, como uma heurística para a determinação do índice de prioridade, pode-se reduzir o tempo de fluxo da produção, e conseqüentemente, reduzir os custos de inventário do sistema de produção. Por outro lado, a regra S/RPT é dirigida à data devida, necessidade premente em sistemas cuja demanda por produtos é orientada a pedidos/encomendas.

Os respectivos índices de prioridade estabelecidos pelas regras heurísticas SPT e S/RPT são mostrados a seguir.

$$S / RPT \text{ _ índice} = \frac{d_i - t - \sum_{q=j}^{m_i} p_{iq}}{\sum_{q=j}^{m_i} p_{iq}} \quad (3.4)$$

$$SPT \text{ _ índice} = \frac{1}{p_{ij}} \quad (3.5)$$

onde:

d_i = data devida do *job i*;

p_{ij} = tempo de processamento da *j*-ésima operação do *job i*;

t = tempo corrente (instantâneo);

m_i = número de operações que o *job i* irá necessitar a partir do tempo *t*;

$\sum_{q=j}^{mi} p_{iq}$ = tempo de processamento remanescente do *job* *i* após a (*j*-1)-ésima operação ter sido completada.

Para o propósito de escalonamento, o tempo de processamento remanescente é considerado como o tempo de processamento que falta para o término de um *job*, que corresponde ao somatório dos tempos de processamento de cada atividade, ainda não processada, dentro do *job*.

Existe a dificuldade de combinar essas duas regras numa única regra por causa do ordenamento das prioridades. A regra S/RPT possui uma ordem crescente de prioridades, o que significa que a ordem, com índice de prioridade de menor valor, corresponde à atividade prioritária. Já a regra SPT possui uma ordem de prioridades decrescente. Para resolver esse conflito, foi introduzida uma inversão de sinal na regra S/RPT e, a regra SPT leva em consideração o tempo remanescente p_{iq} e não mais o tempo p_{ij} , que considera apenas a próxima etapa. O p_{iq} é sempre o maior tempo de processamento para as etapas individualmente, considerando-se as possibilidades geradas pelas alternativas de rota. Neste caso, analisa-se sempre o pior caminho remanescente.

A eq.(3.6) é a representação do índice de prioridade (δ) para a regra heurística combinada e corresponde ao produto das duas regras, mais um ajuste de sinais para o estabelecimento de uma ordem decrescente de prioridade.

$$\delta = \frac{1}{\sum_{q=j}^{mi} p_{iq}} \left[1 - \frac{d_i - t - \sum_{q=j}^{mi} p_{iq}}{\sum_{q=j}^{mi} p_{iq}} \right] \cdot \left[\frac{1}{q_i} \right] \quad (3.6)$$

Finalmente, um fator de quantidade ($1/q_i$) pode ser usado para dar tratamento preferencial aos *jobs* com o menor número de itens sendo processados, quando houver empate em relação ao índice combinado para dois ou mais *jobs*.

Seguindo a ordem de prioridade estabelecida pela RHC, o algoritmo de escalonamento deverá atender as atividades que compõem os *jobs* tendo como critério os valores instantâneos de prioridade para as tarefas dos *jobs* candidatas à alocação de recursos.

Tabela 3.4 - Notação usada pelo algoritmo de BB.

Notação	Significado
δ	Índice de prioridade para a regra heurística combinada.
HP(i)	Representa o i-ésimo horizonte de planejamento.
Cells _i (j)	Representa a j-ésima célula do horizonte de planejamento i.
r _L (iteração)	Lista de recursos livres para a iteração corrente.
E _E (iteração)	Lista de etapas em espera por recursos conforme grafos de precedência.

Na Fig.3.5 é mostrada a representação gráfica do tempo crítico.

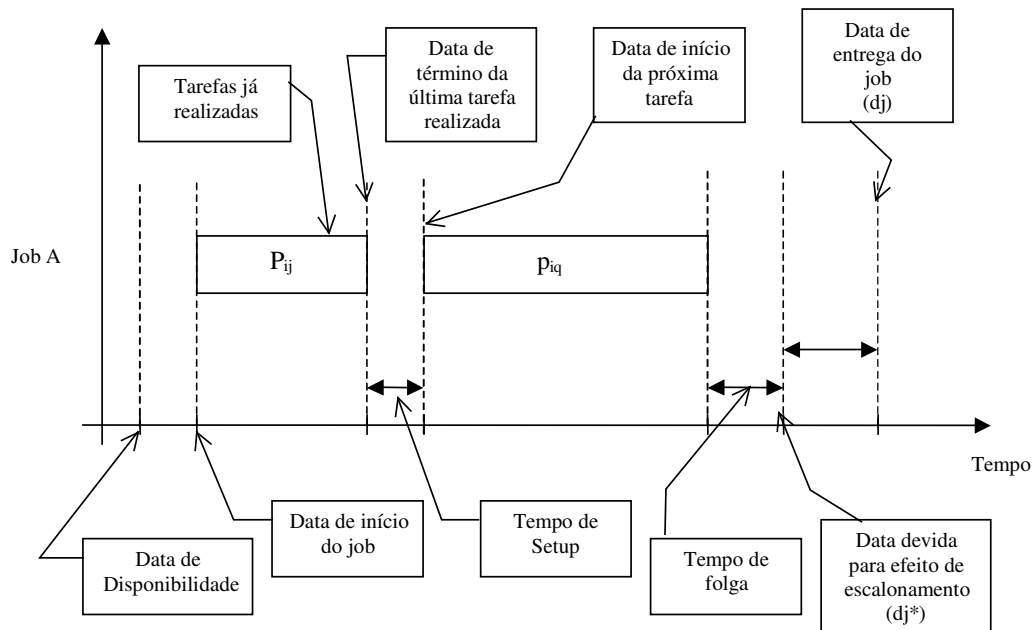


Figura 3.5 - Tempo crítico momentâneo para um *job* em andamento.

O termo “dj*” representa a data devida para efeito de escalonamento e não a data de entrega do pedido ao cliente.

3.3.7 Algoritmo de *Branch&Bound* para o escalonamento e despacho em tempo real

O algoritmo de escalonamento e despacho das células de manufatura virtuais é descrito na Fig.3.6.

Início: inicia HP(i), inicia Cells_i(j) – reserve máquinas (resultado da técnica de *clustering*) para as células virtuais do HP(i), ou seja, não poderão ser usadas por outros processos fora das células virtuais para o HP(i);

Enquanto houver célula ativa no HP(i), repita:

Passo 1: Gerar lista de recursos livres para uso - r_L (iteração);

Passo 2: Gerar lista de etapas em espera por recursos de acordo com os grafos de precedência para os *jobs* - E_E (iteração);

Passo 3: Cálculo do coeficiente de prioridade (δ) somente para as etapas em espera por recursos;

Passo 4: Geração de lista de prioridades para as etapas em espera conforme ordem de valores crescente de (δ);

Passo 5: Designação dos recursos (equipamentos)⁴⁸ para a etapa em espera do topo da lista de prioridade, e também, em concordância com a disponibilidade de máquinas. Os recursos a serem escolhidos para a etapa devem ser aqueles que possibilitem o menor custo e tempo para a etapa;

Passo 6: Atualização da lista de prioridades (sem a etapa já designada no passo anterior);

Passo 7: Se Cells_i(j) terminou, libere máquinas da célula para serem usadas por outra célula virtual, senão vá para diretamente para o passo 8;

Passo 8: Se o HP(i) não terminou, então volte ao passo 2, senão vá para o passo 9;

Passo 9: Fim HP(i) – Libere as máquinas restantes para outros escalonamentos;

Fim repita.

Figura 3.6 – Algoritmo de BB proposto.

Algumas explicações com relação ao algoritmo de BB proposto são necessárias:

⁴⁸ Lembrando que os recursos humanos e materiais não são objeto de estudo dessa tese, de acordo com o exposto anteriormente.

1. um HP(i) contém várias células formadas por vários processos internos. As etapas de cada processo são organizadas em respectivos grafos de precedência. Quando todos os seus processos estiverem terminados a célula estará “desativada”.
2. para gerar a lista de máquinas livres é necessário verificar o *status* de todas as máquinas de todas as células do HP(i).
3. para gerar a lista de etapas em espera, verificam-se todos os processos de todas as células do HP. Uma etapa está em espera quando as condições de precedência estão satisfeitas e, portanto, esperando somente a alocação de um recurso para a sua realização. As etapas em espera devem ter os seus índices de prioridade calculados, e então a lista de etapas deve ser ordenada de forma decrescente conforme esse índice.
4. para se alocar uma etapa a um recurso, verifica-se a disponibilidade dos recursos possíveis de processarem esta etapa. Essa verificação inicia-se com a primeira etapa da lista e segue a ordem de prioridades até que ocorra uma alocação, o que altera o *status* da etapa alocada.
5. ocorrendo a alocação, o algoritmo torna a gerar a lista de máquinas livres, incluindo as alterações nos *status* das máquinas, e também a lista ordenada de etapas em espera, incluindo por exemplo novas etapas que passaram ao *status* de espera, e então realiza mais uma alocação. E assim prossegue-se com sucessivas verificações do *status* atual das máquinas e dos processos e alocação das etapas aos recursos segundo os critérios considerados, até que todos os processos previstos nas células do HP(i) estejam terminados.

3.4 MÉTODO DE ANÁLISE DE RESULTADOS

Para verificar o desempenho do algoritmo de escalonamento e despacho proposto, passa-se a descrever o método de análise e validação que se pretende para o algoritmo.

Em primeiro lugar, necessita-se lembrar que o algoritmo proposto baseia-se numa regra heurística e, que portanto, a validação nesse caso é diferenciada. A abordagem desta tese traduz-se na criação de um modelo de simulação que leve em conta o algoritmo de escalonamento e despacho proposto para designação dos recursos em tempo real. Passa-se, neste momento a descrever o método de simulação e análise dos resultados a serem obtidos pela simulação.

1. criação de um cenário aleatório e artificial de um HP para análise.
2. estabelecimento das características de aleatoriedade temporal⁴⁹ e de indeterminismo provocadas por quebras e paradas de máquinas.
3. repetição de novos cenários e análise do desempenho de cada um dos cenários baseando-se no atraso total para o HP e para as células individualmente como medida de desempenho.
4. análise percentual dos desempenhos insatisfatórios em relação aos satisfatórios.
5. análise gráfica desses desempenhos.
6. e finalmente, apresentação dos resultados e as conclusões finais.

3.5 COMENTÁRIOS FINAIS DO CAPÍTULO

Estabeleceu-se neste capítulo o método de validação da técnica de escalonamento e despacho proposta, bem como a própria técnica de escalonamento e despacho para sistemas do tipo NDDJSS abordados através do paradigma de MCV.

No capítulo seguinte serão apresentados os exemplos de aplicação das novas técnicas de *clustering* e escalonamento e despacho baseado em dois exemplos retirados da literatura técnica e a análise dos resultados obtidos com a simulação através da ferramenta de software desenvolvida para esse propósito.

⁴⁹ O não cumprimento das tarefas de um *job* no tempo estimado (planejado).

CAPÍTULO 4

EXEMPLOS DE

APLICAÇÃO DAS

NOVAS TÉCNICAS E

ANÁLISE DOS

RESULTADOS

Este capítulo está dividido em duas partes. A primeira parte, traz um exemplo de aplicação do algoritmo genético usado na nova técnica de *clustering*. Já a segunda parte, traz dois exemplos de aplicação da nova técnica de escalonamento e despacho. As análises dos resultados foram conduzidas através de um *software* de simulação chamado de DEVICE (*Design of Virtual Cells*), especialmente desenvolvido para esta pesquisa, que possibilita a aplicação do algoritmo genético a processos de uma célula virtual e a análise de desempenho dos processos para cenários de produção artificialmente produzidos, através da nova técnica de escalonamento e despacho.

Este programa de computador possibilita ainda a avaliação de alguns parâmetros dos cenários produzidos, podendo a informação ser armazenada e visualizada em formato textual, gráfico e em arquivos de dados.

A Fig.4.1 mostra a tela de abertura do *software* DEVICE. O programa possibilita armazenar informação em base de dados, aplicar o AG em processos que constituem uma célula virtual para definir as máquinas que compõem a célula e simular cenários de produção usando a técnica de escalonamento e despacho descrita no capítulo3.



Figura 4.1 – Tela de abertura do *software* DEVICE.

Na Fig.4.2 é mostrada a tela de trabalho com o resultado gráfico de uma análise através do algoritmo genético. No gráfico é apresentada a evolução do algoritmo genético para os 64 cenários gerados artificialmente. As populações de indivíduos (rotas) com melhores características genéticas, de acordo com a função *fitness* estabelecida para esta tese são as que apresentaram o “*mean fitness*” igual a 3.

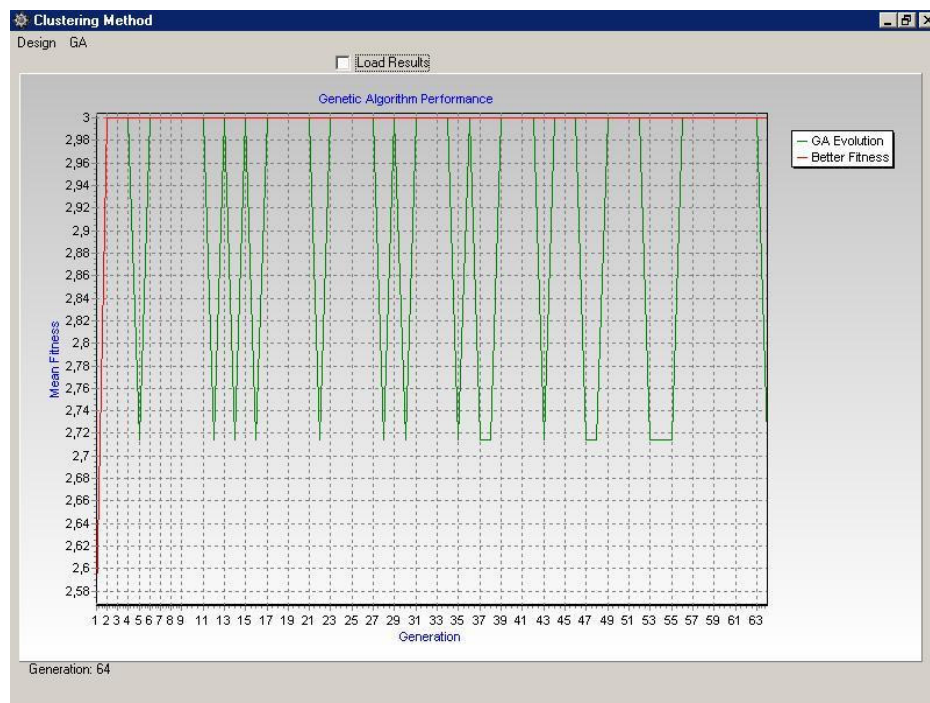


Figura 4.2 – Análise gráfica da evolução do AG através do *software* DEVICE.

Na Fig.4.3 é mostrada a tela de trabalho com o resultado textual de uma análise através do algoritmo genético.

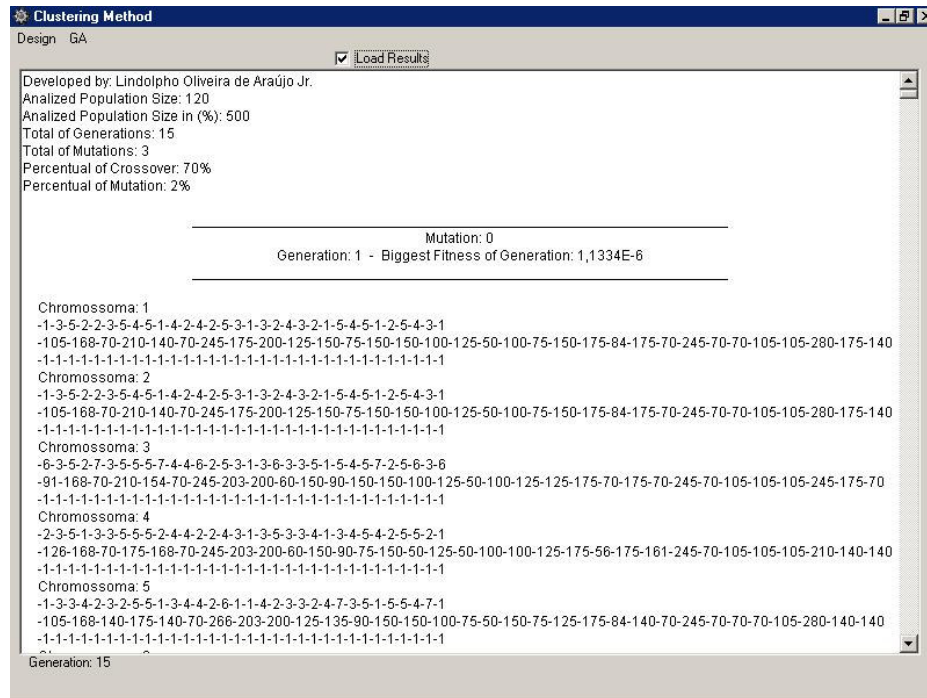


Figura 4.3 – Informação textual da evolução do AG.

Todas as informações sobre os parâmetros aplicados ao AG e os resultados obtidos para cada uma das gerações são armazenados em um arquivo de texto no formato (rtf– *rich text format*) se a opção “load results” estiver selecionada. Para salvar o arquivo é necessário selecionar a opção “salvar resultados em arquivo texto – RTF” no item “GA” do menu principal.

4.1 UM EXEMPLO DE APLICAÇÃO DO ALGORITMO GENÉTICO USADO NA NOVA TÉCNICA DE *CLUSTERING*

Para a análise do AG da técnica de *clustering*, foi extraído da literatura técnica (CHAN, 2003) um exemplo considerado adequado ao tipo de sistema abordado nesta tese, ou seja, sistemas de manufatura do tipo NDDJSS com rotas alternativas de processo. Este exemplo é mostrado na Tabela 4.1. O termo P_{ij} refere-se ao tempo de processamento da tarefa i na máquina j e o termo C_{ij} refere-se ao custo de processamento, respectivamente. O maior problema encontrado nesta tese para as análises do algoritmo genético e do algoritmo de escalonamento, refletiu-se na falta

de exemplos com dimensão maior, como o exemplo hipotético de demonstração adotado na Tabela 2.9.

Tabela 4.1 – Descrição dos Processos Originais de Chan (2003).

Processo	Operação	Máquina	P_{ij}	C_{ij}
1	1	1	105	1
1	1	2	126	1
1	2	3	168	1
1	3	5	70	1
1	4	2	210	1
1	4	1	175	1
2	1	2	140	1
2	1	3	168	1
2	2	3	70	1
2	2	2	112	1
2	3	5	245	1
2	4	4	175	1
3	1	5	200	1
3	2	1	125	1
3	3	4	150	1
3	3	3	135	1
3	4	2	75	1
4	1	4	150	1
4	2	2	150	1
4	3	5	100	1
4	4	3	125	1
4	4	1	75	1
5	1	1	50	1
5	2	3	100	1
5	3	2	75	1
5	3	5	100	1
5	4	4	150	1
6	1	3	175	1
6	1	5	140	1
6	2	2	84	1
6	3	1	175	1
6	4	5	70	1
6	4	3	161	1
7	1	4	245	1
7	1	1	266	1
7	2	5	70	1
7	3	1	70	1
7	3	4	105	1
7	4	2	105	1
8	1	5	105	1
8	1	4	70	1
8	2	4	280	1
8	2	5	210	1
8	3	3	175	1
8	4	1	140	1

No problema original⁵⁰ de Chan (2003), o autor descreve os tempos de processamento unitários. Porém, são feitas análises para diversas demandas por produtos. Os valores de demanda por produtos usados nesta tese são mostrados na

⁵⁰ Chamou-se de problema original os processos propostos por Chan (2003), sem alterações.

Tabela 4.2 e foram escolhidos dentre os perfis de demanda analisados por (CHAN, 2003).

Tabela 4.2 – Demanda por produtos.

Processo	Demanda
1	7
2	7
3	5
4	5
5	5
6	7
7	7
8	7

Nas análises feitas por Chan (2003), o autor encontrou uma medida de desempenho (*makespan*) próxima de 1000 minutos para a demanda escolhida. Chan (2003) usou várias regras de despacho com o objetivo de verificar os efeitos dessas regras juntamente com o uso de rotas alternativas de processo na medida de desempenho de um sistema flexível de manufatura. Para pequenas demandas, o autor concluiu que as regras de despacho têm desempenho muito próximo.

Este pode ser considerado um problema de pequena dimensão por apresentar apenas oito processos. Partiu-se do ponto em que esses foram os processos escolhidos para compor uma célula virtual do sistema, visto que os exemplos encontrados na literatura possuem dimensão de ordem similar ao exemplo escolhido, e que portanto, resultaria em apenas uma única célula ao final da primeira etapa da técnica de *clustering*, que determina o tamanho das células e os processos que irão compor cada célula. Por isso, essa etapa da técnica de *clustering* não foi aplicada neste capítulo. Caso o número de processos fosse bem maior, haveria a necessidade de aplicar os primeiros passos da técnica de *clustering* para definir a quantidade de células do HP, os limites mínimo e máximo para a quantidade de máquinas de cada célula e, finalmente, definir quais os processos que comporiam as células. Parte-se, então, para a aplicação do AG para escolha das melhores rotas de processo e para a definição das máquinas que comporão a célula, juntamente com os processos já identificados. No caso desta tese, os processos do exemplo retirado de Chan (2003).

Todas as possíveis células do HP, onde se encontraria a célula em questão, seriam por definição, balanceadas de acordo com a medida de frequência (volume de trabalho) estabelecida para cada célula do HP. Isso quer dizer que, partindo do pressuposto que a primeira parte da técnica de *clustering* já estaria satisfeita, a questão agora é determinar quais recursos alocar para esta célula, baseado na flexibilidade de rotas e na restrição de tamanho da célula, ou seja, o número máximo de recursos (máquinas) que cada célula poderá comportar, para não comprometer o objetivo de balanceamento quanto à medida de frequência de tarefas da célula. Em outras palavras, como escolher as melhores rotas, que usem o menor número possível de recursos (máquinas), com a melhor medida de desempenho⁵¹ e com o menor custo?

Como forma de obter resposta a essa pergunta, aplicou-se o AG desenvolvido. No caso do problema original, os recursos (máquinas) possuíam alta carga de trabalho, ou seja, uma frequência de utilização dos recursos alta. Isso causa um elitismo natural na população de indivíduos (soluções possíveis). Para obter resultados que pudessem comprovar a eficácia do AG, modificações foram conduzidas, no problema original, de forma a garantir um espaço de soluções mais heterogêneo bem como a varredura no espaço de soluções para poder observar o comportamento do AG na busca da solução.

As modificações introduzidas nada mais foram que o aumento das rotas alternativas de processo, introduzindo-se novas possibilidades de rotas, com o intuito de provocar variações maiores na população de indivíduos (soluções possíveis). A Tabela 4.3 traz essas modificações.

⁵¹ A medida de desempenho é o menor *makespan* possível.

Tabela 4.3 – Descrição dos Processos para o problema modificado.

Processo	Operação	Máquina	P _{ij}	C _{ij}
1	1	1	105	1
1	1	2	126	1
1	1	6	91	1
1	2	3	168	1
1	2	4	140	1
1	3	5	70	1
1	3	3	140	1
1	4	2	210	1
1	4	1	175	1
1	4	4	175	1
2	1	2	140	1
2	1	3	168	1
2	1	7	154	1
2	2	3	70	1
2	2	2	112	1
2	3	5	245	1
2	3	2	266	1
2	4	4	175	1
2	4	5	203	1
3	1	5	200	1
3	1	1	100	1
3	2	1	125	1
3	2	2	60	1
3	2	7	60	1
3	3	4	150	1
3	3	3	135	1
3	4	2	75	1
3	4	4	90	1
4	1	4	150	1
4	1	2	75	1
4	2	2	150	1
4	2	5	75	1
4	3	5	100	1
4	3	4	50	1
4	3	6	100	1
4	4	3	125	1
4	4	1	75	1
5	1	1	50	1
5	1	3	40	1
5	2	3	100	1
5	2	4	150	1
5	3	2	75	1
5	3	5	100	1
5	3	6	125	1
5	4	4	150	1
5	4	3	125	1
6	1	3	175	1
6	1	5	140	1
6	2	2	84	1
6	2	4	56	1
6	2	5	70	1
6	3	1	175	1
6	3	4	140	1
6	4	5	70	1
6	4	3	161	1
7	1	4	245	1
7	1	1	266	1
7	1	3	245	1
7	2	5	70	1
7	2	4	126	1
7	3	1	70	1
7	3	4	105	1
7	3	7	105	1
7	4	2	105	1
7	4	5	70	1
8	1	5	105	1
8	1	4	70	1
8	2	4	280	1
8	2	5	210	1
8	3	3	175	1
8	3	2	140	1
8	4	1	140	1
8	4	2	56	1

4.1.1 Resultados da aplicação do AG ao problema proposto na Tabela 4.1.

Inicialmente, passa-se a explicar os passos adotados na busca da solução, para só então, mostrar os resultados obtidos.

Uma das tarefas mais trabalhosa na aplicação de AG refere-se à sintonia dos parâmetros do AG. O programa desenvolvido permite que esses parâmetros sejam alterados a contento, porém, o próprio programa tenta fazer uma sintonia automática de alguns desses parâmetros. Essa sintonia automática é meramente empírica e baseou-se nas observações do autor dessa pesquisa sobre aplicações de AG.

Resultado da aplicação do AG ao problema original

Os resultados da simulação foram obtidos para os seguintes parâmetros do AG:

- População analisada = 100.
- Gerações = 36.
- Mutações = 6.
- Percentual de *Crossover* = 70%.
- Percentual de Mutação = 2%.

Na Fig.4.4 é mostrada a representação gráfica da evolução do AG obtida pelo programa DEVICE para o problema original.

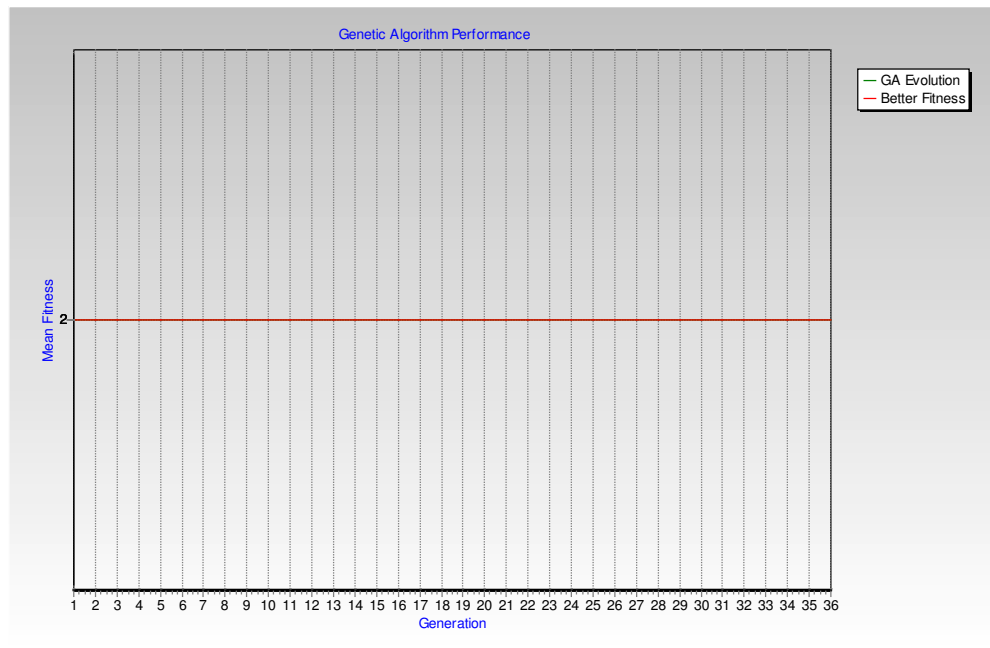


Figura 4.4 – Gráfico de evolução do AG para o problema original de Chan (2003).

O resultado mostrou que a população representada pelas rotas originais é muito homogênea, o que leva a uma situação natural de elitismo⁵². Além disso, a frequência de uso para as máquinas ficou bem próxima, o que garante o uso de todas as máquinas previstas para as rotas com a mesma intensidade.

Na Fig.4.5 é mostrada a frequência de utilização para as máquinas após o término da execução do AG. Essa frequência de utilização calculada pelo programa, não se refere ao tempo de uso das máquinas, e sim, à quantidade de tarefas que usam essa máquina nas gerações encontradas pelo AG.

⁵² Soluções coincidentes para o “*GA evolution*” e o “*Better fitness*”. Todas as gerações encontradas apresentam a mesma medida de desempenho, porém, as gerações não possuem os mesmos indivíduos.

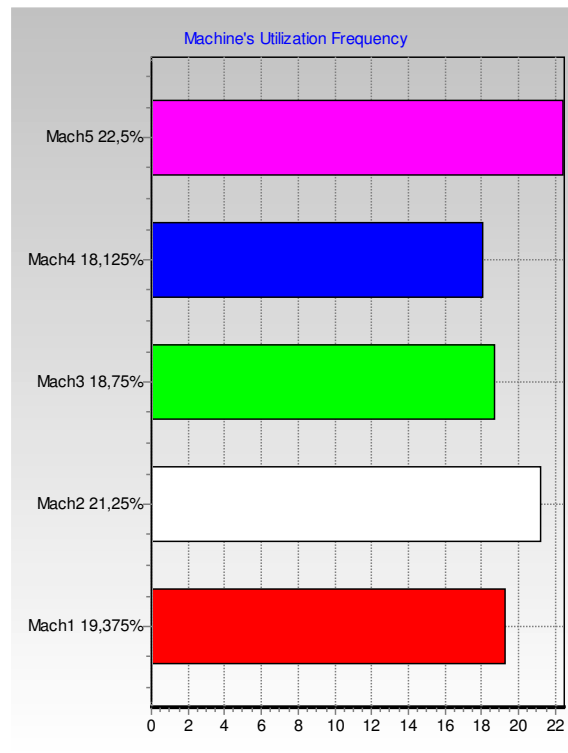


Figura 4.5 – Frequência de utilização das máquinas para o problema original.

Observou-se que o problema original possui uma distribuição homogênea das tarefas dentre as máquinas (volume de trabalho) e que os planos de processo levaram à geração de populações homogêneas.

Na tentativa de ilustrar um pouco melhor a evolução do AG e o seu mecanismo de busca da solução dentro do espaço de soluções possíveis, foram conduzidas modificações nos planos de processo para conferir característica mais heterogênea às populações de indivíduos (soluções). Com um espaço de soluções mais heterogêneo, foi possível perceber mais nitidamente o mecanismo de busca das soluções usado pelo AG.

4.1.2 Resultados da aplicação do AG ao problema proposto na Tabela 4.3.

Resultado da aplicação do AG ao problema modificado.

Os resultados da simulação foram obtidos para os seguintes parâmetros do AG:

- População analisada = 72.
- Gerações = 64.
- Mutações = 8.
- Percentual de *Crossover* = 70%.
- Percentual de Mutação = 2%.

Na Fig. 4.6 é mostrada a representação gráfica da evolução do AG para o problema modificado, obtido através do programa DEVICE.

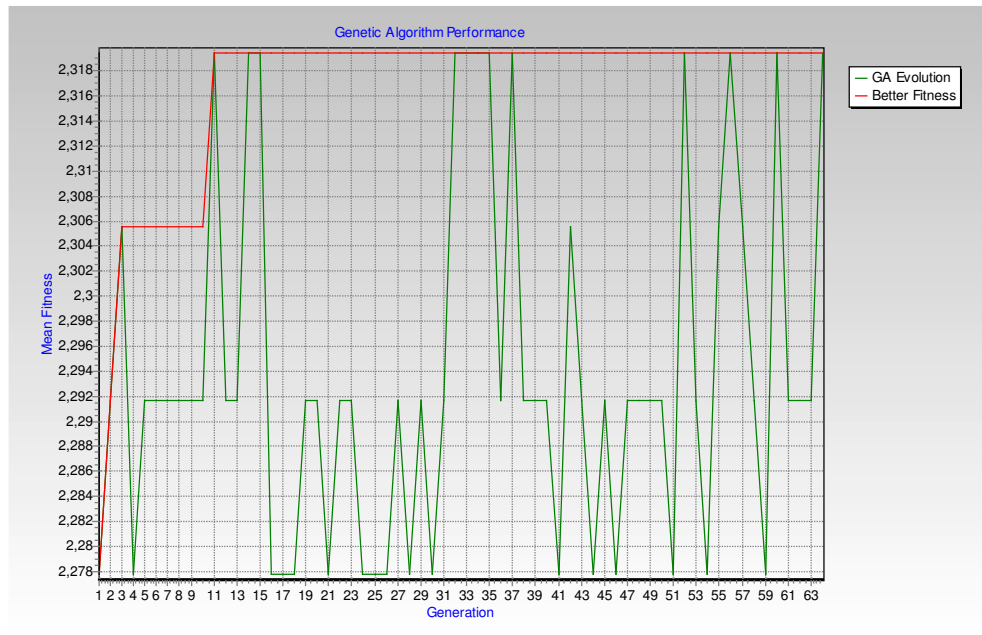


Figura 4.6 - Gráfico de evolução do AG para o problema de Chan (2003) modificado.

O resultado mostrou que a população de indivíduos (soluções) representada pelas rotas modificadas é mais heterogênea do que a população do problema original. Isso se deve ao aumento da flexibilidade de rotas de processo. Além disso, as novas máquinas introduzidas, máquinas m_6 e m_7 , ficaram com uma medida de frequência muito abaixo das outras máquinas como mostrado na Fig.4.7.

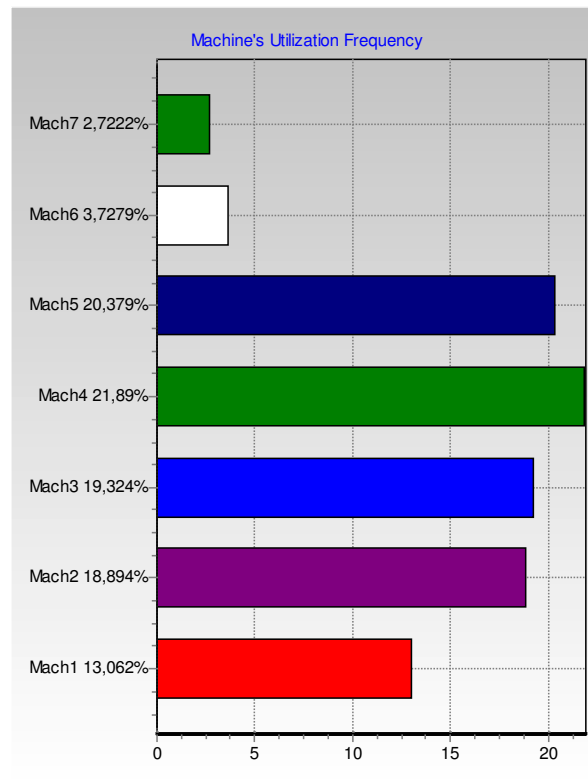


Figura 4.7 - Freqüência de utilização das máquinas para o problema modificado.

Numa análise mais detalhada das simulações de cenários de produção possibilitada pelo programa DEVICE, considerando as rotas que usam essas máquinas e o caso do problema original que não contém essas rotas, verificou-se que a medida de *makespan* do problema modificado ficou abaixo da medida de *makespan* do problema original, como se esperava. Isso comprova que o aumento da flexibilidade de rotas de processo, juntamente com a escolha das melhores rotas que levem à melhor solução considerando como medida de desempenho o *makespan* é verdadeira. Porém, deve-se analisar através de critérios econômicos, até que ponto é benéfico usar mais recursos (máquinas) para produzir, visto que esses recursos melhoram a medida de desempenho do sistema, mas por outro lado, podem ser pouco usados ao longo de um HP, e com isso, serem pouco econômicos.

Para comprovar todas as hipóteses iniciais, realizou-se a análise de dois exemplos de aplicação da técnica de escalonamento e despacho em tempo real. Esses exemplos são mostrados no próximo tópico.

4.2 EXEMPLOS DE APLICAÇÃO DA TÉCNICA DE ESCALONAMENTO E DESPACHO EM TEMPO REAL

Primeiramente, foram escolhidos na literatura técnica, dois exemplos de problema do tipo JSP com rotas alternativas de processo.

Para conferir-lhes características de problemas de escalonamento do tipo NDJSP, o programa DEVICE permite que os tempos de processamento para as operações dos *jobs* possam ser alterados em tempo de execução, como se fossem eventos inesperados, de maneira aleatória, e na proporção desejada pelo projetista. O programa DEVICE permite ainda que o acompanhamento dos processos, para cada cenário de produção criado artificialmente, possa ser feito através da visualização gráfica na tela do computador. As informações dos cenários de produção são passadas através de gráficos de Gantt para as máquinas e para os processos e de um gráfico que totaliza as medidas de desempenho (*makespan*) ocorridas para cada cenário. O programa permite ainda que as penalidades associadas à regra de despacho usada nesta tese, possam ser visualizadas no gráfico. Outra penalidade que pode ser visualizada refere-se ao não cumprimento da data devida. Todas as informações podem ser armazenadas em arquivos no formato gráfico.

Os exemplos conduzidos neste tópico referem-se ao exemplo mostrado na Tabela 4.1 (CHAN, 2003) e ao exemplo extraído de Kumar; Tiwari; Shankar (2003) mostrado na Tabela 4.4, sendo que os tempos de processamento P_{ij} da Tabela 4.4 estão expressos em segundos.

Tabela 4.4 – Representação dos tempos de processamento e das rotas alternativas de processo proposto por Kumar; Tiwari; Shankar (2003).

Processo	Operação	Máquina	P_{ij}
1	1	1	70
1	1	3	40
1	2	2	30
1	3	1	30
1	3	3	60
1	4	1	20
1	4	2	40
2	1	1	80
2	1	2	120
2	2	3	40
2	3	1	70
2	3	2	140
2	4	1	80
2	4	3	40
3	1	1	100
3	1	2	150
3	1	3	80
3	2	2	20
3	2	3	60
3	3	1	20
3	3	3	40
3	4	1	60
3	4	2	30
4	1	2	90
4	1	3	50
4	2	1	60
4	2	3	20
4	3	2	70
4	3	3	120
4	4	1	90
4	4	2	60
4	4	3	30
5	1	1	100
5	1	3	150
5	2	2	70
5	2	3	140
5	3	1	50
5	3	2	80
5	4	1	40
5	4	2	60
5	4	3	80

Kumar; Tiwari; Shankar (2003) propuseram uma abordagem para escalonamento ótimo de sistemas de manufatura flexível através de um algoritmo de colônia de formigas. O resultado obtido pela aplicação do algoritmo desenvolvido pelos autores mostra um gráfico com a medida de desempenho, o *makespan*, variando entre 41,7s e 43,2s. Concluiu-se deste trabalho, que a melhor solução encontrada, representa uma única rota para cada processo, escolhida ao final da execução do algoritmo. De um conjunto de alternativas de rotas, apenas uma única rota por processo foi escolhida ao final, de acordo com a medida de desempenho, o *makespan*. Esse exemplo foi

escolhido não por isso, isto é, a proposta de escalonamento desta tese leva em consideração a utilização de um conjunto das melhores rotas possíveis e assim, precisa-se de um exemplo formado por um conjunto de múltiplos processos, com rotas alternativas de processo, a serem executados num ambiente *job shop*.

O resultado da aplicação do algoritmo de colônia de formigas é apresentado pelos autores como sendo o resultado ótimo para o problema proposto, mas aqui a intenção é comparar o resultado obtido pelo autor com a capacidade da nova técnica de escalonamento e despacho em tempo real de se aproximar desse resultado, visto que as escalas são montadas em tempo de execução num ambiente turbulento.

4.2.1 Resultado da aplicação da nova técnica de escalonamento e despacho ao problema proposto na Tabela 4.4.

Antes de tudo, é necessário esclarecer que o problema proposto por Kumar; Tiwari; Shankar (2003) prevê demanda por apenas 01 unidade de cada produto. Nesta tese, uma modificação foi introduzida no problema original, no sentido de aumentar a demanda pelos produtos, adotando-se uma demanda de 10 unidades para cada produto e tomando-se o valor do melhor *makespan* 41,7s como valor unitário .

Os cenários de produção para o problema da Tabela 4.4 foram gerados da seguinte maneira. Os tempos de processamento sofreram variações percentuais de 0, 1, 2, 3, 4, 5 e 10%, respectivamente. O programa DEVICE produz variações nos tempos de processamento originais dentro de uma faixa estabelecida pelo projetista. Por exemplo, se o projetista especificar um valor de variação de 3%, o programa irá variar os tempos de processamento de algumas operações de alguns processos escolhidas aleatoriamente. Esse valor de 3% significa que os tempos podem variar de 0 a 3% acima ou abaixo de seus valores nominais. A escolha das operações que irão sofrer variação dos tempos de processamento dentro da faixa estabelecida, faz-se mediante uma distribuição normal. O programa estabelece uma distribuição normal, onde os tempos de processamento podem sofrer acréscimo ou decréscimo de acordo com um valor sorteado aleatoriamente. Considerando uma faixa de 0 a 10, os valores sorteados que forem menores que 1, farão com que os tempos de processamento sofram acréscimo e os valores sorteados que forem maiores que 9 farão com que os

tempos de processamento sofram decréscimo. Os valores dentro da faixa de 1 a 9 não alteram os tempos de processamento planejados.

Na Fig.4.8 é apresentado o resultado da medida de desempenho (*makespan*) da simulação de 50 cenários diferentes, para 0% de variação nos tempos de processamento. Todos os 50 cenários gerados apresentaram uma medida de *makespan* igual a 410s para uma demanda por produtos igual a 10.

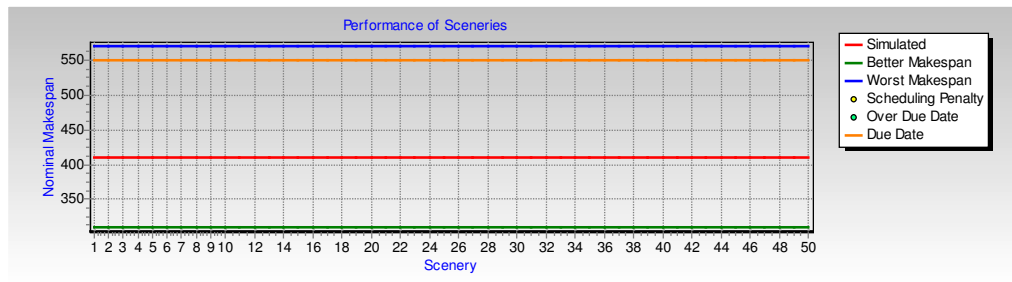


Figura 4.8 – Desempenho do sistema para 50 cenários e variação de 0% no tempo de processamento.

Nas Fig.4.9 e Fig.4.10 são apresentados os gráficos de Gannt⁵³ para os processos e para as máquinas, respectivamente.

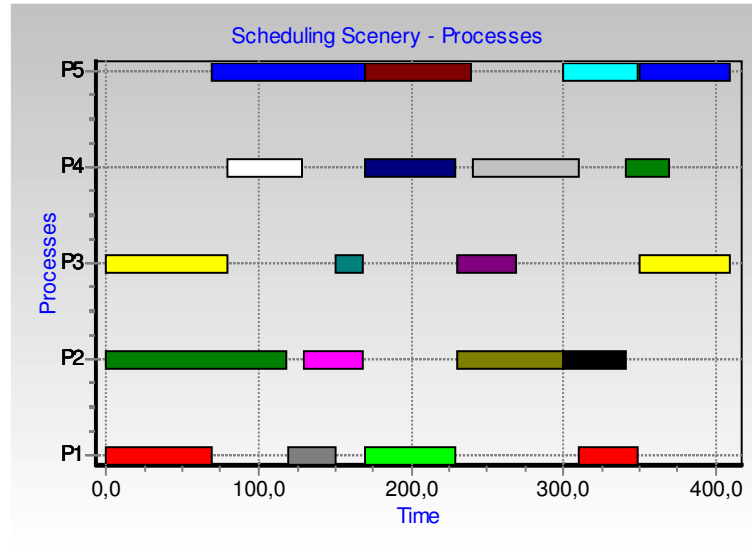


Figura 4.9 – Gráfico de Gannt dos processos do primeiro cenário com variação de 0% no tempo de processamento.

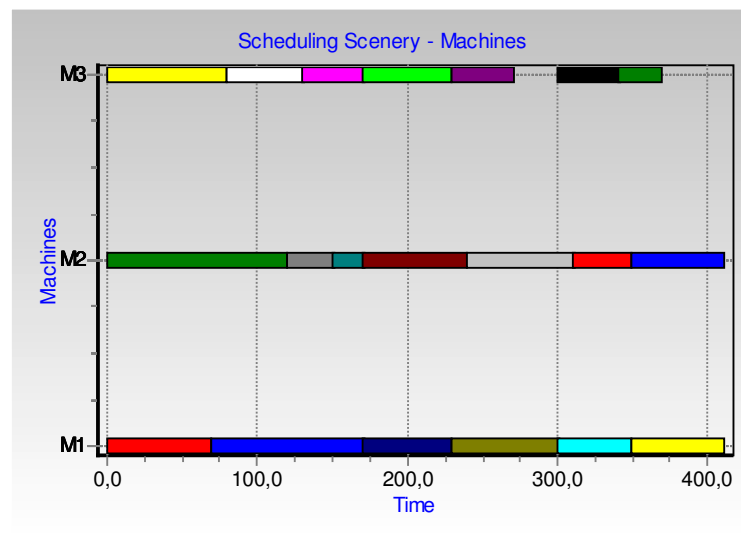


Figura 4.10 – Gráfico de Gannt das máquinas do primeiro cenário com variação de 0% no tempo de processamento.

⁵³ Distribuição das tarefas no tempo.

O resultado obtido pela nova técnica de escalonamento e despacho, quanto à medida de desempenho, o *makespan*, foi superior ao resultado obtido por Kumar; Tiwari; Shankar (2003). A medida do *makespan* médio para as dez peças foi de 41s contra 41,7s da abordagem dos autores. Essa constatação permite concluir que o resultado obtido por Kumar; Tiwari; Shankar (2003) não é ótimo. Também não é possível dizer que o resultado da nova técnica de escalonamento e despacho é ótimo, pelo fato dessa técnica basear-se em heurística. No entanto, conclui-se que o resultado obtido, de acordo com a medida de desempenho, é satisfatório.

Iniciam-se aqui as análises de desempenho da nova técnica de escalonamento e despacho para ambientes turbulentos. A turbulência que se irá introduzir, é a variação aleatória dos tempos de processamento. Foram simulados 50 cenários diferentes para tempos com variação de 1, 2, 3, 4, 5 e 10%, respectivamente. Os resultados são apresentados a seguir.

Nas Fig.4.11, Fig.4.12, Fig.4.13, Fig.4.14, Fig.4.15 e Fig. 4.16 são apresentados os resultados da medida de desempenho (*makespan*) da simulação de 50 cenários diferentes, para as variações de 1, 2, 3, 4, 5 e 10% nos tempos de processamento, respectivamente. Todas as simulações consideraram uma data devida para os processos de 550 segundos e uma demanda de 10 peças.

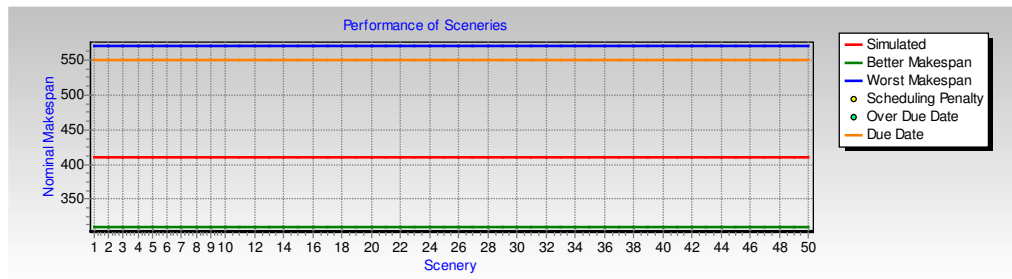


Figura 4.11 – Desempenho do sistema para 50 cenários com variação de 1% no tempo de processamento.

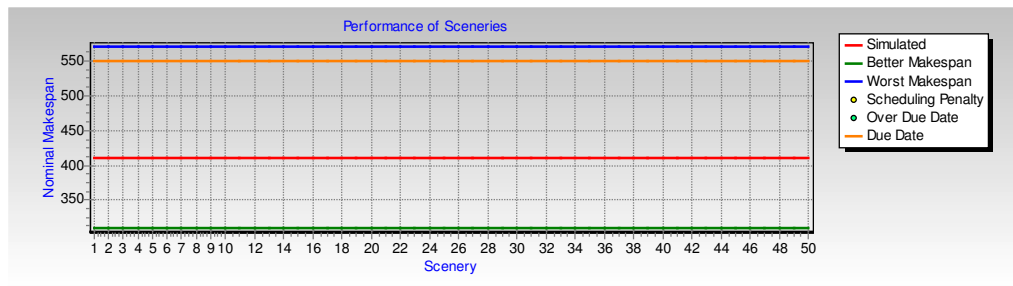


Figura 4.12 – Desempenho do sistema para 50 cenários com variação de 2% no tempo de processamento.

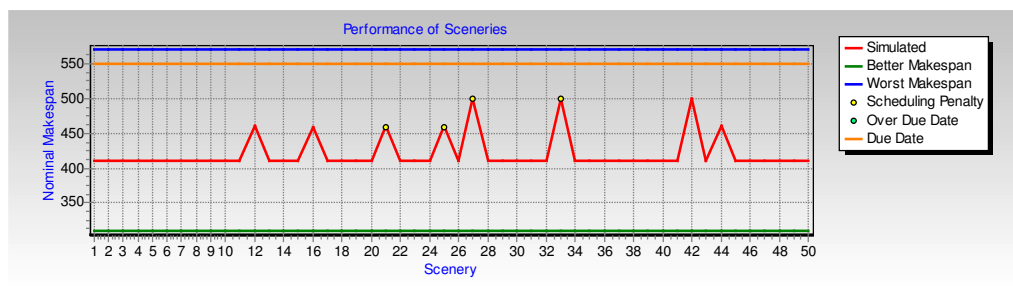


Figura 4.13 – Desempenho do sistema para 50 cenários com variação de 3% no tempo de processamento.

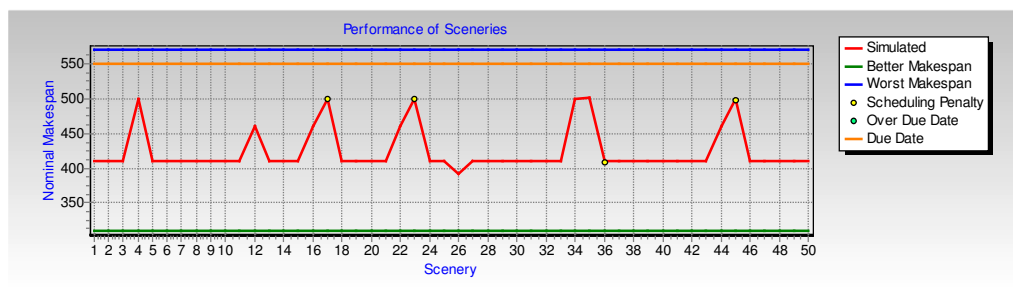


Figura 4.14 – Desempenho do sistema para 50 cenários com variação de 4% no tempo de processamento.

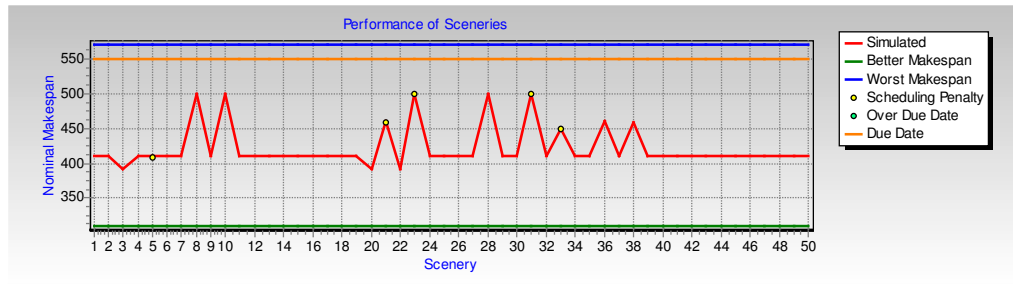


Figura 4.15 – Desempenho do sistema para 50 cenários com variação de 5% no tempo de processamento.

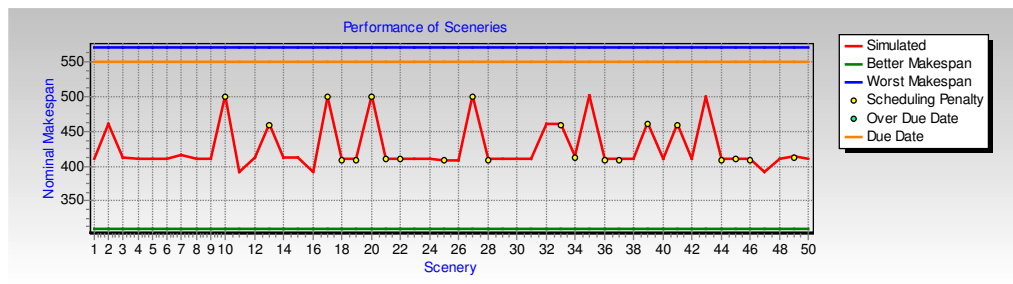


Figura 4.16 – Desempenho do sistema para 50 cenários com variação de 10% no tempo de processamento.

Observou-se que a introdução de turbulências como variações nos tempos de processamento, originadas por fatores como atrasos, paradas momentâneas, quebras de máquinas, paradas para manutenção, dentre outras, resultam em variações na medida de desempenho “*makespan*” para os cenários gerados e, ainda, observou-se que estas medidas não ultrapassaram a data devida, mantendo-se dentro de uma faixa estipulada para o HP.

4.2.2 Resultado da aplicação da nova técnica de escalonamento e despacho ao problema proposto na Tabela 4.1.

O segundo problema de escalonamento tratado nesta tese, refere-se ao problema proposto por Chan (2003) e apresentado na Tabela 4.1.

A medida de desempenho (*makespan*) para a solução ótima encontrada por Chan (2003), considerando a mesma demanda adotada para esta pesquisa, foi de 1000 minutos.

Na Fig.4.17 é apresentado o resultado da medida de desempenho (*makespan*) da simulação de 50 cenários diferentes, para 0% de variação nos tempos de processamento.

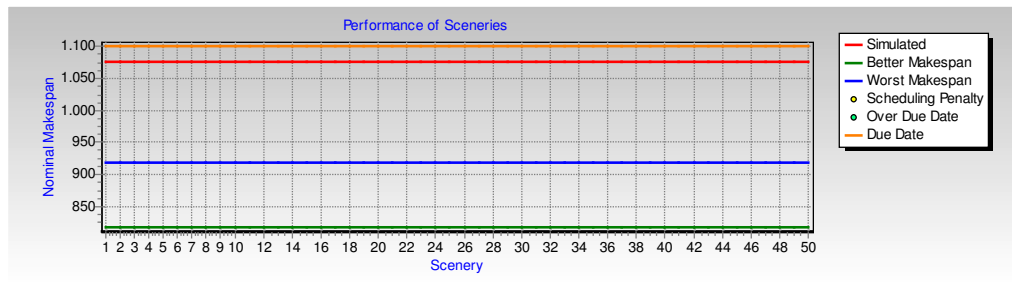


Figura 4.17 – Desempenho do sistema para 50 cenários com variação de 0% no tempo de processamento.

Observou-se que para a simulação de 50 cenários com 0% de variação nos tempos de processamento, ou seja, um problema igual ao problema solucionado por Chan (2003), a solução encontrada não se distanciou muito, mas o *makespan* foi maior que o encontrado pelo autor. Porém, a solução proposta por Chan (2003) não prevê turbulência.

Nas Fig.4.18 e Fig.4.19 são apresentados os gráficos de Gannt para os processos e as máquinas, respectivamente.

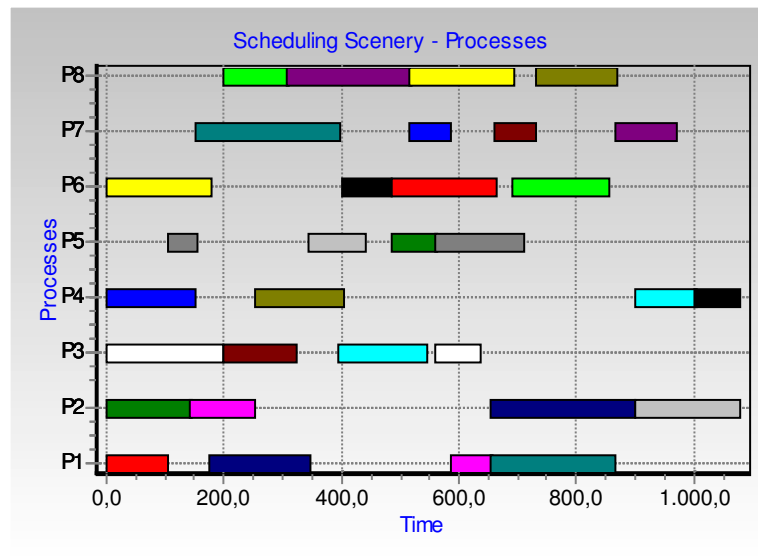


Figura 4.18 – Gráfico de Gannt dos processos do primeiro cenário com variação de 0% no tempo de processamento.

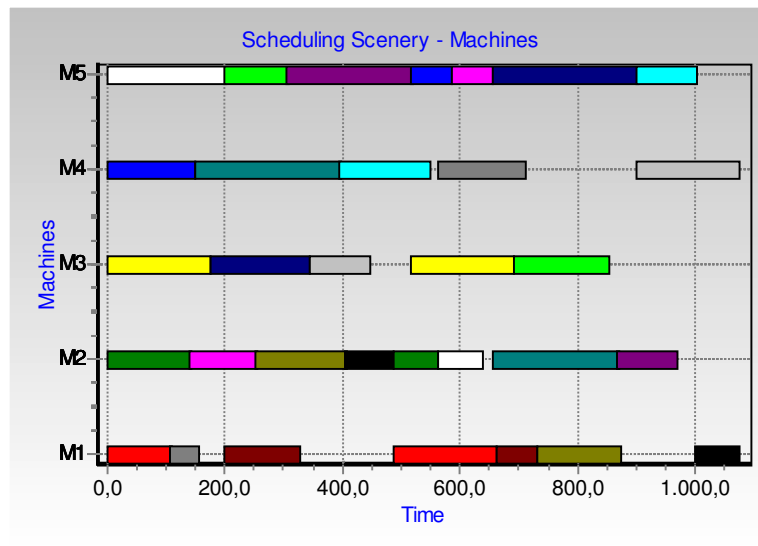


Figura 4.19 – Gráfico de Gannt das máquinas do primeiro cenário com variação de 0% no tempo de processamento.

Iniciam-se aqui as análises de desempenho da nova técnica de escalonamento e despacho para ambientes turbulentos. Foram simulados 50 cenários diferentes para tempos de variação de 5, 6, 7 e 10%, respectivamente. Os resultados são apresentados a seguir.

Nas Fig.4.20, Fig.4.21, Fig.4.22 e Fig.4.23 são apresentados os resultados da medida de desempenho (*makespan*) da simulação de 50 cenários diferentes, para as variações de 5, 6, 7 e 10% nos tempos de processamento, respectivamente. Todas as simulações consideraram uma data devida para os processos de 1100 minutos e uma demanda igual à demanda especificada na Tabela 4.2.

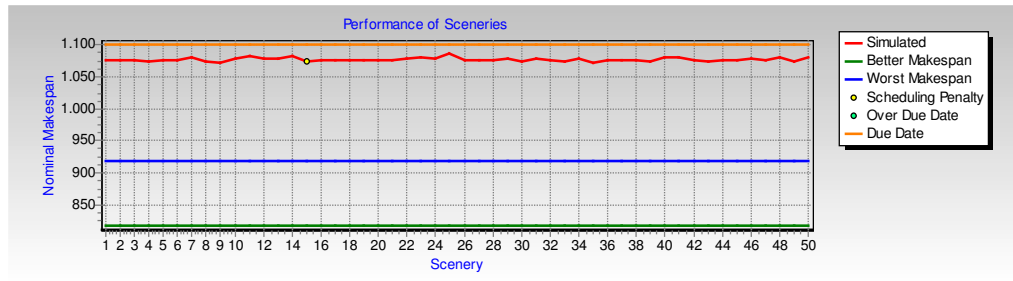


Figura 4.20 – Desempenho do sistema para 50 cenários com variação de 5% no tempo de processamento.

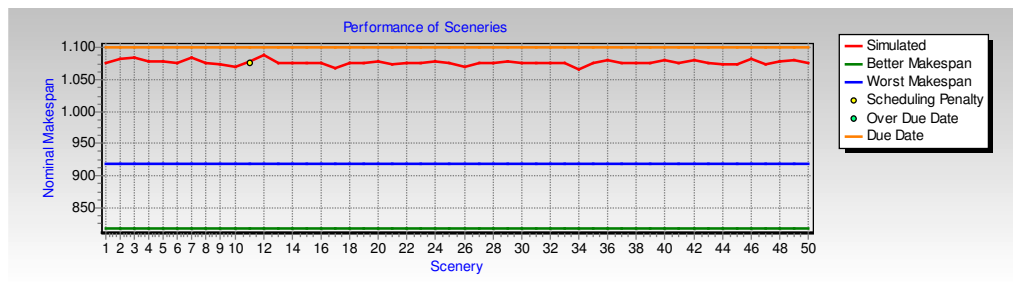


Figura 4.21 – Desempenho do sistema para 50 cenários com variação de 6% no tempo de processamento.

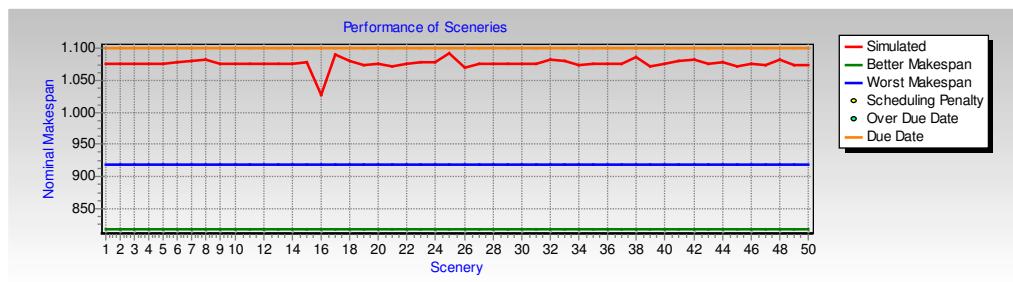


Figura 4.22 – Desempenho do sistema para 50 cenários com variação de 7% no tempo de processamento.

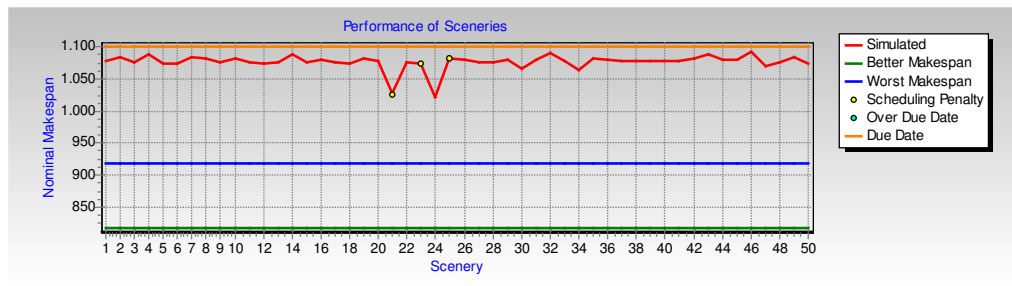


Figura 4.23 – Desempenho do sistema para 50 cenários com variação de 10% no tempo de processamento.

Como no exemplo anterior, as medidas de desempenho mantiveram-se dentro da faixa desejada e com poucas penalidades associadas à capacidade de geração de escalas, pelo algoritmo de escalonamento e despacho.

4.2.3 Análise das penalidades aplicadas às escalas pelo algoritmo de escalonamento

Neste tópico, são analisadas as penalidades aplicadas às escalas geradas em tempo de execução pelo algoritmo de escalonamento e despacho proposto nesta tese. Para ilustrar as penalidades, analisar seus efeitos na construção das escalas e na medida de desempenho para o sistema e mostrar a facilidade de refazer as escalas dos processos penalizados, utilizou-se do cenário de número 23 da Fig.4.23.

Na Fig.4.24 está representado o gráfico de alocação das tarefas⁵⁴ para os processos do problema proposto por Chan (2003) e simulados no programa DEVICE para 50 cenários com variação de 10% no tempo de processamento, referente ao vigésimo terceiro cenário simulado.

⁵⁴ Gráfico de Gannt

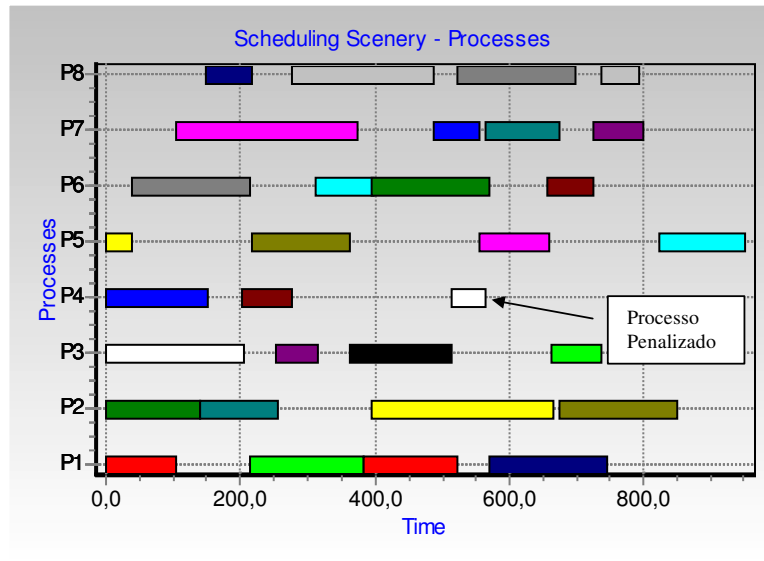


Figura 4.24 - Gráfico de Gannt dos processos do vigésimo terceiro cenário com variação de 10% no tempo de processamento.

Na Fig.4.25 está representado o gráfico de alocação das tarefas para as máquinas do problema proposto por Chan (2003) e simulados no programa DEVICE para 50 cenários com variação de 10% no tempo de processamento, referente ao vigésimo terceiro cenário simulado.

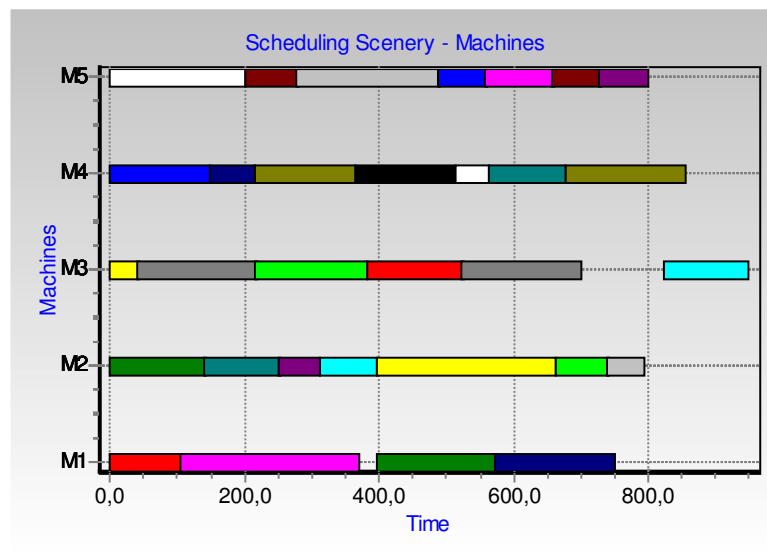


Figura 4.25 - Gráfico de Gannt das máquinas do vigésimo terceiro cenário com variação de 10% no tempo de processamento.

O vigésimo terceiro cenário gerado pelo algoritmo de escalonamento e despacho apresentou uma penalidade relacionada à impossibilidade de geração de toda a escala para o processo de número 4. A última etapa do referido processo não foi escalonada pelo programa DEVICE. Isso não significa que essa etapa não possa ser escalonada. Basta alocar uma máquina para a execução desta tarefa após a execução da última tarefa do último processo.

Essa penalidade gerada pelo programa DEVICE é inerente ao algoritmo de escalonamento e despacho desta tese e serve para visualizar o poder de geração de escalas reativas, sem a interferência humana. O algoritmo observa apenas as informações geradas pelo sistema em tempo de execução. No entanto, quando o ambiente se torna turbulento, com variações intensas de tempos de processamento, penalidades podem ocorrer. Nesses casos torna-se necessário que um operador humano programe uma ou mais etapas para os processos penalizados.

No caso específico do processo de número 4, a última etapa poderia ser realizada nas máquinas de número 3 e 1 com os tempos de processamento em minutos de 125 e 75, respectivamente. A complementação das escalas faltantes deve ser feita após o término da última etapa escalonada pelo sistema. Isso faz com que a medida de desempenho seja pior, ou seja, o *makespan* torna-se maior. No entanto, para todos os cenários simulados nos dois exemplos desta tese, é sempre possível refazer as escalas em caso de penalidades, obedecendo-se a data devida para os processos.

Essas penalidades estão relacionadas aos gargalos de produção e ao uso de um algoritmo baseado em uma fila com os respectivos índices de prioridade para a realização das tarefas. Isso reflete a falta de capacidade da heurística usada de lidar com gargalos de produção em ambientes com muitas perturbações quanto aos tempos de processamento das tarefas. No entanto, para pequenas perturbações, o algoritmo de escalonamento mostrou-se eficaz.

Observou-se que a nova técnica de escalonamento e despacho em tempo de execução consegue trabalhar com eventos inesperados e indeterminismos quanto ao tempo e a seqüência, com pouco ou nenhum prejuízo em relação à medida de desempenho e nenhum prejuízo com relação às datas de entrega. No entanto, quanto maiores as

variações nos tempos de processamento, mais penalidades relacionadas à heurística usada pelo algoritmo de escalonamento, acontecem.

CAPÍTULO 5

CONCLUSÕES FINAIS

O método de programação de sistemas de manufatura do tipo NDDJSS proposto nesta tese, explora pontos positivos da manufatura celular virtual, da computação evolutiva e dos métodos heurísticos de escalonamento e despacho de sistemas de manufatura.

A técnica de escalonamento desta tese mostrou-se autônoma, estável, adaptativa, flexível, contingente, monotônica e relativamente robusta. A robustez foi comprometida pela incapacidade da regra heurística combinada de lidar com gargalos gerados por turbulências no sistema NDDJSS. Porém, isso não comprometeu o bom andamento dos processos, no que tange à construção das escalas em tempo de execução. O único comprometimento ocorrido relacionou-se apenas com a medida de desempenho (*makespan*) e à necessidade de refazer a escala para etapas de processos que sofreram penalidades quanto ao escalonamento.

Como dito, o método de programação mostrou-se adequado ao objetivo inicial, porém a regra heurística combinada possui restrições em lidar com gargalos de produção. Verificou-se a necessidade de incorporar à regra heurística combinada, uma heurística que leve em consideração, possíveis gargalos de produção. Numa primeira avaliação, acredita-se que técnicas de *deadlock avoidance*, aplicadas a sistemas produtivos (NAKAMOTO, 2002), possam ser o caminho para definir regras heurísticas baseadas nessas técnicas para solucionar problemas de gargalo de produção, no que se refere à possibilidade de identificar os gargalos de produção à frente, melhorando a característica progressiva da RHC. Avaliou-se ainda que a característica progressiva da nova técnica de escalonamento e despacho ainda não é totalmente satisfatória. Porém, com rearranjos simples para as últimas tarefas de cada cenário que sofreram penalidades, é possível reconstruir a escala facilmente.

O algoritmo genético mostrou-se adequado para a escolha das melhores rotas, pois possibilitou encontrar soluções factíveis e balanceadas, de forma relativamente rápida e com baixo esforço computacional, para problemas do tipo NP-completo.

Este trabalho contribui nos seguintes aspectos relacionados à programação de tarefas em sistemas do tipo NDDJSS:

- Definição de uma técnica de criação de células de manufatura virtuais disjuntas e balanceadas em relação ao objetivo de formação de células menores e com frequências de tarefas (volume de trabalho) muito próximas, para janelas de tempo determinísticas chamadas de horizonte de planejamento.
- Definição e uso efetivo de rotas alternativas de processo dentro das células virtuais formadas e não a escolha de rotas simples ao final da formação das células, possibilitando a criação de escalas reativas.
- Definição de uma técnica de geração de escalas de execução das operações de manufatura para os processos, em tempo de execução, com alta frequência de utilização para os recursos e medida de desempenho (*makespan*) próximo do valor ótimo, mesmo em ambientes turbulentos.
- Definição de um método de programação de tarefas de sistemas de manufatura do tipo NDDJSS, baseada em células de manufatura virtuais definidas a partir de organizações funcionais e de horizontes de tempo determinísticos.

5.1 TRABALHOS FUTUROS

Observa-se que este trabalho é uma proposta de como um ambiente altamente dinâmico como os sistemas de manufatura do tipo NDDJSS podem ser programados, garantindo-lhes as características de desempenho desejadas, a despeito da complexidade inerente à programação desse tipo de ambiente, alcançando os objetivos de negócios dentro dos perfis de demanda orientados a ordens/encomendas.

Esse método é precursor de uma arquitetura de controle colaborativa que se pretende definir num futuro próximo. O núcleo da atividade de escalonamento e despacho é representado por uma hierarquia dividida em dois níveis, controle das células e controle dos processos dentro de cada célula. Essa atividade de escalonamento representa uma funcionalidade do nível semântico de controle de processos da arquitetura proposta por (MATSUSAKI, 2004).

A solução encontrada para resolver o problema dessa tese aponta para o caminho de como se obter uma solução para o problema de escalonamento e despacho, trazendo para a camada de controle essa atividade. Essa solução também favorece a especificação de uma arquitetura de controle colaborativa para sistemas do tipo NDDJSS. Para isso, torna-se necessário especificar outras atividades inerentes ao controle, não previstas nesta tese, que não apenas as atividades de escalonamento e despacho. Além disso, torna-se necessário trabalhar com uma arquitetura de controle diferente das arquiteturas de controle hierárquicas tradicionais, do tipo *top-down*. O caminho que se propõe refere-se ao tipo de solução proposta por Matsusaki (2004) que propõe uma solução onde as diversas semânticas são separadas em controladores distintos que interagem, entre si, para atingir os objetivos individuais e o objetivo global de controle do sistema produtivo.

Ainda nesta proposta de modelo de controle, problemas como *deadlock avoidance*, problemas de transporte, designação, etc., com semânticas diferentes, devem ser solucionados, somando-se à solução encontrada nesta tese. A solução encontrada nesta tese lida com o nível macro de parte do problema de controle de sistemas do tipo NDDJSS, ou seja, como gerar escalas reativas dentro da camada de controle, possibilitando com isso a implementação das atividades de escalonamento e de despacho em controladores de processo, ou seja, em tempo de execução.

A adoção de um modelo de controle colaborativo dividido nos níveis semânticos de controle dos processos e controle dos recursos, é mais adequada do que arquiteturas hierárquicas do tipo *top-down*, por que segundo Matsusaki (2004), esse tipo de arquitetura permite o tratamento da complexidade do processo global destes sistemas de maneira mais adequada.

Em resposta aos resultados obtidos, apresenta-se como proposta de trabalhos futuros os seguintes itens:

- Melhorar a regra heurística combinada, para conferir à característica progressiva da técnica de escalonamento e despacho, além dos objetivos de aumentar o fluxo de processamento e atender às datas devidas sem penalidades por atraso, um novo objetivo de evitar penalidades devido a gargalos de produção.
- Investigar os mecanismos de solução de problemas de *deadlock avoidance* para propor um procedimento de identificação de gargalos em sistemas do tipo NDDJSS, baseado nos procedimentos de geração de regras adicionais de controle das técnicas de *deadlock avoidance* e traduzir o procedimento de identificação de gargalos em uma heurística para ser combinada à regra heurística proposta nesta tese.
- Ampliar o programa DEVICE para poder simular diversas células ao mesmo tempo, entrada de informação dos processos no formato gráfico através de grafos de precedência e de tabelas associadas a esses grafos e, também, poder comparar resultados da aplicação de mais de uma regra heurística ao mesmo problema e ao mesmo tempo, condições essas necessárias para analisar o modelo de controle que se pretende desenvolver.
- Estabelecer o modelo de controle de sistemas do tipo NDDJSS através da abordagem proposta nesta tese. Esse modelo basear-se-á na proposta de modelagem de Matsusaki (2004), ou seja, um modelo de controle colaborativo, visto que a solução obtida neste trabalho representa apenas uma pequena parte do modelo de controle, ainda no nível macro, mas que era necessário ser resolvido, para só então poder propor o modelo completo de controle de um sistema do tipo NDDJSS.

REFERÊNCIA BIBLIOGRÁFICA

ADIL, G.K.; RAJAMANI, D.; STRONG, D. Cell formation considering alternate routings. *International Journal of Production Research*, v.34, n.5, p.1361-1380, 1996.

ARAÚJO JUNIOR , L.O., MARUYAMA , N., MIYAGI , P.E., MOSCATO, L.A., SANTOS FILHO, D.J.. A control architecture for distributed production systems using a virtual cellular manufacturing and agents society based approach. In: 2003 IEEE International Symposium on Industrial Electronics, 2003, Rio de Janeiro. 2003.

ARAÚJO JUNIOR, L.O., SANTOS FILHO, D.J.. An approach to design control systems for distributed production systems as a collaborative architecture. In: ICAPS International Conference on Automated Planning and Scheduling, 2003, Trento, Italy. 2003.

AZZONE, G.; BERTELE, U. Techniques for measuring the effectiveness of automation and manufacturing systems. *Control and dynamic systems*, v.48, p.1-45, 1991.

BEACH, R.; MUHLEMANN, A.P.; PRICE, D.H.R.; PATERSON, A.; SHARP, J.A. A review of manufacturing flexibility. *European Journal of Operational Research*, v.122, p.41-57, 2000.

BEN-ARIEH, D.; SREENIVASAN, R. Information analysis in a distributed dynamic group technology method. *International Journal of Production Economics*, v.60-61, p.427-432, 1999.

- BROWNE, J.; DUBOIS, D.; RATHMILL, K.; SETHI, S. P.; STECKE, K. E. Classification of flexible manufacturing systems. *The FMS Magazine*, v.2, n.2, p.114-117, 1984.
- BRUCKER, P. **Scheduling Algorithms**. 2.ed. Springer-Verlag, 1998.
- CASAVANT, T. L.; KUHL, J. G. A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems. *IEEE Transactions on Software Engineering*, v.14, n.2, p.141-154, 1988.
- CASSANDRAS, C. G. **Discrete event systems: Modeling and Performance Analysis**. Richard D. Irwin, Inc. and Aksen Associates, Inc. Publishers, 1993.
- CHAN, F. T. S. Effects of Dispatching and Routeing Decisions on the Performance of Flexible Manufacturing System. *International Journal of Advanced Manufacturing Technology*, v.21, p.328-338, 2003.
- CORRÊA, H. L.; GIANESI, I. G. N.; CAON, M. **Planejamento, Programação e Controle da Produção - MRP II / ERP: Conceitos, Uso e Implantação**. 4.ed. Atlas, 2001.
- DARWIN, C. On the origin of species. Disponível em: <http://www.hn.psu.edu/faculty/jmanis/darwin.htm>. Acesso em: 14 Jan. 2003.
- DE PAOLI, F.; TISATO, F. On the complementary Nature of Event-Driven and Time-Driven Model. *Control Eng. Practice*, v.4, n.6, p.847-854, 1996.
- DOS SANTOS, N.R.; ARAÚJO JUNIOR, L.O. Computational system for group technology: PFA case study. *International Journal of Integrated Manufacturing Systems, England*, v.14, n.2, p.138-152, 2003.

- DROLET, J.; ABDULNOUR, G.; RHEAULT, M. The Cellular Manufacturing Evolution. Computers Industrial Engineering, Great Britain, v.31, n.1-2, p.139-142, 1996.
- FRANCIS, L.R.; MCGINNIS Jr, L.F.; WHITE, J. A. **Facility Layout and Location: An Analytical Approach**. Prentice Hall, Englewood Cliffs, NJ, 1992.
- GINDY, N.N.Z.; RATCHEV, T.M.; CASE, K. Component grouping for cell formation using resource elements. International Journal of Production Research, v.34, n.3, p.727-752, 1996.
- GOLDBARG, M. C.; LUNA, H. P. L. **Otimização Combinatória e Programação Linear**. Campus, Rio de Janeiro, 2000.
- GOLDBERG, D. E. **Genetic Algorithms in Search, Optimization, and Machine Learning**. Addison-Wesley publishing company inc., U.S.A., 1989.
- GOULART, C. P. **Proposta de um modelo de referência para planejamento e controle da produção em empresas virtuais**. 2000. Tese (Doutorado) - Escola de Engenharia, Universidade de São Carlos. São Carlos, 2000.
- GRABOT, B.; GENESTE, L. Dispatching Rules in Scheduling: a Fuzzy Approach. International Journal of Production Research, v.32, n.4, p.903-915, 1994.
- GREENE, T. J.; CLEASRY, C. M. Is Cellular manufacturing Right for you? **Proceedings** of Annual International Industrial Conference, p.181-190, 1985.
- GREENE, T. J.; SADOWSKI, R. P. Cellular Manufacturing Control. Journal of Manufacturing Systems, v.2, n.2, p.137-145, 1982.

GUPTA, D.; BUZACOTT, J. A. A framework for understanding flexibility of manufacturing systems. *Journal of manufacturing systems*, v.8, n.2, p.89-97, 1989.

HATONO, I.; SUZUKA, T.; UMANO, M. Towards Intelligent Scheduling for Flexible Manufacturing: Application of Fuzzy Inference to Realizing High Variety of Objectives. In: JAPAN/USA SYMPOSIUM ON FLEXIBLE AUTOMATION, 1992. **Anais**. v.1, p.433-440, 1992.

HO, Y.C. Special Issues on Dynamics of Discrete Event Systems. **Proceedings** of IEEE, v.77, p.3-6, 1989.

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

HUTCHINSON, G.K.; PFLUGHOEFT, K.A. Flexible process plans: their value in flexible automation systems. *International journal of production research*, v.32, n.3, p.707-719, 1994.

HYER, N. L.; BROWN, K. A. The discipline of real cells. *Journal of Operations Management*, v.17, p.557-574, 1999.

ISHII, N.; TALAVAGE, J. J. A Mixed Dispatching Rule Approach in FMS Scheduling. *The International Journal of Flexible Manufacturing Systems*, v.6, p.69-87, 1994.

INSTRUMENTATION, SYSTEMS AND AUTOMATION SOCIETY. **Enterprise-Control System Integration**: Part 1: Models and Terminology, Part 2: Object Model Attributes, Draft 14, Available from The Instrumentation, Systems and Automation Society (ISA), RTP North Carolina, USA, 2000.

KUMAR, R.; TIWARI, M. K.; SHANKAR, R. Scheduling of flexible manufacturing systems: an ant colony optimization approach. *Proc. Instn Mech. Engrs.*, v.217, part B, J. Engineering Manufacture, 2003.

LUCERO, A. G. R. **Um Método de Otimização para a Programação da Manufatura em Pequenos Lotes**. 2001. 99f. Dissertação (Mestrado) - Universidade Federal de Santa Catarina. Florianópolis, 2001.

MANUFACTURING EXECUTION SYSTEMS ASSOCIATION. **International, Controls Definition & MES to Controls Data Flow Possibilities**. White Paper, n.3, 2000.

MATSUSAKI, C. T. M. **Modelagem de Sistemas de Controle Distribuídos e Colaborativos de Sistemas Produtivos**. 2004. 154f. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2004.

MCLEAN, C. R.; BLOOM, H. M.; HOOP, T. H. The Virtual Manufacturing Cell. **Proceedings** of Fourth IFAC/IFIP Conference on Information Control Problems in Manufacturing Technologie, Gathersburg, MD, USA, 1982.

MITROFANOV, S.P. **Scientific principles of group technology**. English Translation, National Library for Science and Technology, Washington, D.C., 1966.

MORAES, C. C.; CASTRUCCI P. L. **Engenharia de Automação Industrial**. LTC – Livros Técnicos e Científicos Editora S.A., Rio de Janeiro, 2001.

MOREIRA, D. A. **Administração da Produção**. Thomson Learning, 2002.

NAKAMOTO, F. Y. **Sistematização do Projeto do Controle de Sistemas Produtivos**. 2002. 128f. Dissertação (Mestrado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2002.

OLIVEIRA, A. C. **Uma contribuição para o problema de programação de tarefas em job shop dinâmico não determinístico com datas de entrega**. 2002. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2002.

- PACHECO, R. F.; SANTORO, M. C. Proposta de Classificação Hierarquizada dos Modelos de Solução para o Problema de Job Shop Scheduling. *Gestão e Produção – Revista do Departamento de Engenharia de Produção*, Universidade de São Carlos, p.1-15, 1999.
- PINEDO, M.; CHAO, X. **Operations Scheduling With Applications in Manufacturing and Services**. Irwin McGraw-Hill, 1999.
- PINTO, A. S. R. **Abordagem de escalonamento dinâmico de tarefas baseada em sistemas classificadores**. 2004. 62f. Dissertação (Mestrado) - Universidade Federal de Santa Catarina. Florianópolis, 2004.
- PORTER, K.; LITTLE, D.; PECK, M.; ROLLINS, R. Manufacturing classifications: relationships with production control systems. *Integrated Manufacturing Systems*, v.10, n.3-4, p.189-198, 1999.
- RABELO, R.; KLEN, A. A. P. Cooperative inter-organizational scheduling. *Gest. Prod.*, v.7, n.3, p.226-246, 2000.
- RAIS, S.; SUGIMURA, N.; KOKUBUN, A. Study on Process Planning System for Holonic Manufacturing (Selection of Machining Sequences and Sequences of Machining Equipment). *JSME International Journal. Series C*, v.45, n.2, p.527-533, 2002.
- RAJMANI, D.; SINGH, N.; ANEJA, Y. P. A model for cell formation in manufacturing systems with sequences dependence. *International Journal of Production research*, v.30, n.6, p.1227-1235, 1992.
- RATCHEV, S. M. Concurrent process and facility prototyping for formation of virtual manufacturing cells. *International Journal of Integrated Manufacturing Systems*, v.12, n.4, p.306-315, 2001.
- RODRIGUES, G. S.; BRANCO, R. M.; ROSA, L. K. Tecnologia de grupo: algoritmos genéticos na solução de problemas de agrupamentos celulares de máquinas e peças. XXIII ENEGEP, 2003.

- ROHDE, L. R.; BOREINSTEIN, D. Representação em espaço de estados para a flexibilidade de roteamento. *Rev. Gestão e Produção*, v.11, n.2, p.251-261, 2004.
- ROSER, C.; NAKANO, M.; TANAKA, M. Constraint Management in Manufacturing Systems. *JMSE International Journal, Series C*, v.46, n.1, p.73-80, 2003.
- SANTOS FILHO, D. J. **Aspectos do Projeto de Sistemas Produtivos**. Tese (Livre-Docência) - Escola Politécnica, Universidade de São Paulo. São Paulo, 2000.
- SARKER, B. R.; BALAN, C. V. Cell formation with operation times of jobs for even distribution of workloads. *International Journal of Production Research*. v.34, n.5, p.1447-1468, 1996.
- SCRICH, C. R.; ARMENTANO, V. A.; LAGUNA, M. Tardiness minimization in a flexible job shop: A tabu search approach. *Journal of Intelligent Manufacturing*, v.15, n.1, p.103-115, 2004.
- SEIXAS FILHO, C. A Automação nos Anos 2000: Uma Análise das Novas Fronteiras da Automação. CONAI Congresso Nacional de Automação, 2000.
- SEIXAS FILHO, C. Manufatura Colaborativa – um novo modelo para uma nova realidade empresarial. *Revista InTech*, n.59, p.52-55, 2004.
- SERRADA, A. P. Una introducción a la computación evolutiva. Disponível em: <http://www.geocities.com/igoryepes/spanish.zip>. Acesso em 08 set. 2004.
- SHARP, J.M.; IRANI, Z.; DESAI, S. Working towards agile manufacturing in the UK industry. *Int. Journal of Production Economics*, v.62, p.155-169, 1999.

- SHIMILOVICI, A.; MAIMON, O. Z. Heuristics for Dynamic Selection and Routing of Parts in an FMS. *Journal of Manufacturing Systems*, v.11, n.4, p. 285-296, 1992.
- SHIVARATRI, N.G., KREUGER, P., SINGHAL, M.. Load Distributing for Locally Distributed Systems. *Computer*, v.25, n.12, 1992, pp.33-44.
- SUN, R.; XIONG, Y.; DU, R.; DING, H. FAS scheduling based on operation flexibility. *Assembly automation*, v.22, n.3, p.277-282, 2002.
- UPTON, D. The management of manufacturing flexibility. *California Management Review*, Winter, p.72-89, 1994.
- VIDAL, T.; BECK, J. C. Practical Approaches to Handling Uncertainty in Planning & Scheduling. ICAPS Tutorial, Trento, Italy, June 10, p.140, 2003.
- WALTER, C. **Planejamento e Controle da Produção - PCP**. [S.l.:s.n.], Apostila de aula: Mestrado em Engenharia de Produção, UFRGS, 1999.
- WATANABE, T.; TOKUMARU, H.; NAKAJIMA, Y. Job-shop Scheduling Using Fuzzy Inference to Make Profit into Account. **Proceedings** of the 1992 JAPAN/USA Symposium on Flexible Automation, ASME, v.1, p.423-427, 1992.
- WHITLEY, D. A genetic algorithm tutorial. Disponível em: http://www.geocities.com/igoryepes/ga_tutorial.zip. Acesso em 08 set. 2004.
- WILHELM, W. E.; SHIN, H. M. Effectiveness of Alternate Operations in a Flexible Manufacturing System. *International Journal of Production Research*, v.23, n.1, p.65-79, 1985.
- WILLIAMS, T. J. A Reference Model for Computer Integrated Manufacturing (CIM): A Description From the Viewpoint of Industrial

Automation, Minutes, CIM. Instrument Society of American, Research Triangle Park, NC. Disponível em:

<http://www.pera.net/Pera/PeraReferenceModel/ReferenceModel.html>.

Acesso em: 10 mar. 2004.

ZOLFAGHARI, S.; LIANG, M. Comparative Study of Simulated Annealing, Genetic Algorithms and Tabu Search for Solving Binary and Comprehensive Machine-grouping Problems. International Journal of Production Research, v.40, n.9, p.2141-2158, 2002.