

# Uma abordagem metodológica para o desenvolvimento de sistemas automatizados e integrados de manufatura

MARCO ANTONIO Buseti de Paula

Eduardo Alves Portela Santos

PUCPR

## Resumo

Este trabalho apresenta uma metodologia de projeto aplicada a sistemas automatizados e integrados de manufatura. A metodologia consiste em um desenvolvimento cíclico de três etapas – modelagem, síntese e implementação – até o atendimento da aplicação demandada para o sistema real, resultando no projeto do sistema automatizado integrado. Esta forma de desenvolvimento permite uma revisão contínua dos resultados obtidos em cada etapa. Para testar e validar a metodologia, é apresentado um exemplo de re-projeto de um protótipo de sistema de manufatura em função da necessidade de inserção de um novo produto.

## Palavras-chave

Automação industrial, manufatura integrada, metodologia de projeto, integração de sistemas, tecnologias de manufatura.

## *A methodological approach to automated and integrated manufacturing systems development*

### Abstract

*This paper presents a methodology of project applied to automated and integrated manufacturing systems. The methodology consists of a cyclic three stages development — modeling, synthesis and implementation — till the accomplishment of the application required by the real system, resulting in the project of the automated and integrated system. This kind of development allows a continuous revision of the results of each stage. To submit to a test and to validate the methodology, it is given an example of a re-project of a prototype of a manufacturing system at the time of the introduction of a new product on the market.*

### Key words

*Industrial automation, integrated manufacturing, design methodology, systems integration, manufacturing technologies.*

## INTRODUÇÃO

Para ser competitiva, a manufatura, mais que qualquer outra atividade ou setor da economia, deve continuamente adaptar-se a mudanças existentes no mercado. O crescimento da competição global está forçando as empresas a reduzir o tempo de resposta de lançamento dos seus produtos e a oferecer preços competitivos. A diversidade, as flutuações de demanda, o curto ciclo de vida dos produtos em função da freqüente introdução de novas necessidades, além do crescimento das expectativas dos clientes em termos de qualidade e tempo de entrega, constituem atualmente os principais desafios com que as empresas devem lidar para permanecerem competitivas e sobreviverem no mercado.

Nesse contexto, agilidade e flexibilidade são conceitos reconhecidos como requisitos para um sistema de manufatura satisfazer as necessidades de um mercado global competitivo. Isso significa que o sistema deve responder rapidamente a mudanças de produção tanto no que se refere a volume como a variedade, de forma efetiva, confiável e a um baixo custo. De maneira geral, o tempo de implantação do sistema automatizado deverá ser o menor possível, uma vez que novos produtos requerem novos requisitos em relação à automação (MOORE *et al.*, 2003).

Entretanto, Erbe (2002) identifica que muitas empresas de manufatura estão encontrando dificuldades em buscar na automação uma estratégia competitiva. De acordo com o autor, a principal razão para esta dificuldade está relacionada a insuficiente flexibilidade em sistemas altamente automatizados. A combinação de perdas em consequência da conversão, o tempo ocioso de equipamentos e o alto custo da manutenção especializada restringem os benefícios econômicos esperados. Assim, muitas empresas estão diminuindo, ou planejam fazê-lo, o nível de automação no chão de fábrica.

No contexto nacional, Moraes e Castrucci (2001) afirmam que as empresas estão dependendo muitos recursos na implantação de um sistema automatizado. Para indicar tal fato, os autores mostram que historicamente até 90% dos esforços de programação são realizados após o *start-up* inicial do sistema. Isso significa que a maior parcela do tempo utilizada na implantação é dedicada à detecção e correção de falhas e a otimizações, em uma etapa posterior à instalação do sistema. Nesse contexto, uma série de trabalhos propõe abordagens inovadoras para lidar com o ciclo de desenvolvimento de sistemas automatizados de manufatura.

Podem-se citar, inicialmente, os trabalhos que utilizam formalismos matemáticos para a construção da estrutura de controle de um sistema de manufatura. Nesse caso, tais formalismos refletem os diferentes tipos e objetivos na aná-

lise dos sistemas em estudo. Cassandras e Lafortune (1999) enumeram os principais modelos utilizados: Redes de Petri com e sem temporização, Redes de Petri Controladas com e sem temporização, Cadeias de Markov, Teoria das Filas, Processos Semi-Markovianos Generalizados e Simulação, Álgebra de Processos, Álgebra Max-Plus, Lógica Temporal e Lógica Temporal de Tempo Real, Teoria de Linguagens e Autômatos.

Dentre os modelos citados anteriormente, dois merecem atenção: as Redes de Petri Controladas (HOLLOWAY *et al.*, 1997) e o modelo de Ramadge e Wonham (1989), denominado Teoria de Controle Supervisório (TCS), baseado em Linguagens e Autômatos (CARROL; LONG, 1989). Diferentemente dos outros modelos que enfatizam a análise de sistemas, além de construir os controladores através de ajustes, os dois modelos citados são dotados de procedimentos de síntese de controladores. Estes modelos, por esta característica, têm dado forte contribuição ao desenvolvimento da teoria de controle de sistemas automatizados de manufatura.

O modelo proposto por Ramadge e Wonham (1989) propicia um processo automático de síntese de supervisores, ao invés dos usuais procedimentos manuais ou heurísticos. Além desta vantagem, o procedimento de síntese de controladores impõe que o supervisor obtido sempre atende as especificações de controle. Assim, novos controladores podem ser rapidamente e automaticamente projetados quando o sistema é modificado ou os objetivos de controle trocados. Além disso, a idéia de síntese do supervisor minimamente restritivo, característica desta abordagem, atribui um maior grau de liberdade ao sistema controlado. Por estas razões, o presente trabalho utiliza a TCS como ferramenta formal de obtenção de supervisores para sistemas automatizados de manufatura.

Identificam-se ainda as ferramentas que tratam da síntese da estrutura de controle, baseadas em formalismos matemáticos específicos. Como exemplo, o processo de síntese de controladores na TCS, pode ser realizado através das seguintes ferramentas: TCT (WONHAM, 1999), DESCO (FABIAN; HELLGREN, 2000), UKDES (CHANDRA *et al.*, 2002) e VER (BALEMI *et al.*, 1993). A ferramenta computacional SUPREMICA (AKESSON, 2002) possibilita ainda a tradução de uma linguagem formal (autômatos) numa linguagem normalizada de programação de controladores programáveis. Assim, é possível, a partir de uma estrutura em autômatos, gerar o código de controlador de acordo com a IEC 61131-3 (1998).

O GRAIL, originalmente concebido como uma ferramenta de manipulação de autômatos (RAYMOND; WOOD, 1996), tem agora algoritmos desenvolvidos pelo grupo de Automação da UFSC (CURY, 2001) que tratam todo o processo de síntese de controladores baseado na abordagem de

Ramadge e Wonham (1989). A grande vantagem do Grail reside no fato deste possuir o código-fonte aberto, permitindo uma maior liberdade ao desenvolvimento de módulos de programas de acordo com uma realidade específica. Também permite ao projetista mudanças e inclusões na ferramenta de uma forma mais rápida e sistemática.

Este trabalho apresenta uma abordagem metodológica para o projeto de sistemas automatizados e integrados de manufatura. A abordagem utiliza a Teoria de Linguagens e Autômatos para a modelagem do sistema de manufatura e das especificações (por exemplo, requisitos operacionais, de segurança, de roteiros de produção, dentre outros). Decorrente destes modelos, supervisores modulares são concebidos de acordo com a Teoria de Controle Supervisório. Em seguida, ocorre a implementação dos supervisores modulares conforme estrutura de controle apresentada por Queiroz e Cury (2002b).

Para fins de validação, a estrutura de controle é inicialmente simulada, permitindo ao projetista a substituição ou modificação de modelos. Gradativamente, partes da estrutura são realizadas em dispositivos de controle que estão acoplados ao sistema real. A consolidação da fase de implementação se dá no momento da realização completa da estrutura de controle concebida na fase de síntese.

A abordagem metodológica consiste na integração de formalismos matemáticos (Teoria de Linguagens e Autômatos e TCS), de ferramenta computacional de síntese GRAIL (CURY, 2001), de bibliotecas de modelos de autômatos (de subsistemas e de especificações) e de tecnologias (AMTs – Advanced Manufacturing Technologies) (GOUVÊA DA COSTA *et al.*, 2000), e de técnica de simulação.

A contribuição da abordagem proposta, caracterizada por um ciclo de desenvolvimento – modelagem, síntese e implementação – consiste em tratar o projeto de sistemas automatizados e integrados de manufatura com maior eficiência, eficácia e confiabilidade quando novas aplicações forem necessárias. Tais aplicações podem ser decorrentes da inserção de novos produtos, de modificações em processos e produtos existentes, de novas necessidades de demanda, da modernização tecnológica, dentre outros.

O presente trabalho apresenta uma revisão conceitual do formalismo matemático que envolve a modelagem e o controle de sistemas de manufatura, bem como a técnica de implementação associada. Na seqüência, a metodologia proposta é apresentada e discutida. Como teste de aplicação, a abordagem é utilizada para o reprojeto de integração e automação de um protótipo de sistema de manufatura em função da necessidade de inserção de um novo produto (alteração das especificações). Em seguida, são discutidos aspectos relacionados à eficiência, eficácia e confiabilidade do projeto.

## MODELAGEM E CONTROLE DE SISTEMAS DE MANUFATURA

Os modernos sistemas automatizados de manufatura são caracteristicamente compostos por um grande número de subsistemas responsáveis pela realização de tarefas específicas do processo global. Entre esses subsistemas se encontram tipicamente sistemas de produção, como usinagem, soldagem, pintura, montagem, entre outros; de movimentação, como esteiras, mesas giratórias, veículos autoguiados (AGV); e de armazenamento de material (ASRS) (GROOVER, 2001).

O principal objetivo do controle de sistemas de manufatura é a coordenação dos subsistemas que o compõem, de forma que uma série de tarefas individuais e conjuntas sejam atendidas, garantindo o correto funcionamento global. Diversas abordagens têm sido utilizadas no projeto de controle de sistemas de manufatura, destacando-se a Teoria de Controle Supervisório (TCS) (RAMADGE; WONHAM, 1989). Esta abordagem permite a síntese automática de controladores a partir da modelagem da dinâmica do sistema em malha aberta (modelos dos subsistemas sem nenhuma ação de controle) e na modelagem dos requisitos desejados (especificações).

O modelo proposto por Ramadge e Wonham (1989) faz uma distinção clara entre o sistema a ser controlado, denominado planta, e a entidade que o controla, que recebe o nome de supervisor. A planta é um modelo que reflete o comportamento fisicamente possível dos subsistemas, isto é, todas as ações que estes são capazes de executar na ausência de qualquer ação de controle. Em geral, este comportamento inclui a capacidade de realizar determinadas atividades que produzam um resultado útil, sem contudo se limitar a esse comportamento desejado. Por exemplo, dois robôs trabalhando em uma célula de manufatura podem ter acesso a um depósito de uso comum, o que pode ser útil para passar peças de um ao outro. No entanto, cria-se com isso a possibilidade física de ocorrer um choque entre ambos, o que é, em geral, indesejável (CURY, 2001).

O papel do supervisor na TCS é, então, o de exercer uma ação de controle restritiva sobre os subsistemas, de modo a confinar seus comportamentos àqueles que correspondem a uma dada especificação. Uma vantagem deste modelo é a de permitir a síntese de supervisores, sendo estes obtidos de forma a restringir o comportamento da planta apenas o necessário para evitar que esta realize ações proibidas. Desta forma, pode-se verificar se uma dada especificação de comportamento pode ou não ser cumprida e, caso não possa, identificar a parte dessa especificação que pode ser implementada de forma minimamente restritiva. Um critério de aceitação pode então ser utilizado para determinar se, com a parte implementável da especificação, o sistema trabalha de maneira satisfatória.

## Modelagem de Sistemas de Manufatura

Na abordagem proposta por Ramadge e Wonham (1989), o sistema espontaneamente gera eventos, classificados como controláveis ( $\sigma \in \Sigma_c$ ) quando o evento pode ser desabilitado pelo supervisor, ou não controláveis ( $\sigma \in \Sigma_u$ ). Seja  $\Sigma^*$  o conjunto de todas as cadeias finitas de elementos em  $\Sigma$ , incluindo a cadeia vazia  $\varepsilon$ . Uma linguagem é um subconjunto de  $\Sigma^*$ . O comportamento de um determinado sistema pode ser modelado por linguagens que, quando regulares, são representadas por autômatos. Um autômato é uma quintupla  $G = (\Sigma, Q, \delta, q_0, Q_m)$ , sendo  $\Sigma$  o conjunto de eventos,  $Q$  o conjunto de estados,  $q_0 \in Q$  o estado inicial,  $Q_m \subseteq Q$  o subconjunto de estados marcados, e  $\delta : \Sigma \times Q \rightarrow Q$  a função de transição definida em cada estado de  $Q$  para um subconjunto de  $\Sigma$ . Então,  $G$  é caracterizado por duas linguagens: a correspondente ao seu comportamento fechado  $L(G)$  e a correspondente ao seu comportamento marcado  $L_m(G)$ . O prefixo fechamento de uma linguagem  $L$  é denotado por  $\bar{L}$ . Autômatos são ilustrados por diagramas de transição de estados que são grafos direcionados onde os nós representam os estados e os arcos rotulados representam os eventos. Os estados marcados são representados por círculos concêntricos e o estado inicial por um arco de entrada. Arcos com um traço representam eventos controláveis.

De acordo com Queiroz e Cury (2002a), no projeto de sistemas de maior complexidade, a modelagem das diversas partes envolvidas é geralmente um passo intermediário na representação do comportamento conjunto do sistema. Tais sistemas são geralmente modelados pela composição de subsistemas de menor porte, podendo estes ser assíncronos entre si. Considerando a estrutura descentralizada do sistema de manufatura, pode-se representá-lo como um conjunto de subsistemas completamente assíncronos, denominado Sistema Produto (RAMADGE; WONHAM, 1989).

Seja um sistema produto composto de subsistemas  $G_i = (\Sigma_i, Q_i, \delta_i, q_{0i}, Q_{mi})$ ,  $i \in I = \{1, \dots, n\}$ , e, para  $j = 1, \dots, m$ , sejam as especificações genéricas,  $E_{gen,j}$ , linguagens definidas, respectivamente, em  $\Sigma_{gen,j} \subseteq \Sigma$ . Para  $j = 1, \dots, m$ , a planta local  $G_{loc,j}$  correspondente a  $E_{gen,j}$  é definida por  $G_{loc,j} = \parallel_{i \in I_{loc,j}} G_i$ , com  $I_{loc,j} = \{k \in I \mid \Sigma_k \cap \Sigma_{gen,j} \neq \emptyset\}$ , onde o operador  $\parallel$  representa a composição síncrona (RAMADGE e WONHAM, 1989). Assim, a planta local  $G_{loc,j}$  é composta somente pelos subsistemas originais diretamente restringidos por  $E_{gen,j}$ . Para  $j = 1, \dots, m$ , a especificação local  $E_{loc,j}$  é definida como  $E_{loc,j} = E_{gen,j} \parallel L_m(G_{loc,j})$  (QUEIROZ; CURY, 2002a).

Na TCS, o comportamento de um sistema de manufatura é modelado como uma seqüência de eventos. O conjunto de todas estas seqüências forma uma linguagem que representa todas as possíveis execuções do sistema. O problema básico de controle supervisiório é modificar o comportamento em

malha aberta do sistema através da eliminação de seqüências indesejadas de eventos, restringindo o seu comportamento à especificação desejada.

## Síntese Modular

A abordagem proposta por Ramadge e Wonham (1989) apresenta um problema, que é o fato de que a solução do problema de síntese é dada na forma de um supervisor único, representado por um autômato que pode ter um número grande de estados. Isto gera um problema de legibilidade da lógica resultante, ou seja, o engenheiro, tendo dificuldades de “interpretar” esta lógica, tem em geral resistência em aceitá-la. Além disso, a implementação de um programa de controle que implemente este supervisor monolítico é pouco estruturada.

Uma solução para estes problemas é a abordagem modular de síntese proposta por Queiroz e Cury (2002a), que explora o caráter modular do sistema de manufatura e das especificações. O resultado é um conjunto de supervisores de pequeno porte que atuam conjuntamente. Esta forma de resolver o problema leva a lógicas elementares de maior legibilidade e permite uma melhor estruturação do programa, que pode ser dividido em módulos de controle. Além disso, estes módulos muitas vezes podem ser implementados de forma distribuída, em diferentes processadores atuando sobre partes do sistema. Na abordagem modular o sistema é representado através de um conjunto de autômatos assíncronos e as especificações são expressas localmente de acordo com os subsistemas afetados. Supervisores ótimos são sintetizados a partir das especificações locais e têm posteriormente seu tamanho reduzido por um algoritmo de minimização.

Neste artigo, um supervisor é representado por um gerador  $S$ , cujas mudanças de estado são ditadas pela ocorrência de eventos na planta  $G$ . A ação de controle de  $S$ , definida em cada um dos seus estados, é desabilitar em  $G$  eventos que não podem ocorrer em  $S$  após uma cadeia observada de eventos. O comportamento do sistema controlado  $S/G$  pode ser descrito pelo autômato  $S||G$ . Também, além de restringir o comportamento da planta, o supervisor desmarca estados. Em outras palavras, uma tarefa do sistema em malha fechada é considerada completa somente quando ela é marcada pela planta e pelo supervisor.

Um supervisor  $S$  é não bloqueante se  $L_m(S/G) = L(S/G)$ . A condição necessária e suficiente para a existência de um supervisor não bloqueante  $S$  que satisfaça uma dada especificação  $K \subseteq L_m(G)$  ( $L_m(S/G) = K$ ) é a controlabilidade de  $K$  (RAMADGE; WONHAM, 1989).  $K$  é controlável (com relação a  $G$ ) se a classe de linguagens controláveis contidas em  $K$  possui um elemento supremo  $\bar{K}\Sigma_u \cap L(G) \subseteq \bar{K}$  chamado  $SupC(K, G)$ . Quando o problema de controle compreende múltiplas especificações pode-se projetar um supervisor

monolítico para o conjunto global de especificações ou um supervisor modular para cada especificação. Na abordagem modular, para  $j=1, \dots, m$ , supervisores não bloqueantes  $S_{loc,j}$  são obtidos diretamente a partir das especificações locais  $E_{loc,j}$  tal que  $Lm(S_{loc,j}/G_{loc,j}) = SupC(E_{loc,j}, G_{loc,j})$ .

### Ciclo de Desenvolvimento da Abordagem Modular

De acordo com a abordagem modular, o funcionamento livre (sem controle) dos diversos subsistemas pode ser modelado como um conjunto de autômatos assíncronos (sem eventos em comum). Dessa forma, Queiroz e Cury (2002a) afirmam ser viável a obtenção de uma representação por Sistema Produto, conforme discutido na seção anterior. De acordo com a aplicação demandada ao sistema de manufatura, constroem-se especificações que a atendam. Em seguida, procura-se construir um supervisor local para cada especificação, modelando-o apenas em termos dos subsistemas afetados por sua ação. De posse dos supervisores obtidos no processo de síntese, garante-se que a aplicação demandada será atendida.

O ciclo de desenvolvimento da abordagem modular para sistemas automatizados de manufatura pode ser descrito sinteticamente pela seqüência de passos a seguir:

1. Identificar o conjunto de subsistemas envolvidos no sistema de manufatura;
2. Construir o autômato  $G_i$  de cada subsistema  $i$  envolvido da forma mais sintética possível;
3. Calcular a mais refinada Representação por Sistema Produto (RSP), fazendo-se a composição dos subsistemas síncronos;
4. Modelar cada especificação isoladamente, considerando apenas os eventos relevantes;
5. Obter a planta local para cada especificação compondo-se os subsistemas da RSP que tenham eventos em comum com ela;
6. Calcular a linguagem de cada planta local que satisfaça a especificação, através do produto síncrono da cada planta local com sua respectiva especificação;
7. Calcular a máxima linguagem controlável contida em cada especificação local;
8. Verificar a modularidade local das linguagens resultantes;
9. Se não forem modulares, procurar resolver o problema de não modularidade por outra abordagem;
10. Se forem modulares, implementar um supervisor local para cada linguagem controlável.

### Estrutura de Controle Modular Local

Objetivando a realização da abordagem modular numa estrutura organizada, o sistema de controle é programado em três níveis hierárquicos (QUEIROZ; CURY, 2002b), de acordo com a Figura 1. Esta estrutura, cuja dinâmica é explicada a seguir, pode ser implementada em linguagem

de Controladores Programáveis (IEC 61131-3, 1998), em linguagens de PC (C, Java) ou mesmo diretamente em *hardware* (circuito elétrico, pneumático ou hidráulico).

O nível dos Supervisores Modulares (SM) é implementado exatamente como concebido na abordagem proposta por Ramadge e Wonham (1989). O programa atualiza os estados ativos de acordo com a estrutura dos geradores e com as mudanças de estado no nível do sistema produto. Um mapa de realimentação associa os estados ativos a um conjunto de sinais de desabilitação que controla o sistema produto.

O nível do Sistema Produto (SP) tem como principal função executar os comandos que são permitidos e não são desabilitados pelo supervisor. A evolução paralela dos subsistemas assíncronos segue os comandos e respostas do nível das seqüências operacionais, sinalizando então as mudanças de estado para os supervisores. O programa necessita garantir que a ação dos supervisores modulares seja sempre atualizada antes de uma nova transição ocorrer no sistema produto. Este cuidado evita o envio de mais de uma mudança de estado para os supervisores instantaneamente.

O nível Seqüências Operacionais (SO) trabalha como uma interface entre o Sistema Produto teórico e o sistema real. Neste nível, o programa interpreta os comandos abstratos como seqüências lógicas que guiam a operação de cada subsistema particular. Este nível mais baixo de controle gera os sinais de saída e lê os sinais de entrada, fornecendo ao Sistema Produto respostas lógicas que refletem a ocorrência de eventos não controláveis.

Durante a etapa de modelagem do sistema físico é recomendado realizar uma abstração dos detalhes de implementação tecnológica das diversas atividades e funções realizadas pelo sistema físico. Este detalhamento é realizado no nível das Seqüências Operacionais, possivelmente através de um *Sequential Function Chart* (SFC) (IEC 848, 1988). Os *comandos* recebidos pelas seqüências operacionais estarão relacionados às transições do SFC, e as *respostas*, enviadas pelas mesmas, estarão relacionadas a ações do SFC.

### METODOLOGIA PROPOSTA

Esta seção apresenta a metodologia para o projeto de integração e automação de sistemas de manufatura. A Figura 2 ilustra o ciclo de desenvolvimento do projeto, caracterizado em três etapas: modelagem, síntese e implementação. Na etapa de modelagem seleciona-se, a partir de bibliotecas de subsistemas e de especificações, um conjunto de modelos para a representação do sistema real e da aplicação, respectivamente. O resultado desta etapa consiste em dois conjuntos de modelos: dos subsistemas e das especificações. Na etapa de síntese, os modelos são utilizados na geração dos supervisores modulares, de acordo com a Teoria de Controle

Supervísório (RAMADGE; WONHAM, 1989) e a Teoria de Controle Modular (QUEIROZ; CURY, 2002a). Na etapa de implementação, os três níveis da estrutura de controle (supervisores modulares, sistema produto e seqüências operacionais) são integrados e implementados gradativamente em três fases: simulação; inserção das tecnologias de controle e comunicação; acoplamento gradativo dos subsistemas.

A metodologia consiste em um desenvolvimento cíclico de três etapas – modelagem, síntese e implementação – até o atendimento da aplicação demandada para o sistema real, resultando no projeto do sistema automatizado e integrado. Esta forma de desenvolvimento permite uma revisão contínua dos resultados obtidos em cada etapa. Por exemplo, o projetista poderá receber uma nova necessidade (revisão da aplicação) e selecionar novos modelos de especificações ou de subsistemas que atendam adequadamente a esta nova necessidade. Outro exemplo pode ocorrer em casos de manutenção do sistema de controle devido à ocorrência sistemática de erros, panes ou falhas do sistema de manufatura. Nesse caso, novos modelos de especificações podem ser integrados sistematicamente ao modelo já existente, resultando uma nova estrutura de controle.

Uma biblioteca de Tecnologias Avançadas de Manufatura – AMTs dá suporte a base tecnológica na definição das seqüências operacionais relacionadas aos subsistemas e na implementação dos supervisores modulares. Segundo Gouvêa da Costa *et al.* (2000), o conceito de AMT abrange aparatos de base numérica e computacional (*software* e *hardware*), projetados para realizar ou suportar atividades e tarefas da manufatura.

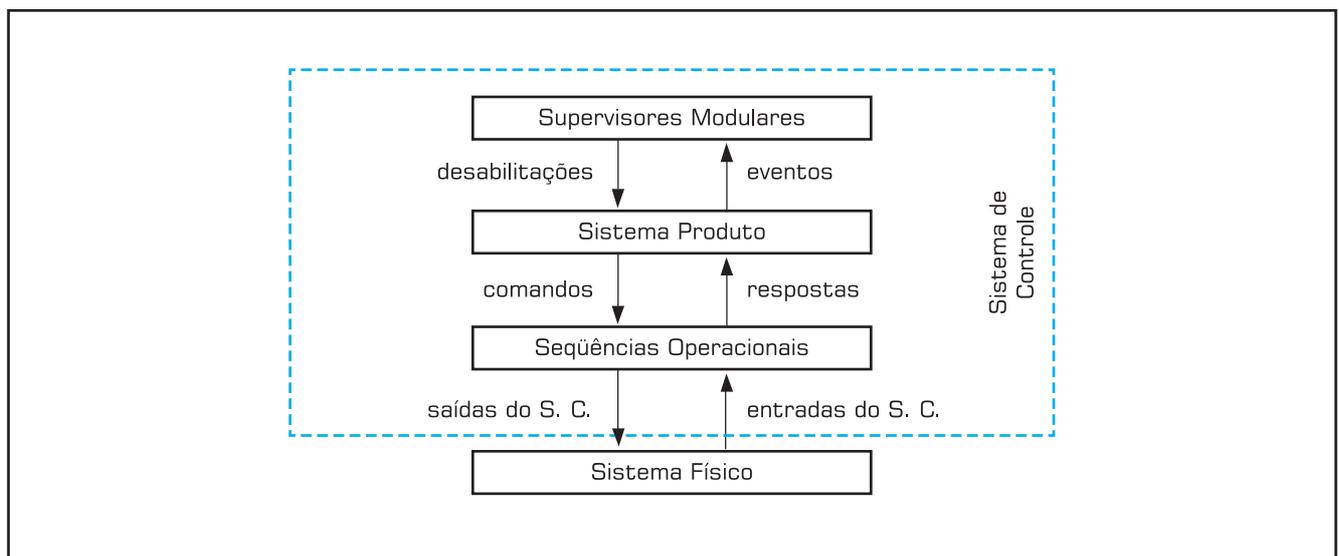
### Etapa de Modelagem

De acordo com o ciclo de desenvolvimento apresentado na Figura 2, inicialmente deve-se selecionar modelos adequados de todos os subsistemas que compõem o sistema real. Em seguida, selecionar modelos de autômatos que representem as especificações a serem aplicadas ao sistema, sendo que estas definem a coordenação de todos os subsistemas. Em muitos casos a modelagem dos subsistemas consiste em uma tarefa relativamente simples, uma vez que a própria configuração espacial do sistema real permite ao modelador selecionar os diversos subsistemas existentes. A atenção aqui deve ser considerada na identificação dos estados de cada um dos equipamentos. Deve-se compatibilizar as funções de coordenação esperadas pelo sistema de supervisão com a correta identificação dos estados dos diversos subsistemas que compõem o sistema real. De maneira geral, nos subsistemas que constituem o sistema de manufatura podem-se identificar os seguintes estados:

- Estado inativo (geralmente estado inicial);
- Estados ativos ou em funcionamento (um subsistema eventualmente pode ter diferentes estados de funcionamento);
- Estado de quebra ou falha;
- Estado(s) intermediário(s).

Na modelagem das especificações, faz-se a construção do conjunto de modelos correspondentes aos requisitos a serem impostos aos subsistemas. Cassandras e Lafortune (1999) definem classes de especificações comumente presentes em sistemas a eventos discretos (nos quais estão incluídos os sistemas de manufatura). Os autores consideram quatro

Figura 1: Estrutura de controle proposta por Queiroz e Cury (2002b)

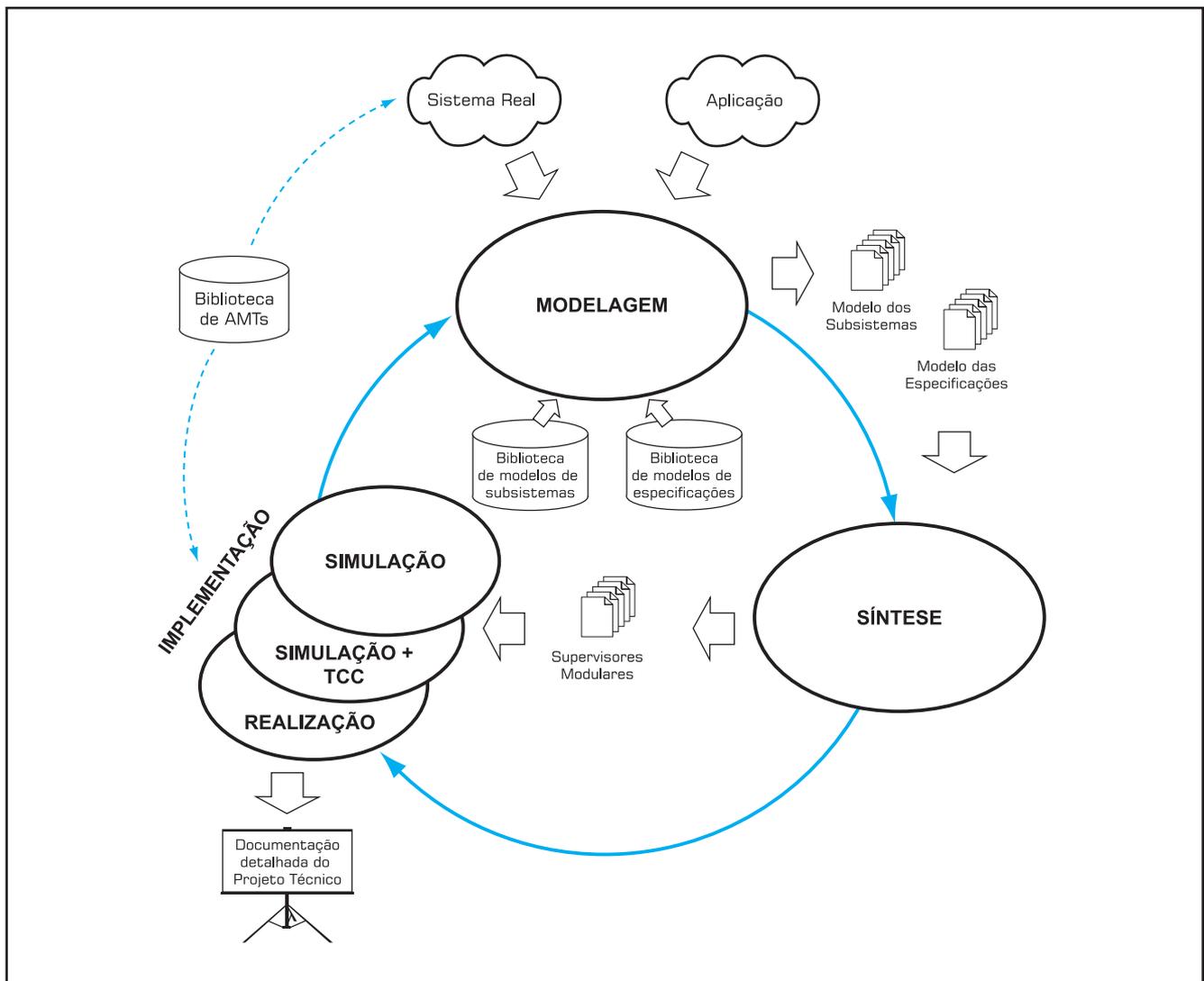


casos em que se baseiam os modelos de especificações: estados proibidos, alternância de eventos, cadeias ilegais e refinamento de estados. No primeiro caso, identifica-se no modelo do sistema real quais estados não são factíveis de ocorrer, em função de restrições físicas ou de segurança. O modelo da especificação é simplesmente obtido pela exclusão destes estados do sistema. No segundo caso, o requisito de coordenação impõe a alternância entre eventos. Por exemplo, a necessidade de alternar dois eventos *a* e *b*, com *a* ocorrendo primeiro, leva à construção de um autômato de dois estados que captura esta alternância. No terceiro caso, identificam-se como ilegais todas as cadeias do modelo do sistema real que contêm determinadas subcadeias. No último caso, necessita-se memorizar como um determinado estado do sistema foi alcançado de forma a especificar qual

comportamento futuro é admissível. Deve-se então refinar tal estado em quantos estados forem necessários.

Em trabalhos anteriores, Santos (2003) propõe a criação de bibliotecas de modelos de especificações relacionadas à configuração física de sistemas de manufatura. O principal aspecto explorado é o tipo de transporte utilizado entre estações de trabalho. Dessa forma, a utilização de transportadores síncronos e assíncronos (GROOVER, 2001) conduz a um grupo particular de especificações. A utilização de posições intermediárias (em função de restrições físicas ou previsão de atraso de transferência de produtos) entre estações de trabalho também gera modelos de especificações particulares. Santos (2003) explora outras diversas configurações possíveis em sistemas de manufatura, e a cada uma destas um modelo de especificação é associado.

Figura 2: Metodologia para a integração e automação de sistemas de manufatura.



Além da utilização de bibliotecas, a etapa de modelagem pressupõe que todos os subsistemas que compõem o sistema de manufatura estão identificados e têm, cada um destes, uma Seqüência Operacional (ver Figura 1) associada.

**Etapa de Síntese**

A viabilidade do uso da metodologia proposta na etapa de síntese passa forçosamente pela disponibilidade de ferramentas computacionais que a implementem. A ferramenta Grail é um ambiente de computação simbólico para máquinas de estado finitas, expressões regulares e linguagens finitas. A ferramenta foi elaborada com a intenção de ser usada em ensino e pesquisa (RAYMOND; WOOD, 1996). Posteriormente, foram implementados algoritmos a esta ferramenta, de forma que esta também execute o procedimento de síntese de acordo com a TCS e a abordagem modular (CURY, 2001).

No Grail o formato de um autômato consiste em uma lista de instruções armazenada em um arquivo ASCII. A seguir é apresentado um problema e como resolvê-lo utilizando o Grail. Suponha um sistema constituído de duas máquinas e um buffer, de acordo com a Figura 3. Os eventos  $\Sigma_c = \{a_1, a_2\}$  indicam início de operação e depósito de peça no buffer

e os eventos  $\Sigma_u = \{b_1, b_2\}$  indicam fim de operação. As máquinas devem ser modeladas sem a possibilidade de quebra. Os autômatos que representam as máquinas M1 e M2 e os respectivos formatos no Grail são apresentados na Figura 4.

A restrição de coordenação, ou especificação, para este sistema consiste em evitar *overflow* e *underflow* no buffer. O autômato que modela esta restrição bem como a representação no Grail são apresentados na Figura 5.

A parte de modelagem do exemplo está concluída no momento em que cada modelo é armazenado em um arquivo. Pode-se então aplicar os filtros do Grail para encontrar o supervisor minimamente restritivo que atenda à especificação selecionada. A Tabela 1 apresenta alguns filtros do Grail.

Os passos a seguir mostram os filtros necessários a cada operação do procedimento de síntese do supervisor:

1. Construir o modelo do sistema, através da composição síncrona das máquinas M1 e M2:

```
fmsync M1 M2 > sistema_manufatura
```

Normalmente joga-se o resultado da função num outro arquivo (> sistema\_manufatura);

Figura 3: Ilustração de um sistema de manufatura.

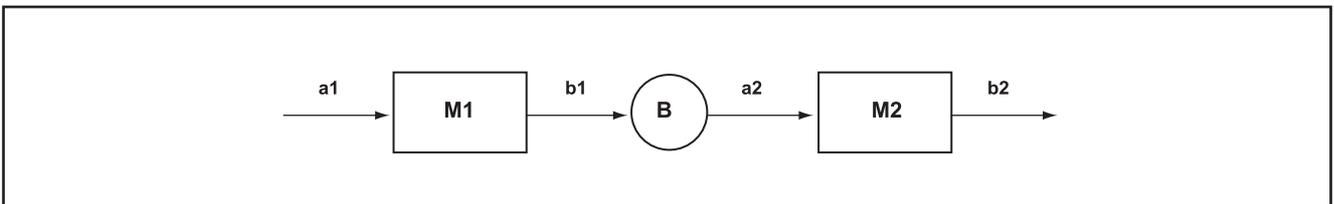
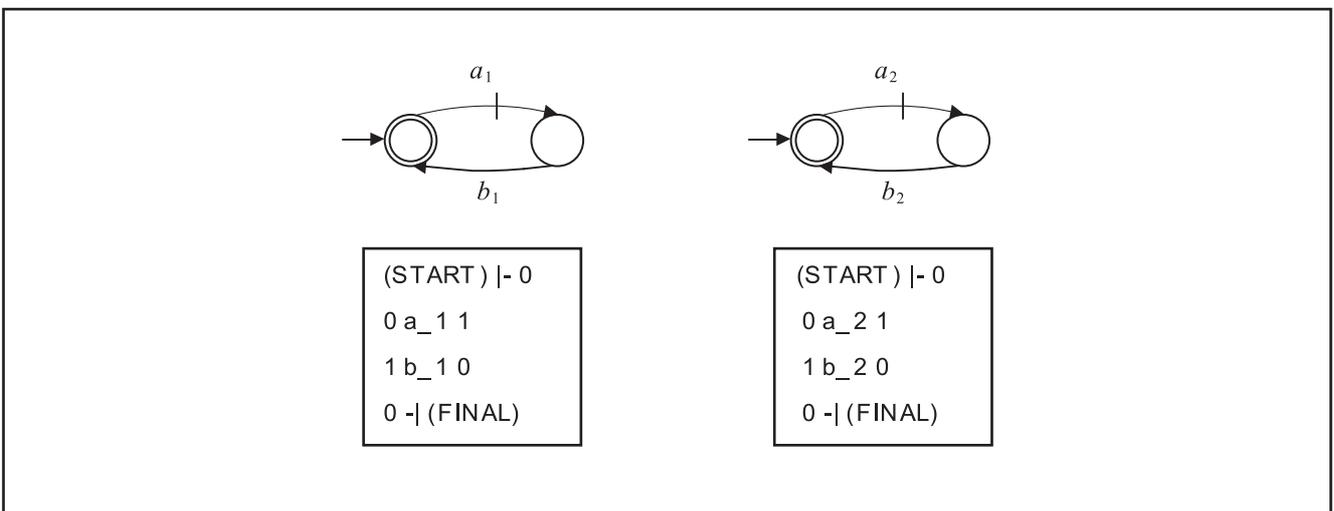


Figura 4: Representação de dois autômatos na ferramenta Grail.



2. Realizar a composição da planta com a especificação:

```
fmsync sistema_manufatura especificação > s
```

3. Encontrar a componente co-acessível de s:

```
fmtrim s > strim
```

4. Criar um arquivo de um único estado com self-loop dos eventos não controláveis (arquivo n-cont, por exemplo):

```
(START) |- 0
0 b_1 0
0 b_2 0
0 -| (FINAL)
```

5. Encontrar o supervisor minimamente restritivo:

```
fmsupc sistema_manufatura strim ncon > supervisor
```

6. Obter dados de controle do supervisor obtido:

```
fmcondat sistema_manufatura supervisor > dados_desabilitações
```

O resultado final, o supervisor, corresponde a um autômato que desabilita determinados eventos controláveis em seus estados. Dessa forma o Grail fornece como resultado a representação do autômato supervisor com a respectiva ação de controle em cada estado. A Figura 6 apresenta o resultado fornecido pelo Grail do exemplo tratado. De acordo com a figura 6, o supervisor é um autômato de dois estados em que os eventos a<sub>2</sub> e a<sub>1</sub> são desabilitados, respectivamente. A linha pontilhada indica essa desabilitação.

Em problemas de maior porte, diversos modelos de sub-sistemas e um conjunto de especificações devem resultar num conjunto de supervisores modulares, de acordo com a síntese modular. Assim, o processo de obtenção do supervisor deve ser realizado quantas vezes forem necessárias, de acordo com o número de especificações construídas.

Figura 5: Representação do autômato correspondente a especificação no Grail.

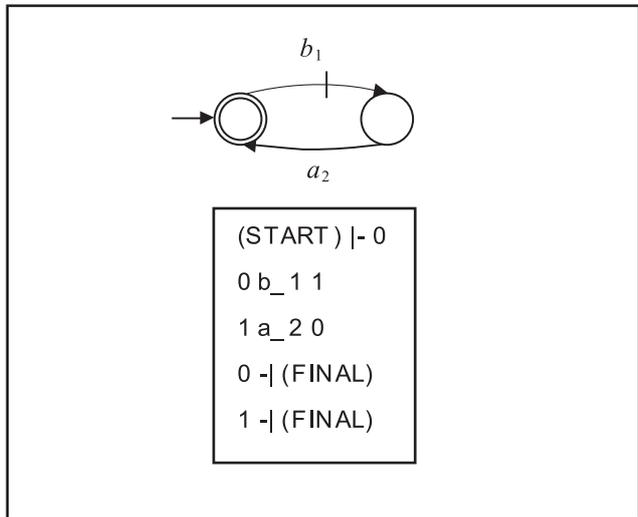


Figura 6: Representação do supervisor obtido no Grail.

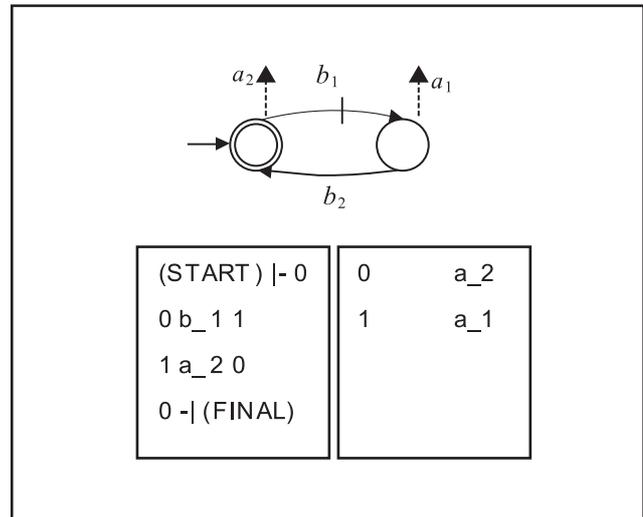


Tabela 1: Alguns filtros do Grail.

<b>fmsync</b>	Realiza a composição síncrona de dois autômatos
<b>fmtrim</b>	Encontra a componente trim de um autômato
<b>fmsupc</b>	Encontra a máxima linguagem controlável
<b>fmcondat</b>	Informa dados de controle de um autômato

### Etapa de Implementação

A estrutura de controle (supervisores e sistema produto) obtida na etapa de síntese é implementada, inicialmente, em um ambiente de simulação. Progressivamente, tanto componentes reais de tecnologia (dispositivos de controle e de comunicação), como seqüências operacionais reais (conectadas aos subsistemas) são agregados a este ambiente de simulação. Esta etapa compreende três fases, conforme ilustra a Figura 7:

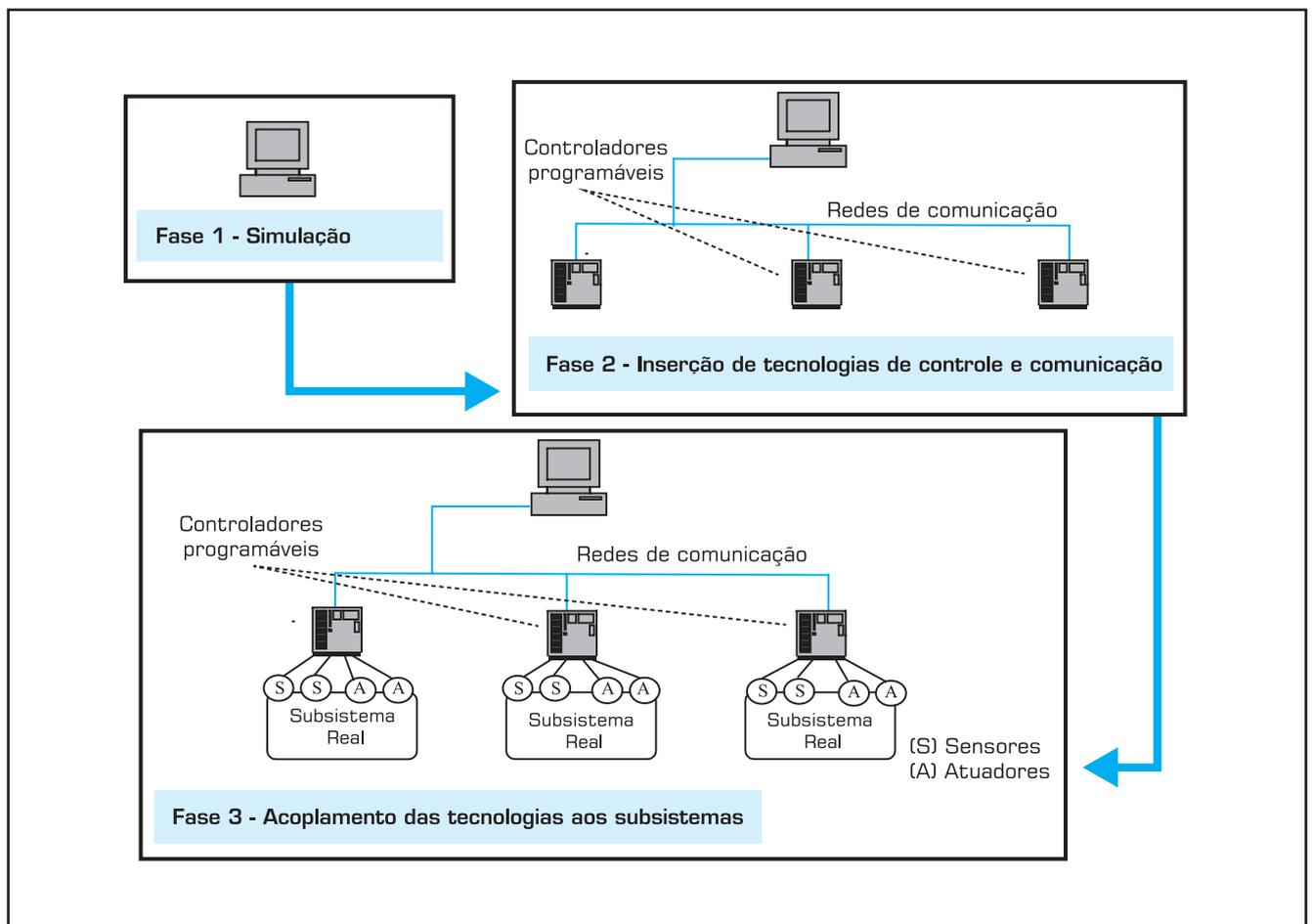
- i) Os supervisores modulares e o sistema produto são traduzidos em código válido para plataformas de simulação (por exemplo, ferramentas de simulação, tecnologias de supervisão e controle sendo utilizadas como simuladores, entre outros) e as seqüências operacionais são implementadas através de rotinas temporizadas;
- ii) As seqüências operacionais (SO) são progressivamente implementadas em outros dispositivos através da inserção de tecnologias de controle e comunicação;

- iii) O(s) dispositivo(s) que implementa(m) as seqüências operacionais são progressivamente acoplados aos respectivos subsistemas.

Na primeira fase da etapa de implementação, faz-se a simulação dos três níveis (supervisores modulares, sistema produto e seqüências operacionais) da estrutura de controle. A simulação da estrutura de controle é útil tanto para validar os modelos construídos (subsistemas e especificações) como para detectar modificações e inclusões necessárias. O projetista pode iniciar o funcionamento do nível dos supervisores modulares e sistema produto e acompanhar a evolução de estados e ações de controle associadas.

Na segunda fase, tecnologias de controle e comunicação são inseridas ao ambiente de simulação com a finalidade de testar e validar a topologia física de controle distribuído das seqüências operacionais (nível inferior da estrutura de controle). As tecnologias de controle simulam o conjunto de seqüências operacionais, sendo que o projetista pode

Figura 7: Fases da etapa de implementação



associar a estas, rotinas de temporização e de interface com botoeiras e sinalizadores. Por exemplo, conhecendo-se o tempo de execução de atividade de um determinado subsistema, pode-se simular a seqüência operacional através de um código de temporização.

Na terceira fase, os dispositivos que implementam as seqüências operacionais são progressivamente acoplados aos respectivos sensores e atuadores dos subsistemas reais. O projetista pode então substituir gradativamente um subconjunto de SO simuladas por SO reais até a implementação completa da estrutura de controle. Nesta fase também ocorre a tradução do conjunto de supervisores modulares e o sistema produto (conjunto de modelos dos subsistemas) em linguagem própria de programação (IEC 61131-3, 1998) de controladores programáveis.

## APLICAÇÃO EXPERIMENTAL DA METODOLOGIA

A metodologia proposta no presente trabalho é utilizada no projeto e reprojeto de integração e automação de um protótipo de sistema de manufatura (sistema real). Inicialmente, o projeto é desenvolvido considerando-se a produção de uma única família de produtos (aplicação inicial). Em seguida, em virtude de uma nova necessidade de mercado (nova aplicação), o reprojeto é desenvolvido considerando-se a inserção de uma nova família de produtos no sistema real já existente.

Esta seção apresenta a descrição do sistema real (protótipo), das aplicações demandadas e das etapas do ciclo de desenvolvimento – modelagem, síntese e implementação – para o projeto e reprojeto de integração e automação.

### Descrição do sistema real (protótipo do sistema de manufatura)

O protótipo do sistema, apresentado na Figura 8, executa operações típicas na manufatura: processos de fabricação, transporte, medição, armazenamento e classificação. O sistema tem por objetivo principal classificar, processar e armazenar os produtos de acordo com determinados atributos. O sistema é composto de seis subsistemas: fornecimento de material (G1), classificação e medição (G2), transporte (G3), processamento 1 (G4), processamento 2 (G5) e armazenamento (G6).

O subsistema G1 tem por objetivo armazenar e suprir matéria-prima sem classificação ao subsistema G2. O subsistema G2 realiza duas atividades sobre a matéria-prima: classifica-a quanto ao tipo (cor e material) e realiza a medição da altura. A atividade de medição se dá em função de uma eventual não uniformidade da matéria-prima, acarretando a existência de diferentes classes de tolerância dimensional. Faz-se necessário um dispositivo de medição da dimensão, sendo o próprio subsistema G2 capaz de descartar materiais

que não se enquadram na tolerância desejada. A classificação das peças é realizada através da combinação de sinais de sensores. Por exemplo, a peça metálica é identificada através da leitura dos sensores indutivo, capacitivo e óptico, enquanto a peça plástica vermelha, através dos sensores óptico e capacitivo. Após realizadas estas operações, a matéria-prima segue para o subsistema G3.

O subsistema G3 realiza o transporte entre os subsistemas G2, G4, G5 e G6. Os subsistemas G4 e G5 realizam processos de fabricação específicos. Por fim, o subsistema G6 armazena o produto final de acordo com os atributos obtidos no subsistema G2 (medição e classificação) e com o roteiro de produção realizado.

### Descrição das aplicações demandadas

Existem três tipos de matéria-prima: material MPA (metálico), material MPB (plástico de cor preta) e material MPC (plástico de cor vermelha). Considera-se inicialmente que a matéria-prima está armazenada num único magazine sem classificação.

A aplicação inicial demanda a produção de três tipos de produto final: PFA1, PFB1 e PFC1, em função das três matérias-primas MPA, MPB e MPC e do roteiro de produção R1. A nova aplicação, em função de uma necessidade de mercado, demanda a produção de outros três tipos de produto final: PFA2, PFB2 e PFC2, em função das três matérias-primas MPA, MPB e MPC e do roteiro de produção R2. A quantidade e o tipo de produto final a ser produzido são definidos pelo usuário de acordo com a demanda externa. A Tabela 1 apresenta o resumo dos produtos finais.

O roteiro de produção R1 utiliza os processos de fabricação dos subsistemas G4 e G5 e o R2 utiliza apenas o processo de fabricação do subsistema G4. A Figura 9 ilustra o protótipo de sistema de manufatura com os roteiros de produção R1 e R2.

### Etapas de modelagem

De acordo com a metodologia proposta, a primeira etapa (modelagem) consiste em representar em autômatos os subsistemas que compõem o sistema real e o conjunto de especificações (aplicação). Para tanto, utiliza-se a biblioteca de modelos proposta no presente trabalho. Para a modelagem dos subsistemas pode ser selecionado o modelo de dois estados (estado inativo e estado ativo), apresentado na Figura 10.

Para a modelagem das especificações analisa-se a configuração física do sistema, os roteiros de produção e as restrições de coordenação, de forma que o fluxo de matéria entre os subsistemas seja representado corretamente. Na aplicação inicial, para o correto fluxo de matéria entre os subsistemas G1 e G2 pode ser selecionada uma especificação de exclusão mútua entre os mesmos. Esta especificação impõe que G1

e G2 não podem funcionar simultaneamente e, ao mesmo tempo, define a seqüência de execução das atividades relacionadas a estes subsistemas. Após a realização da classificação e medição pelo subsistema G2, a matéria-prima segue para o subsistema de transporte G3. Este é definido como um transportador síncrono (mesa giratória) (GROOVER, 2001). Tal subsistema, movendo um produto de uma posição a outra, acarreta a movimentação dos demais produtos para as posições subseqüentes. Neste momento, o projetista seleciona modelos considerando as seguintes informações: transporte síncrono de quatro posições; primeira posição para chegada de peças, segunda e terceira posições para processos e quarta posição para retirada de peças. O conjunto de especificações relacionado ao transportador síncrono impõe

o correto fluxo de matéria entre os subsistemas G2, G4, G5 e G6. Os modelos das nove especificações necessárias ( $E_a, E_b, E_{c1}, E_{c2}, E_{c3}, E_{c4}, E_{d1}, E_{d2}, E_{d3}$ ) para a aplicação inicial são apresentados na figura 11.

Na nova aplicação, mantém-se a especificação de exclusão mútua entre os subsistemas G1 e G2. Após a realização da classificação e medição pelo subsistema G2, a matéria-prima segue para o subsistema de transporte G3. Nesta nova aplicação, o projetista seleciona modelos considerando as seguintes informações: transporte síncrono de quatro posições; primeira posição para chegada de peças, segunda posição para processo, terceira posição sem operação e quarta posição para retirada de peças. O conjunto de especificações relacionado ao transportador síncrono impõe o correto fluxo

Figura 8: Protótipo de sistema de manufatura.

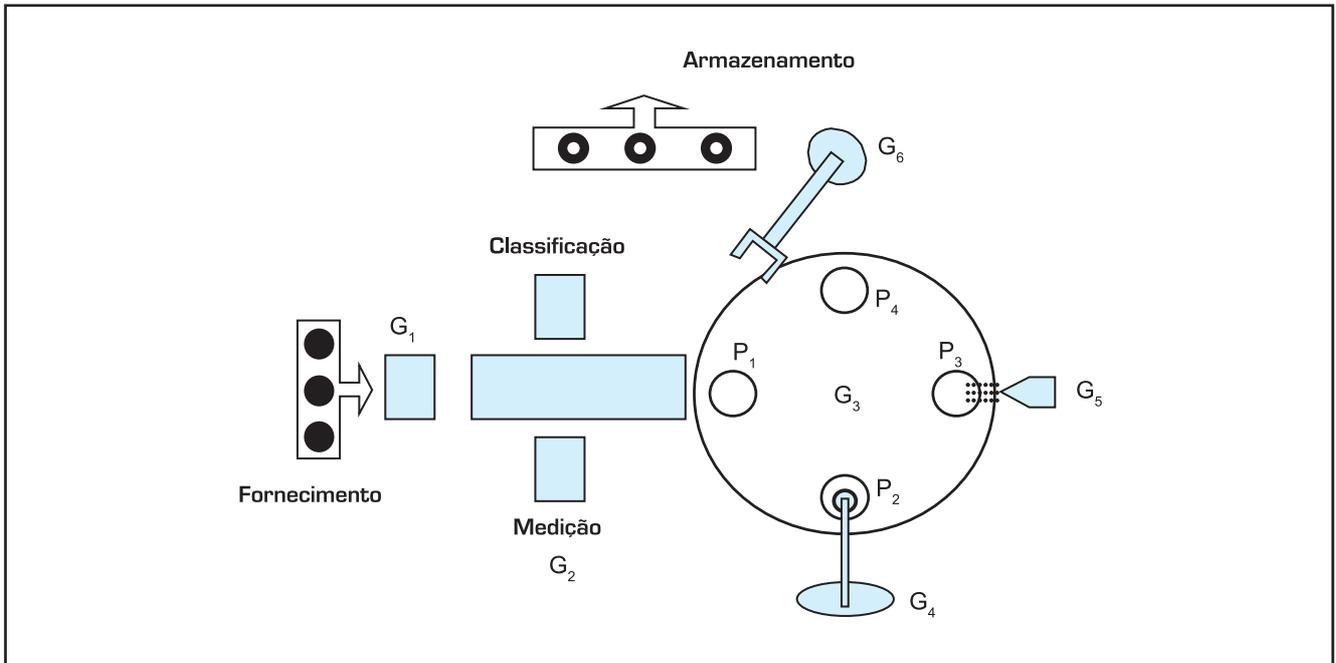


Tabela 1: Descrição da matéria-prima e roteiros de produção associados.

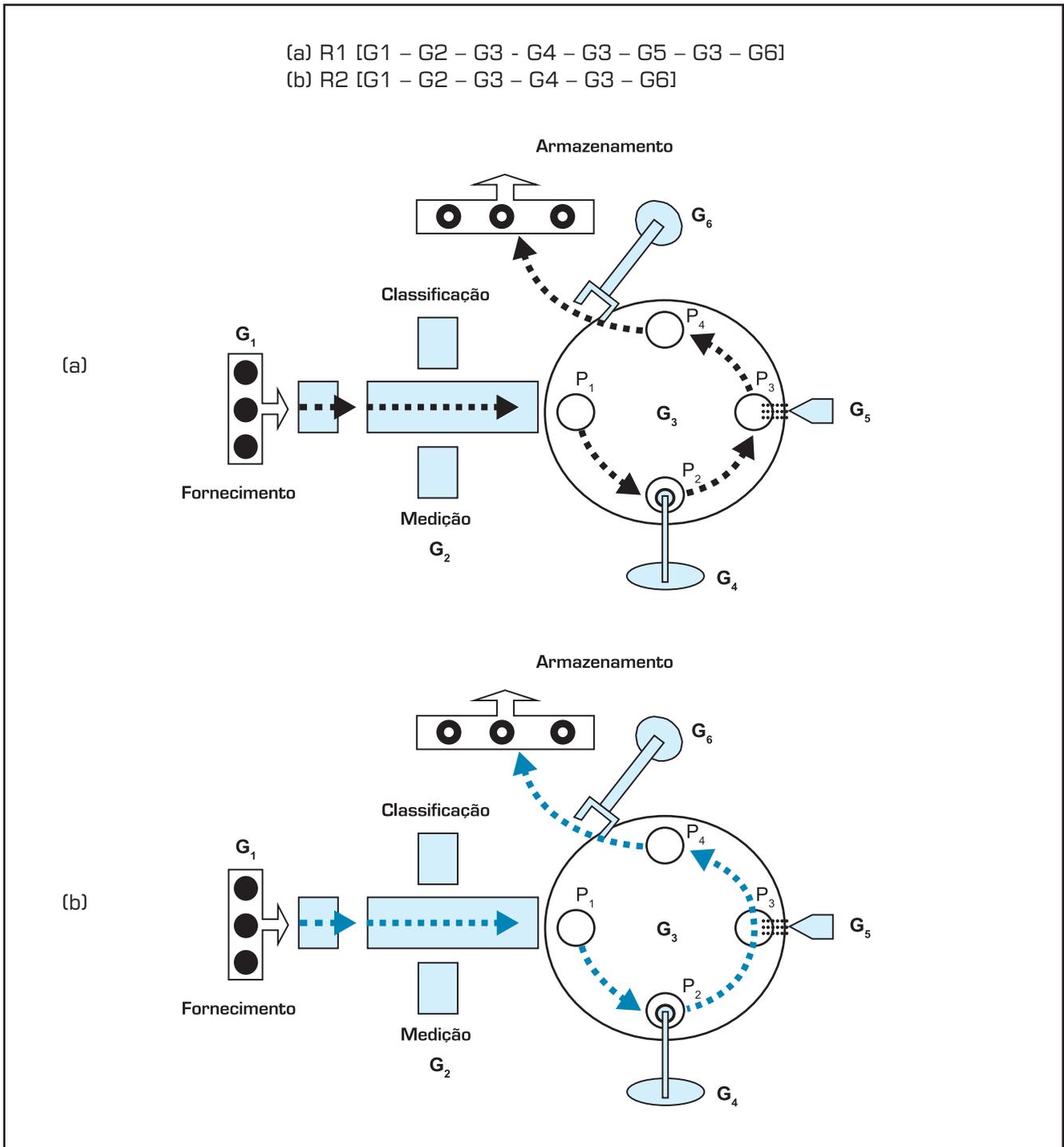
PRODUTO FINAL	MATÉRIA-PRIMA	ROTEIRO DE PRODUÇÃO
PFA1	MPA	R1
PFA2	MPA	R2
PFB1	MPB	R1
PFB2	MPB	R2
PFC1	MPC	R1
PFC2	MPC	R2

de matéria entre os subsistemas G2, G4 e G6. Os modelos das nove especificações necessárias ( $E_a, E_b, E_{c1}, E_{c2}, E_{c3}, E_{c4}, E_{d1}, E_{d2}, E_{d3}$ ) para a nova aplicação são apresentados na Figura 12.

**Etapa de síntese dos supervisores modulares**

A próxima etapa – síntese – corresponde à aplicação da TCS e da abordagem modular local. Assim, para cada especificação selecionada (aplicação inicial e nova aplicação),

**Figura 9: Roteiros de produção.**



um supervisor é obtido utilizando a ferramenta GRAIL. A ação conjunta dos supervisores obtidos restringe o comportamento dos subsistemas (que compõem o sistema real) às respectivas aplicações.

Para a aplicação inicial nove supervisores são sintetizados, cada um com uma finalidade específica, a partir das especificações selecionadas e apresentadas na Figura 11. Por exemplo, considerando a especificação  $E_{d1}$ , a planta local é obtida através da composição síncrona dos autômatos correspondentes aos subsistemas que compartilham

eventos com esta especificação ( $G_{loc,d1} = G2 \parallel G3 \parallel G4$ ). A especificação local é obtida através da composição da especificação genérica  $E_{d1}$  com a correspondente planta local ( $E_{loc,d1} = E_{d1} \parallel G_{loc,d1}$ ). Pode-se então calcular a máxima linguagem controlável contida na especificação, que é  $SupC(E_{loc,d1}, G_{loc,d1})$ . Em seguida, através de um algoritmo de minimização de supervisores (VAZ; WONHAM, 1986), obtém-se um supervisor com um menor número de estados e com a mesma ação de controle. A Tabela 2 apresenta os supervisores calculados a partir dos modelos dos subsistemas e das especificações selecionados para a aplicação inicial.

Por exemplo, a Figura 13 apresenta o supervisor  $SupC(G_{loc,d1}, E_{loc,d1})$ , resultante da especificação  $E_{d1}$ . A linha tracejada indica a ação de controle do supervisor, que é desabilitar eventos controláveis dos subsistemas G2, G3 e G4. Por exemplo, no estado 1 o supervisor desabilita a ocorrência do evento  $\alpha_3$ .

Para a nova aplicação, também nove supervisores são sintetizados, cada um com uma finalidade específica,

Figura 10: Modelo dos subsistemas

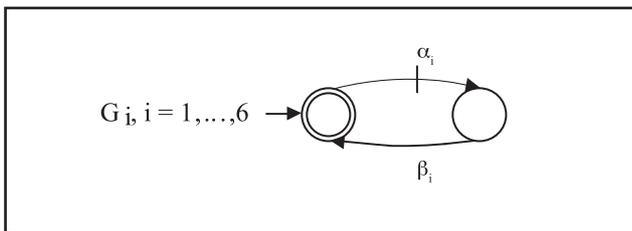
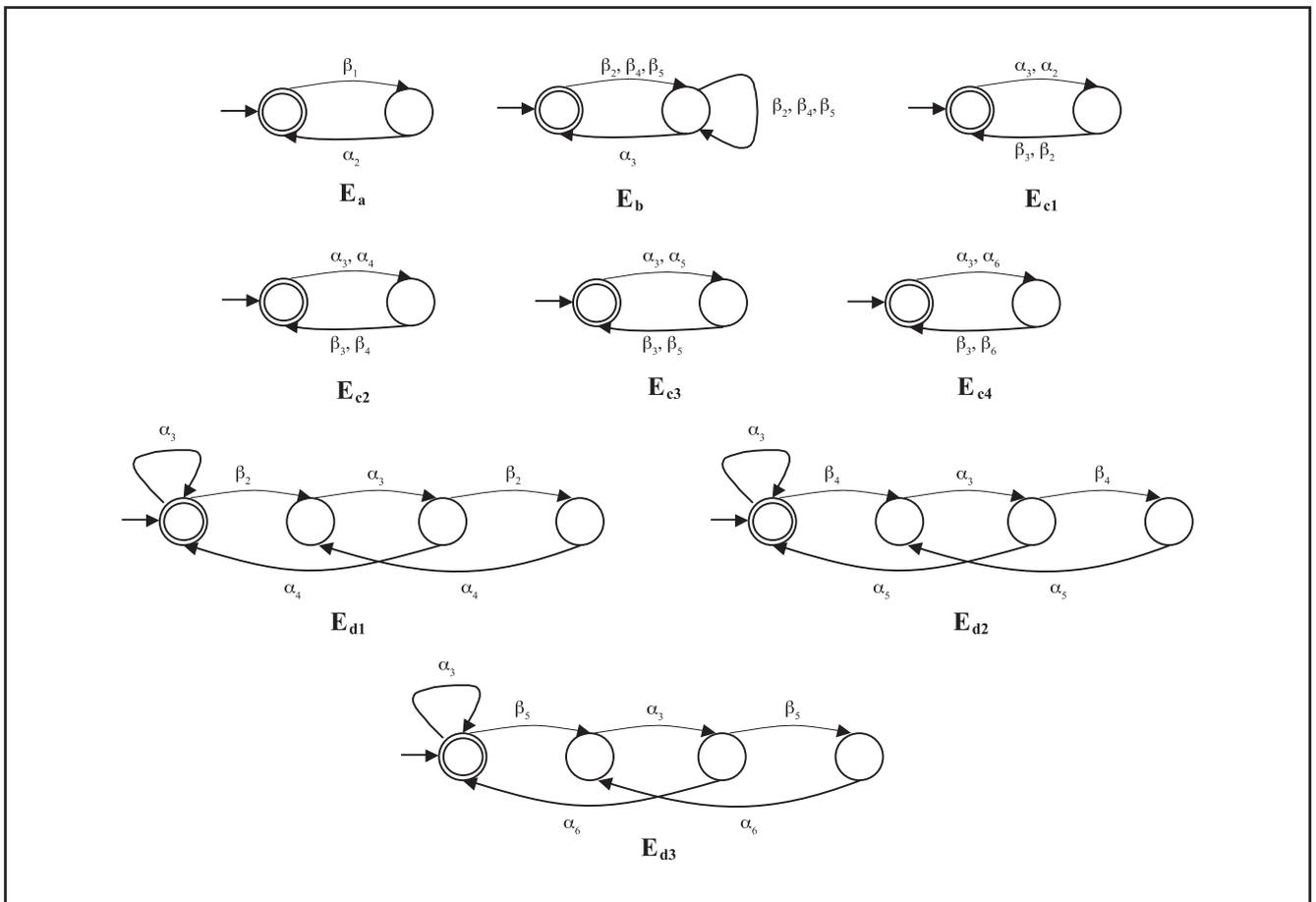


Figura 11 – Modelos das especificações para a aplicação inicial.



a partir das especificações selecionadas e apresentadas na Figura 12. Da mesma forma que na aplicação inicial, sintetizam-se nove supervisores modulares para a nova aplicação.

**Etapa de implementação**

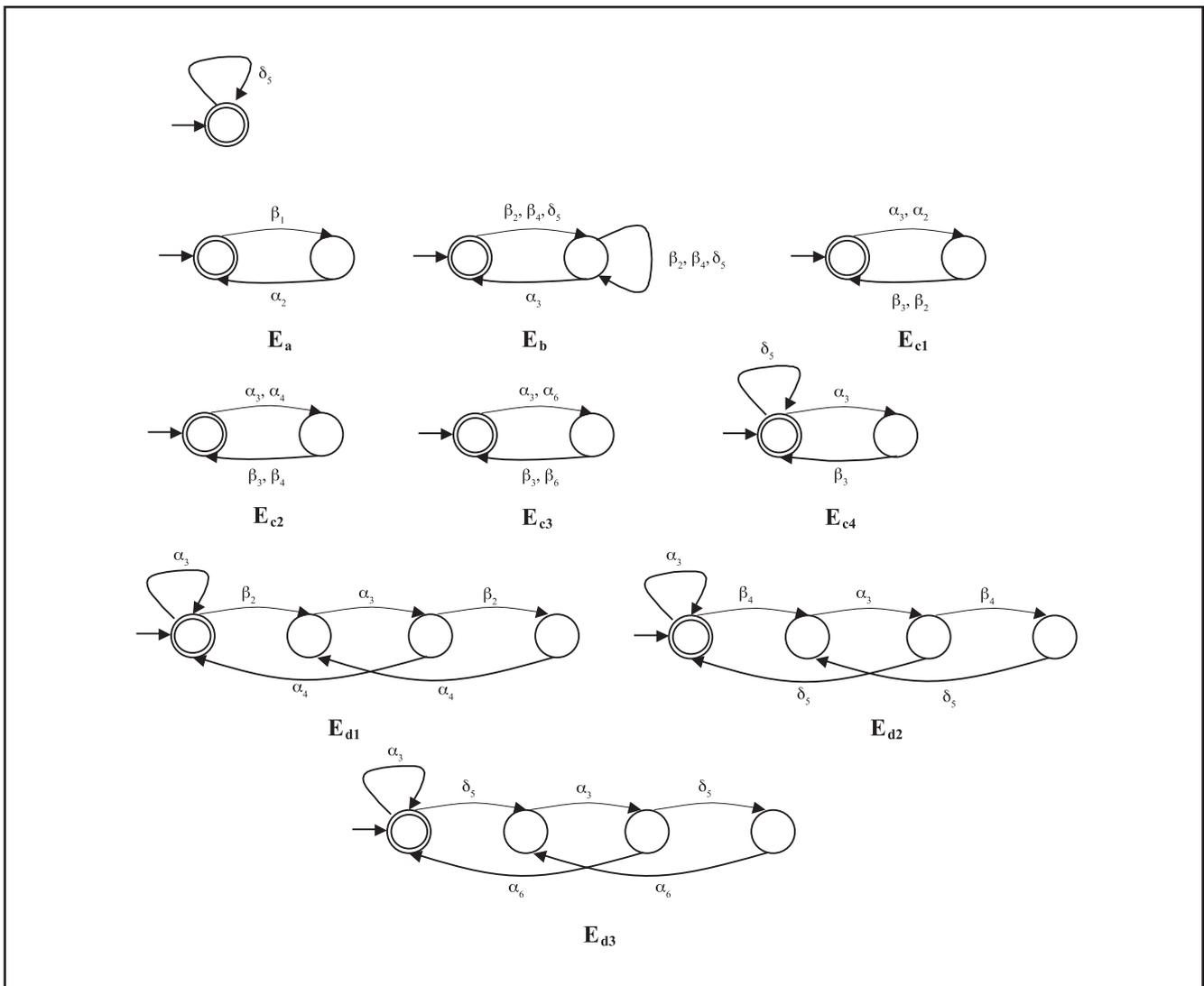
Em uma primeira fase da etapa de implementação, faz-se a simulação dos três níveis (supervisores modulares, sistema produto e seqüências operacionais) da estrutura de controle. Para tal fim, foi utilizada a ferramenta LUCAS (LUCAS – Flexible Workcell Controller, 1999), que permite a implementação destes níveis, a criação de interface gráfica e a simulação. Através da interface gráfica, o projetista pode visualizar, comandar e modificar a evolução de estados e ações de controle no nível dos supervisores

modulares e sistema produto.

Em uma segunda fase, tecnologias de controle e comunicação são inseridas no ambiente de simulação, tais como controladores programáveis (CLPs), redes industriais, interface homem-máquina (IHM), softwares supervisórios, entre outros. Nas aplicações tratadas, utilizou-se uma rede de comunicação industrial PROFIBUS FMS (PROFIBUS, 2005) entre o LUCAS e seis controladores programáveis. Os CLPs simulam o conjunto de seqüências operacionais através de rotinas de temporização e de interface com botoeiras e sinalizadores.

Em uma terceira fase, progressivamente os seis controladores programáveis são conectados aos sensores e atuadores dos seis subsistemas  $G_i$  ( $i = 1, \dots, 6$ ), conforme ilustra a Figura 14.

**Figura 12: Modelos das especificações para a nova aplicação.**



## CONCLUSÃO E TRABALHOS FUTUROS

O trabalho apresentou uma metodologia para o projeto de sistemas automatizados e integrados de manufatura. A inserção da TCS baseada na teoria de Linguagens e Autômatos, de ferramentas computacionais para a síntese e simulação, de biblioteca de modelos de subsistemas e especificações trouxe uma diminuição do tempo de desenvolvimento do sistema automatizado de manufatura.

A TCS é uma abordagem formal que permite a síntese automática de supervisores ótimos. Aliada a TCS, a abordagem modular traz uma maior agilidade ao projeto, uma vez que gera uma estrutura distribuída do sistema de controle. Essa característica permite uma maior flexibilidade ao projetista, uma vez que modificações e ajustes são mais simples de serem realizados. A biblioteca de modelos, além de facilitar e sistematizar a etapa de modelagem, permite a reutilização de modelos em projetos subsequentes. A integração das etapas de simulação e implementação permitiu uma maior confiabilidade na validação, otimização e realização da estrutura de controle.

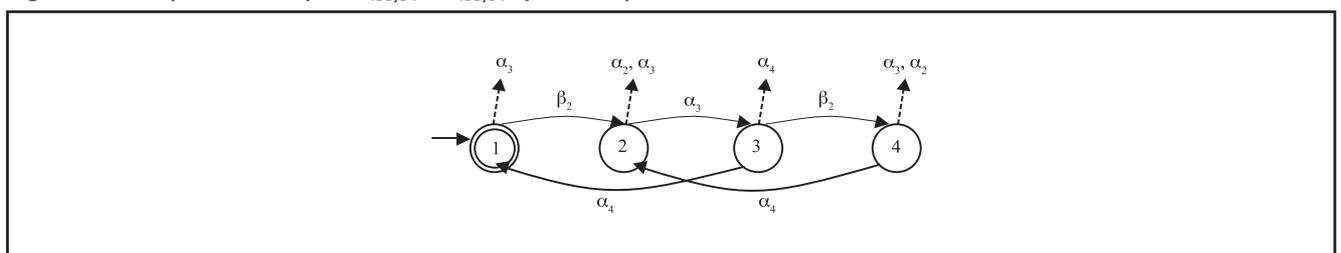
A metodologia proposta no presente trabalho aborda de maneira inovadora o problema de implementação de estruturas de controle em sistemas automatizados de manufatura em que modificações são corriqueiramente necessárias. A flexibilidade exigida atualmente na indústria de manufatura faz com que muitas empresas descartem a automação como uma solução viável. Nesse sentido, a utilização de modelos formais nas etapas da metodologia proposta induz a um projeto confiável, em que modificações podem ser feitas de forma sistemática. Essa característica evita os problemas oriundos de um projeto e implementação intuitivos, baseados na experiência e inspiração de projetistas da área.

A simulação da estrutura de controle e a implementação progressiva dos dispositivos de controle caracterizam-se também como ferramentas importantes para a consolidação da metodologia. Essas ferramentas permitem uma diminuição no tempo de implantação global do sistema automatizado e integrado de manufatura, uma vez que se evita o *start-up* de uma planta com todo o sistema de software e hardware instalado. Também, utilizando-se uma implementação progressiva, a equipe de manutenção pode detectar e

Tabela 2: Modelos utilizados no procedimento de síntese e supervisores locais resultantes (aplicação inicial).

ESPECIFICAÇÃO GENÉRICA	PLANTA LOCAL	ESPECIFICAÇÃO LOCAL	MÁXIMA LINGUAGEM CONTROLÁVEL (SUPERVISOR MODULAR)
$E_a$	$G_{loc,a} = G1 \parallel G2$	$E_{loc,a} = G_{loc,a} \parallel E_a$	$SupC(G_{loc,a}, E_{loc,a})$
$E_b$	$G_{loc,b} = G2 \parallel G3 \parallel G4 \parallel G5$	$E_{loc,b} = G_{loc,b} \parallel E_b$	$SupC(G_{loc,b}, E_{loc,b})$
$E_{c1}$	$G_{loc,c1} = G2 \parallel G3$	$E_{loc,c1} = G_{loc,c1} \parallel E_{c1}$	$SupC(G_{loc,c1}, E_{loc,c1})$
$E_{c2}$	$G_{loc,c2} = G3 \parallel G4$	$E_{loc,c2} = G_{loc,c2} \parallel E_{c2}$	$SupC(G_{loc,c2}, E_{loc,c2})$
$E_{c3}$	$G_{loc,c3} = G3 \parallel G5$	$E_{loc,c3} = G_{loc,c3} \parallel E_{c3}$	$SupC(G_{loc,c3}, E_{loc,c3})$
$E_{c4}$	$G_{loc,c4} = G3 \parallel G6$	$E_{loc,c4} = G_{loc,c4} \parallel E_{c4}$	$SupC(G_{loc,c4}, E_{loc,c4})$
$E_{d1}$	$G_{loc,d1} = G2 \parallel G3 \parallel G4$	$E_{loc,d1} = G_{loc,d1} \parallel E_{d1}$	$SupC(G_{loc,d1}, E_{loc,d1})$
$E_{d2}$	$G_{loc,d2} = G3 \parallel G4 \parallel G5$	$E_{loc,d2} = G_{loc,d2} \parallel E_{d2}$	$SupC(G_{loc,d2}, E_{loc,d2})$
$E_{d3}$	$G_{loc,d3} = G3 \parallel G5 \parallel G6$	$E_{loc,d3} = G_{loc,d3} \parallel E_{d3}$	$SupC(G_{loc,d3}, E_{loc,d3})$

Figura 13: Supervisor  $SupC(G_{loc,c1}, E_{loc,c1})$  para a aplicação inicial.



corrigir eventuais erros antes e após a realização completa de software e hardware. Esse aspecto naturalmente acarreta uma maior confiabilidade ao projeto global e permite uma maior agilidade à equipe de manutenção.

Como resultado dessa abordagem metodológica têm-se os elementos necessários para a documentação detalhada do projeto técnico. Para tal fim, existem ainda outros fatores a serem considerados, tais como manutenção, relações comerciais, custos, atualizações e revisões técnicas de equipamentos já instalados, dentre outros.

Embora a metodologia possa contribuir de forma significativa no projeto de sistemas automatizados e integrados de manufatura, alguns aspectos trazem dificuldades na sua aplicação. Inicialmente, pode-se citar a modelagem de subsistemas e especificações utilizando-se autômatos. Essa tarefa requer certa experiência tanto na modelagem através desse formalismo quanto certa experiência no funcionamento de sistemas automatizados de manufatura. Assim, apesar da vantagem da TCS residir na síntese automática de supervisores, a construção de modelos para o sistema real e especificações poderá depender da experiência e inspiração do projetista, comprometendo a confiabilidade, o tempo necessário e o custo global do desenvolvimento.

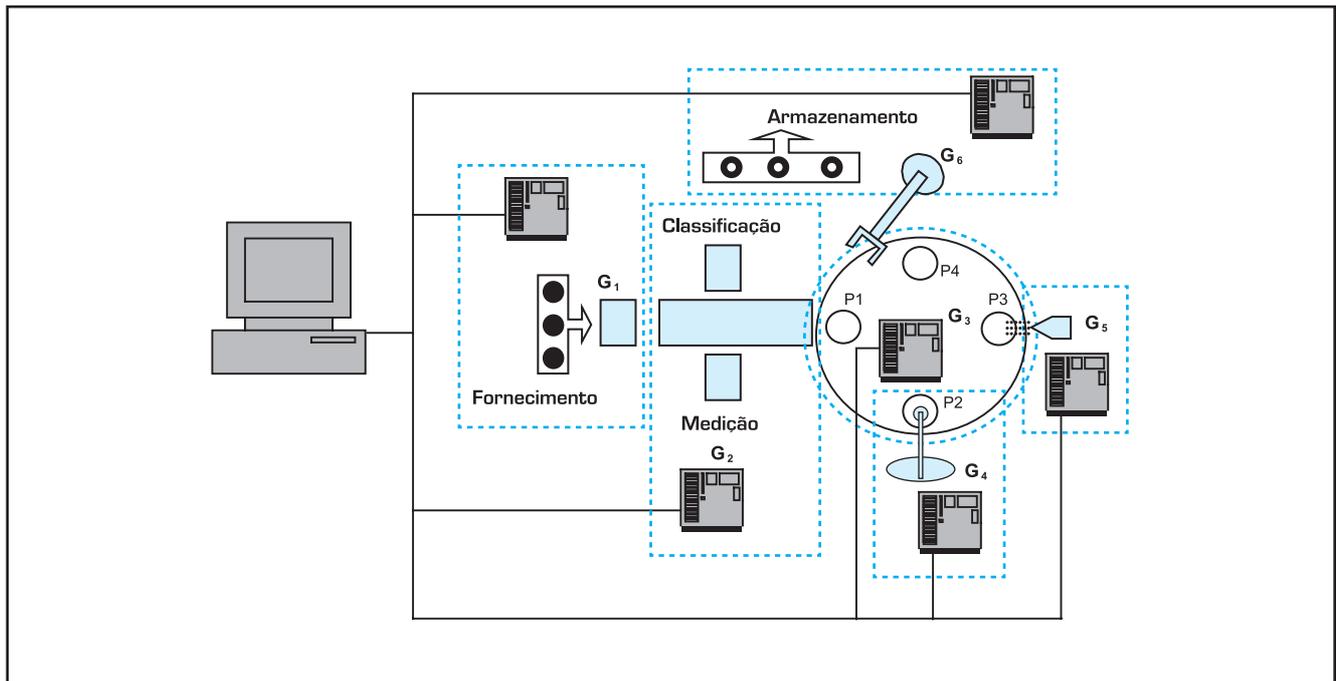
Outro aspecto diz respeito ao procedimento de síntese de supervisores da TCS. A viabilidade do uso da metodologia de síntese passa forçosamente pela disponibilidade de ferramentas computacionais que a implementem. Algumas destas estão hoje disponíveis, porém, em geral, são desenvolvidas

no meio acadêmico, e não têm as boas características de um produto, no que diz respeito a suas interfaces e capacidade de lidar com problemas de porte. O desenvolvimento de ferramentas computacionais “comerciais” é de fundamental importância para a disseminação e aplicação da metodologia aqui apresentada.

Outra dificuldade na aplicação da metodologia relaciona-se à implementação progressiva da estrutura de controle. Como essa técnica utiliza-se de tecnologias de comunicação e controle, muitos problemas de integração e automação surgem no decorrer da implementação. A principal causa de tais problemas é a adoção de pacotes proprietários (protocolos de comunicação, redes industriais, softwares, dentre outros) por parte dos fabricantes existentes no mercado. Tradicionalmente, cada fabricante de equipamentos, de dispositivos ou de *software* adota um determinado modelo ou pacote. A dificuldade está justamente em lidar sistemas tecnologicamente heterogêneos, uma vez que é necessário estabelecer progressivamente a comunicação com os diversos dispositivos que controlam a planta.

A metodologia proposta apresenta ainda algumas limitações que pressupõem a continuidade do trabalho desenvolvido. No início do ciclo, devem ser aprofundados os critérios de segmentação do sistema real em subsistemas, bem como a identificação de um conjunto de especificações para uma dada aplicação. As bibliotecas de modelos de subsistemas e de especificações devem ser constantemente atualizadas. A experiência de cada novo projeto deve ser

Figura 14: Estrutura física do sistema de controle após a etapa de implementação.



utilizada na atualização das bibliotecas. Entretanto, a seleção dos modelos ainda é muito dependente da experiência do projetista. Devem ser pesquisados métodos para sistematizar a seleção de modelos da biblioteca a partir do sistema real e da aplicação.

A biblioteca das Tecnologias Avançadas de Manufatura –

AMTs deve também ser constantemente revisada e atualizada. Uma abordagem sistemática deve ser pesquisada para auxiliar tanto no reconhecimento dos subsistemas a partir das tecnologias, quanto na definição das tecnologias de controle e comunicação necessárias a implementação e realização da estrutura de controle modular.

Artigo recebido em 12/07/2005

Aprovado para publicação em 27/09/2007

## Referências

- AKESSON, K. *Methods and Tools in Supervisory Control Theory*. PhD Thesis, Chalmers University of Technology, Sweden, 2002.
- BALEMI, S.; HOFFMANN, G. J.; GYUGYI, P. *et al.* Supervisory control of a rapid thermal multiprocessor. *IEEE Transactions on Automatic Control*, v. 38, n. 7, p. 1040-1059, 1993.
- CARROL, J.; LONG, D. *Theory of Finite Automata*. Prentice-Hall International Editions, 1989.
- CASSANDRAS, C. G.; LAFORTUNE S. *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, 1999.
- CHANDRA, V.; ORUGANTI, B.; KUMAR, R. *UKDES: A graphical software tool for the design, analysis & control of discrete event systems*, clue.eng.iastate.edu/~rkumar/PUBS/ukdes.ps. 17:34, 2002.
- CURY, J. E. R. *Teoria de Controle Supervisório de Sistemas a Eventos Discretos*. V Simpósio Brasileiro de Automação Inteligente, Gramado, 2001.
- ERBE, H.-H. Low Cost Intelligent Automation in Manufacturing. *Proceedings of the 15th Triennial IFAC World Congress*, Barcelona, Spain, 2002.
- FABIAN, M.; HELLGREN A. *Descos – a Tool for Education and Control of Discrete Event Systems*. Kluwer Academic Publishers, 2000.
- GOUVÊA DA COSTA, S.; PLATTS, K.; FLEURY, A. Advanced Manufacturing Technology: defining the object and positioning it as an element of manufacturing strategy. *Proceedings of VI International Conference on Industrial Engineering and Operations Management*, São Paulo, 2000.
- GROOVER, M. *Automation, Production Systems, and Computer-Integrated Manufacturing*, 2. ed. New Jersey: Prentice-Hall, 2001.
- HOLLOWAY, L. E.; KROGH, B. H.; GIUA, A. A survey of Petri nets methods for controlled discrete event systems, *Discrete Event Dynamic Systems: Theory and Applications*, 7, p. 151-190. Boston: Kluwer Academic Publishers, 1997.
- IEC 61131-3. International Electrotechnical Commission, *Programmable Controllers*. Programming Languages, 1998.
- IEC 848. International Electrotechnical Commission. *Preparation of function charts for control systems*, Switzerland, 1988.
- LUCAS. Distributed Work Cell Controller and Teleservice. User Manual, Dortmund, 1999.
- MOORE, P. R.; PU, J.; NG, H.C. Virtual engineering: an integrated approach to agile manufacturing machinery design and control. *Mechatronics*, 13, p. 1105-1121, 2003.
- MORAES, C. C.; CASTRUCCI, P. L. *Engenharia de Automação Industrial*. Livros Técnicos e Científicos (LTC), 2001.
- PROFIBUS. Disponível em: <<http://www.profibus.org/>>. Acesso em: 01 jul. 2005.
- QUEIROZ, M. H.; CURY, J. E. R. Controle Supervisório Modular de Sistemas de Manufatura. *Revista Controle & Automação*, SBA, 13, p. 115-125, 2002a.
- QUEIROZ, M. H.; CURY, J. E. R. Synthesis and implementation of local modular supervisory control for a manufacturing cell, *Discrete Event Systems: Analysis and Control*, Kluwer Academic Publishers, p. 103-110, 2002b.
- RAMADGE, P.J.; WONHAM, W. M. The control of discrete event systems, *Proceedings of IEEE, Special Issue on Discrete Event Dynamic Systems*, 77, p. 81-98, 1989.
- RAYMOND, D.; WOOD, D. G. Engineering automata in C++: Version 2.5 <<http://www.csd.uwo.ca/research/grail/>>. Verificada em 31 de janeiro de 2003.
- SANTOS, E. A. P. *Contribuições ao projeto conceitual de sistemas de manipulação e montagem automatizados*. Tese de Doutorado (Engenharia Elétrica), Universidade Federal de Santa Catarina, 2003.
- VAZ, A. F.; WONHAM, W. M. On supervisor reduction in discrete-event systems. *International Journal of Control*, v. 44, n. 2, p. 475-491, 1996.
- WONHAM, W. M. *Notes on control of discrete event systems*. Course notes for ECE 1636F/1637S. Dept. of Electrical and Computer Engineering, University of Toronto, Canada, 1999.

## Sobre os autores

### Marco Antonio Busetti de Paula

Programa de Pós-Graduação em Engenharia de Produção e Sistemas – PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
End.: Rua Imaculada Conceição, 1155 – Prado Velho – Curitiba – PR – 80215 901  
Tel: +55 (41) 3271-1344, Fax: +55 (41) 3271-1345  
Email: marco.busetti@pucpr.br

### Eduardo Alves Portela Santos

Programa de Pós-Graduação em Engenharia de Produção e Sistemas – PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
End.: Rua Imaculada Conceição, 1155 – Prado Velho – Curitiba – PR – 80215 901  
Tel: +55 (41) 3271-1344, Fax: +55 (41) 3271-1345  
Email: eduardo.portela@pucpr.br