



**Universidade de São Paulo**

**Biblioteca Digital da Produção Intelectual - BDPI**

---

Departamento de Mecatrônica e Sistemas Mecânicos - EP/PMR    Comunicações em Eventos - EP/PMR

---

2014-08-20

# Método para projeto de sistema de controle supervisorio considerando flexibilidade de reconfiguração

---

Congresso Brasileiro de Automática - CBA, 20, 2014, Belo Horizonte

<http://www.producao.usp.br/handle/BDPI/50135>

*Downloaded from: Biblioteca Digital da Produção Intelectual - BDPI, Universidade de São Paulo*

## MÉTODO PARA PROJETO DE SISTEMA DE CONTROLE SUPERVISÓRIO CONSIDERANDO FLEXIBILIDADE DE RECONFIGURAÇÃO

ROBSON M. DA SILVA<sup>†\*</sup>, FABRÍCIO JUNQUEIRA<sup>†</sup>, DIOLINO J. SANTOS FILHO<sup>†</sup>, LUCAS A. MOSCATO<sup>†</sup>,  
PAULO E. MIYAGI<sup>†</sup>

*\*Universidade Estadual de Santa Cruz  
Rodovia Jorge Amado, Km 16  
Ilhéus, Bahia, Brasil*

*†Escola Politécnica da Universidade de São Paulo  
Av. Prof. Luciano Gualberto, trav.3, n.380  
São Paulo, São Paulo, Brasil*

Emails: rmsilva@uesc.br, fabri@usp.br, diolinos@usp.br, lamoscat@usp.br,  
pemiyagi@usp.br

**Resumo**— Os atuais sistemas de controle supervísório da produção necessitam ter uma estrutura distribuída implementando diferentes domínios com características de inteligência, agilidade, e tolerância a falhas, a fim de considerar a flexibilidade de reconfiguração. A técnica de sistema de controle holônico (SCH) é adequada para integrar as tecnologias heterogêneas desses domínios, pois propõe o desenvolvimento de entidades de controle com autonomia, reatividade, proatividade, cooperação, interface social, recursos de aprendizagem e, além disso, disponibiliza plataformas de sistema multi-agente para facilitar a implementação. No entanto, existe necessidade de métodos formais para garantir que as especificações de projeto sejam efetivamente implementadas. Portanto, este trabalho apresenta um método para projeto deste supervísório combinando as técnicas de SCH com rede de Petri. Um estudo de caso de sistema de manufatura é considerado para mostrar a viabilidade da proposta e exemplificar a análise dos resultados alcançados.

**Palavras-chave**— sistema produtivo; reconfiguração; tolerância a falhas; sistema de controle holônico; rede de Petri.

### 1 Introdução

A solução de controle de um sistema produtivo (SP) integra *hardware* e *software* dos seus subsistemas, os quais estão organizados em arquiteturas e especificações de heterogêneas tecnologias (Strasser e Froschauer, 2012). De fato, existem diversas soluções de engenharia que facilitam o desenvolvimento de aplicações dedicadas ao controle dos subsistemas de um SP. Entretanto, a integração destas soluções não é trivial, pois deve organizar diferentes visões/ abordagens dos vários domínios existentes em um tempo relativamente curto devido à concorrência entre empresas.

A arquitetura de controle de um SP especifica a integração das tecnologias envolvidas, definindo suas interfaces, inter-relações e restrições. Em geral, a arquitetura de controle é dividida em vários níveis de uma estrutura hierárquica, onde o nível de controle supervísório envolve a coordenação de grupos de máquinas e equipamentos de modo sincronizado e integrado com os demais níveis (Groover, 2011). Em relação ao tipo de controle envolvido, um SP pode ser classificado como um sistema a eventos discretos (SED) se os parâmetros e as variáveis dos processos produtivos são discretos e modificados em momentos também discretos no tempo (David e Alla, 1994). O sistema de manufatura (SM) é uma classe representativa de SP classificado como um SED que transforma a matéria-prima para obtenção de um produto final com certo valor de mercado.

Novos paradigmas têm sido propostos de acordo com as possíveis variações do que se espera de um SP. Mehrabi et al. (2000) relacionam paradigmas de SMS: (i) SM em massa com redução de custos de produto, (ii) SM enxuta com elevação da qualidade do produto, (iii) SM flexível com aumento da variedade de produtos, e (iv) SM reconfigurável com o aumento da capacidade de resposta e adaptação com a necessidade de mudanças.

Por outro lado, baseado nos aspectos que caracterizam os sistemas feitos pelo homem e sua propriedade intrínseca de ocorrência de falha, sistemas tolerantes a falhas têm sido propostos para assegurar que os processos de um SP não sofram interrupções. Zhang e Jiang (2008) propõem um sistema modular de controle que envolve a detecção da falha, o estudo dos seus efeitos, identificação das causas e, além disso, possui um módulo de reconfiguração do sistema que é feita realocando funções e escolhendo caminhos alternativos de interação entre os processos.

Neste contexto, consideramos que SPs que consideram flexibilidade de reconfiguração, ou simplesmente sistemas produtivos reconfiguráveis (SPRs), são aqueles com a capacidade de mudança dinâmica de produção de itens ou serviços para atender novas demandas com agilidade, flexibilidade e aumento de eficiência. Novas demandas podem ser um novo produto, quantidade de produtos, integração de novas tecnologias/ processos e, além disso, pode envolver situações causadas

por ocorrência de falhas.

A técnica de sistema de controle holônico (SCH) (Vrba et al., 2011) propõe o desenvolvimento de sistemas de controle integrado de tecnologias heterogêneas. Um SCH explora os conceitos de superposição de holons e agentes, tais como autonomia, reatividade, proatividade, cooperação, interface social, recursos de aprendizagem, e tira proveito das vantagens características da implementação de holons por meio de plataformas de sistema multi-agente. Desse modo, um SCH permite adicionar, remover ou modificar a estruturas de software e hardware do sistema de controle, abrangendo as características requeridas para o sistema de controle supervisorio de SPR. Entretanto, segundo Hsieh e Lin (2013), existe um longo caminho para disseminação e divulgação desta tecnologia pois, isso depende do desenvolvimento de novos métodos e ferramentas que facilitem a migração das abordagens tradicionais para novas soluções.

Portanto, neste trabalho é proposto um método de desenvolvimento de um sistema de controle supervisorio considerando flexibilidade de reconfiguração por meio da composição dinâmica e de modelagem adaptável dos subsistemas de um SP. O método proposto contribui para inovação tecnológica no controle de SED, combinando as técnicas de SCH com abordagens *bottom-up* e *top-down* de modelagem de sistemas através do uso das extensões da rede de Petri (RdP): a RdP estendida (RdPE) (David e Alla, 1994) e o esquema de fluxo de produção - *production flow schema* (PFS) (Hasegawa et al., 1999). Esta proposta expande trabalhos anteriores, em Silva et al. (2012), o foco foi o desenvolvimento de mecanismos de reconfiguração, combinando SCH com técnicas de controle tolerante a falhas. Em da Silva et al. (2014), nós exploramos técnicas de arquitetura orientada a serviços (Erl, 2005) para propor uma nova arquitetura de controle de SP.

Na Seção 2 uma síntese de conceitos fundamentais é descrita. Na Seção 3 é apresentado o método proposto. Na Seção 4 um estudo de caso é descrito destacando as vantagens da proposta. Na Seção 5 têm-se as principais conclusões do presente estudo.

## 2 Rede de Petri, holons e agentes

A RdP é uma técnica formal de modelagem que tem a vantagem de simulação e visualização de SED em diferentes níveis de abstração e de permitir comunicação entre pessoas relacionadas ao projeto (Zurawski e Zhou, 1994; Hasegawa et al., 1999; Hsieh e Lin, 2013).

O *production flow schema* (PFS) (Hasegawa et al., 1999) é uma extensão de RdP canal-agência interpretada para modelar conceitualmente os SPs. O PFS permite uma descrição estrutural hi-

erárquica de um SP num nível relativamente alto de abstração. Um outro modelo em RdP é gerado para descrever o comportamento dinâmico e detalhado do sistema.

A RdP estendida (RdPE) (David e Alla, 1994) é uma 8-tupla  $(P, T, F, I, H, W, t, M)$  onde:  $P = P_1, P_2, P_3, \dots, P_m$  é um conjunto finito ( $m \in \mathbb{N}^*$ ) de lugares<sup>1</sup> da rede;  $T = T_1, T_2, T_3, \dots, T_m$  é um conjunto finito ( $m \in \mathbb{N}^*$ ) de transições da rede;  $F = PXT \cup TXP$  é o conjunto de arcos orientados;  $I = PXT$  é o conjunto de arcos inibidores;  $H = PXT$  é o conjunto de arcos habilitadores;  $W : F, I, H \rightarrow \mathbb{N}^*$  é uma função peso associados aos arcos, sendo que a omissão de peso associado aos arco indica peso unitário;  $t = tm_1, tm_2, tm_3, \dots, tm_n$  é o conjunto finito ( $n \in \mathbb{N}^*$ ) de temporizações associadas às respectivas transições, em que a omissão de valor indica uma temporização nula;  $M = M_0, M_1, M_2, \dots, M_n$  é a marcação da rede, onde  $M = m_1, m_2, \dots, m_i$  ( $i \in \mathbb{N}^*$ ) é definida pelo vetor destas marcações contendo o número de marcas  $m_i$  no lugar  $P_i$  e  $M_0$  é a marcação inicial.

Existem várias ferramentas para edição e simulação dos modelos de RdP, tais como o *software* PIPE2 (Bonet et al., 2007) que possui recursos gráficos e módulos para análise das propriedades de RdPE.

Um resumo de arquiteturas e aplicações em SCH é apresentado em Leitão et al. (2013). PROSA (Van Brussel et al., 1998), ADACOR (Leitão et al., 2013), ANEMONA (Botti e Boggino, 2008) e Rockwell (Vrba et al., 2011) são alguns exemplos de arquitetura de controle supervisorio baseado em conceitos de SCH.

Na Fig. 1, tem-se um exemplo de aplicação do conceito de holons que podem representar desde todo o SM a seus dispositivos de controle, tais como sensores e atuadores. Um holon pode formar holons maiores ou serem decompostos até os holons de nível mais inferior (atômico). Os holons interagem com holons do mesmo nível e com holons de níveis superiores ou inferiores. Para implementar um SCH utilizam-se especificações da *foundation for intelligent physical agents* (FIPA) e plataformas como JADE (Bellifemine et al., 2007), pois um sistema multi-agente requer a implementação de certas funcionalidades específicas, tais como transporte de mensagens, codificação, análise sintática, e ontologias. Exemplos de padrões de ontologia para SM são propostos por ANSI/ISA 88 (ANSI/ISA, 1995) e Béhé et al. (2014).

## 3 Método para projeto de AHCS

Analisando-se os estudos acima citados, os desafios identificados são: (i) o desenvolvimento de mecanismos para a composição de holarquias e

<sup>1</sup>Termos associados a RdP são sublinhados.

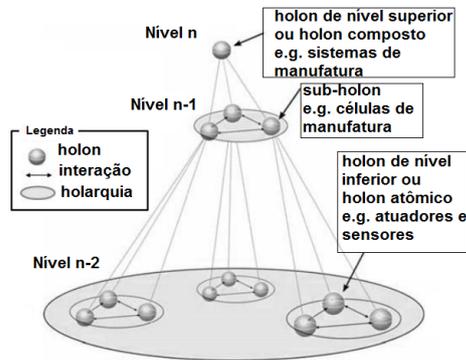


Figura 1: Recursividade de holons aplicado a SM.

a execução de processos produtivos considerando restrições temporais e a capacidade de processamento; (ii) o desenvolvimento de mecanismos de colaboração baseados na funcionalidade do equipamento para executar uma ordem de produção, (iii) a adequação de protocolos e sistematização de implementação e/ ou atualização dos projetos; (iv) a definição da arquitetura de controle e supervisão; e (v) a especificação de estratégias de controle e supervisão.

Esta seção apresenta o método de projeto de um sistema de controle supervisor de SPR, denominado de *active holonic control system* (AHCS), que considera soluções para esses desafios. Conforme ilustrado no PFS da Fig. 2, o método está dividido em fases e subfases. A fase de integração garante um ciclo de retroalimentação para gerenciamento, atualização e otimização.

**Fase 1 - análise de requisitos:** são definidas especificações do sistema, tais como: objetivo, objeto de controle, dispositivos de controle, divisão dos subsistemas, características estruturais da planta física sob controle, *layout*, recursos, definição das tarefas, estratégias e funções de controle, descrição da integração entre as partes do sistema, e casos de reconfiguração. Esta fase é dividida em subfases.

**Subfase 1.1 — especificação da ontologia:** definem-se os planos de processos, produtos disponíveis e as estratégias de controle em situação normal e sob efeito de falhas. As especificações usam a ontologia proposta e, desse modo, tem-se um modelo semântico adequado para interação entre as entidades do AHCS.

**Subfase 1.2 – identificação dos holons:** em AHCS, um *holon* pode pertencer a diferentes holarquias tanto no mesmo nível de controle como em níveis diferentes. Holons de baixo nível de controle podem ser parte de holons de alto nível colaborando com outros holons. Desse modo, estas holarquias representam um *middleware* para interação entre os diferentes níveis de controle baseada em colaboração de funcionalidades. No nível de máquina e de dispositivo, os quais têm soluções proprietárias dos fabricantes, é possível tra-

tar a interoperabilidade, usando uma linguagem unificada tais como *extensible markup language* (XML). Seguindo as especificações da FIPA, nesta subfase, a estrutura e relações entre os holons é especificada de acordo com diagramas UML. Os *holons* de AHCS (Fig. 3) são:

- *holon* de produto – *product holon* (*PrH*), que contém o conhecimento necessário para escolha do plano de produção. *PrHs* estão associados às matérias-primas, produtos intermediários e produtos finais. A especificação do *PrH* envolve a definição das funções de controle de cada produto a ser processado e dos tempos e planos de execução das ordens. Desse modo, cada *PrH* tem um fluxo interno envolvendo uma ou mais estratégias de controle baseadas na colaboração de holons;
- *holon* de tarefas – *task holon* (*TH*), que contém o conhecimento para gerenciar a execução de cada ordem de trabalho que deve resultar num produto. Os *THs* especificam tanto as tarefas de controle de operações normais como sob ocorrência de falhas, bem como as medidas a serem tomadas em casos de falta de energia, erros de operação, etc.;
- *holon* supervisor – *supervisor holon* (*SuH*), que contém o conhecimento para a supervisão do produto/ serviço executado pelos holons, coordenando suas operações, registrando as habilidades de cada componente e fornecendo funções combinadas com outras entidades do sistema de controle. O *SuH* coordena os *holons* supervisionando as operações que estão sendo executadas no mesmo; e
- *holon* operacional – *operational holon* (*OpH*), que está associado a algum recurso físico da planta tais como, time de manutenção e operação, equipamentos e dispositivos de controle. O *OpH* estabelece o comportamento destes recursos de acordo com os objetivos e habilidades de cada um e segue uma lista planejada de operações (agenda) que o recurso tem de executar para entregar seu serviço. Analisando as dependências físicas entre os objetos de controle, é possível agregar um ou mais recursos em um único *holon*.

**Subfase 1.3 — especificação de mecanismos de reconfiguração:** mecanismos que permitem autonomia, tolerância a falhas e a reconfiguração do sistema são especificados. Os objetivos desta especificação são: (i) identificação das falhas, (ii) levantamento das especificações para o diagnóstico de falhas e (iii) levantamento das estratégias para tratamento de falhas e reconfiguração.

**Fase 2 - modelagem:** Nesta fase devem ser desenvolvidos os modelos dos holons em PFS e RdPE. Esta fase é realizada em subfases.

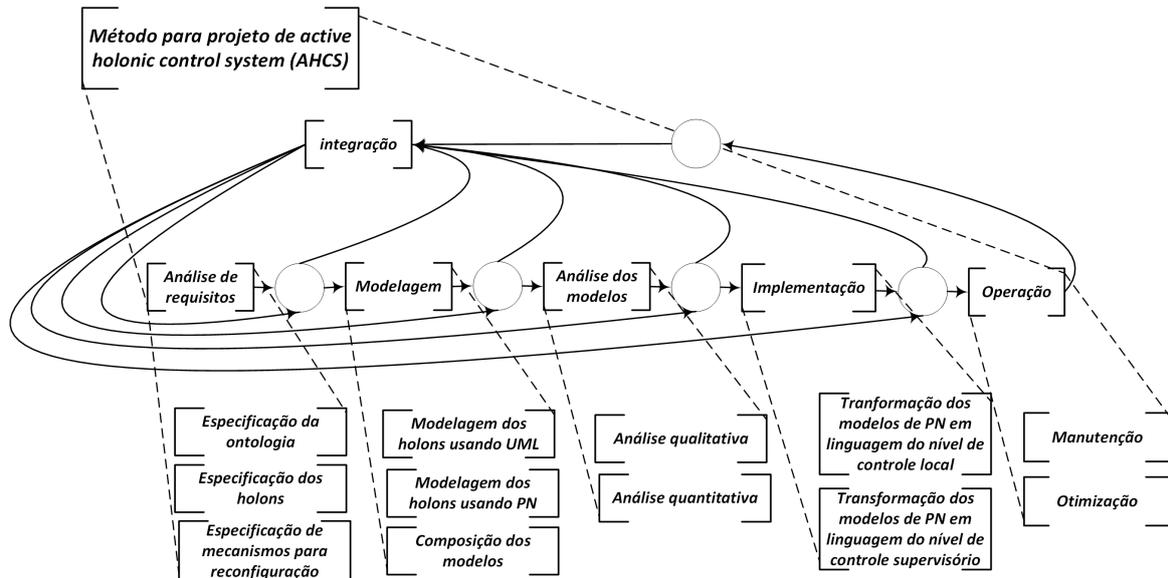


Figura 2: PFS que apresenta o enlace entre as fases do método de projeto de um sistema de controle supervisorio considerando flexibilidade de reconfiguração (*active holonic control system* - AHCS).

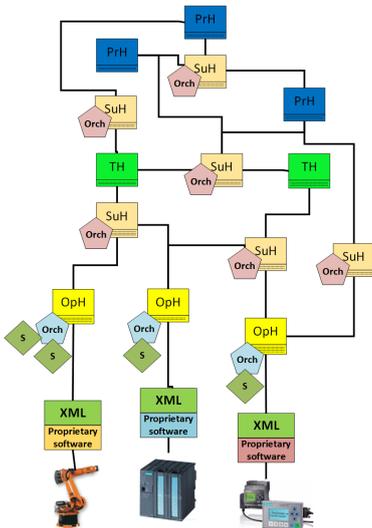


Figura 3: Arquitetura de controle de AHCS.

**Subfase 2.1 — modelagem dos holons:** nesta subfase o *workflow* dos holons é desenvolvido em PFS e, em seguida, refinado até o nível mais baixo de detalhamento utilizando a RdPE onde um lugar representa um estado do componente/ processo, enquanto uma transição representa uma operação que leva a uma mudança de estado.

A modelagem de *PrHs* envolve a modelagem do plano de produção necessário para obtenção dos produtos finais e o *workflow* de cada produto. Cada *workflow* executa uma ou mais operações gerando outros produtos que podem ser intermediários ou final. Cada *PrH* é representado inicialmente por uma atividade do PFS.

Os *THs* representam estratégias de controle e cada ordem de trabalho através de atividade,

fluxos primários e secundários que representam a interação entre elas.

A modelagem do *SuH* envolve atividades de registro de habilidades, registro da troca de mensagens, otimização de tarefas, composição de holarquias, supervisão da execução e das ações relacionadas ao tratamento de falha. As trocas de mensagens entre os holons são inicialmente representadas por distribuidores.

A modelagem de *OpHs* envolve o detalhamento do comportamento dinâmico dos objetos de controle. Para modelá-los com o requisito de tolerância à falhas a solução é considerar os estados (normal e sob falhas) dos dispositivos de atuação em conjunto com os estados (disponível ou indisponível) dos outros dispositivos de controle. Lugares representam estados do objeto e transições representam a mudança para um outro estado. Esta mudança é oriunda de um sinal de comando para um outro estado normal (sem falhas) ou de algum evento externo que causa um estado de falha.

**Subfase 2.2 — modelagem da reconfiguração:** neste estudo a reconfiguração é considerada tanto para melhorar o desempenho do sistema pelo aumento da qualidade ou quantidade da produção como em casos de ocorrência de falhas. Nesta fase são modelados os mecanismos para reconfiguração do sistema (vide (Silva et al., 2012)).

**Subfase 2.3 — composição dos modelos:** os modelos em RdPE são compostos representando o compartilhamento de recursos, a troca de mensagens, o envio de comandos para atuadores e recebimento de sinais de sensores. Esta composição é feita tanto com arcos habilitadores ou inibidores bem como através de lugares auxiliares, pois isto facilita a visualização do envio e

recebimento de sinais de controle, o que por sua vez facilita a tradução dos modelos para linguagens de programação. O projetista deve analisar no projeto a necessidade de definir novas interações, pois a composição dos modelos de holarquias envolve específicas interações entre os modelos dos holons.

**Fase 3 – análise/ simulação:** esta fase é desenvolvida com apoio de ferramentas de edição, análise estrutural e simulação de RdPE. Esta fase é subdividida em: análise qualitativa e quantitativa. A análise qualitativa permite a verificação das propriedades estruturais e comportamentais dos modelos em RdPE, para derivar conclusões sobre o funcionamento do sistema. Para realizar a análise quantitativa é necessária a introdução do parâmetro de tempo, para verificar se o desempenho é consistente com as especificações.

**Fase 4 - implementação:** para implementação, os modelos resultantes são interpretados como especificações de programas de controle a serem executados por controladores programáveis (nível de controle local) e computadores (nível de controle supervisorio). Essa fase também compreende a codificação, parametrização e desenvolvimento de interfaces específicas de cada equipamento. No nível de controle local considera-se um dos padrões de linguagem de IEC 61131, como o SFC, e para o nível de controle supervisorio, a linguagem JAVA com o uso da plataforma JADE.

**Fase 5 - operação:** nesta fase, a supervisão e controle é realizada sincronizando o funcionamento dos modelos em RdPE com os sinais de sensores do estado dos dispositivos, permitindo a geração de relatórios e gráficos de controle, a fim de supervisionar e controlar o sistema. A introdução ou remoção de novos componentes requer a adição ou remoção de novas marcas nos modelos correspondentes de RdPE. A interface gráfica do usuário de uma plataforma multi-agente pode ser utilizada para monitorar e controlar o estado dos agentes.

#### 4 Estudo de caso

O método descrito na seção anterior é aplicado em um SP do laboratório de sistemas de automação da USP (SP-LSA) (Fig. 4). Nesta Seção são destacados alguns dos modelos e resultados obtidos.

O objetivo do SP-LSA é a montagem de produtos compostos pelas peças: uma *base* (preta [bcb], vermelha [rcb] ou prata [acb]), um *pino* (preto [bp] ou prata [ap]), uma *mola* [s] e uma *tampa* [co]. Os pinos prata são montados nas bases pretas, enquanto as demais bases são montadas com pino preto. Molas e tampas são as mesmas em todas as montagens.

O SP-LSA é dividido em cinco subsistemas produtivos: *SP-EA* – estação de alimentação, *SP-EI* – estação de inspeção, *SP-ET* – estação de

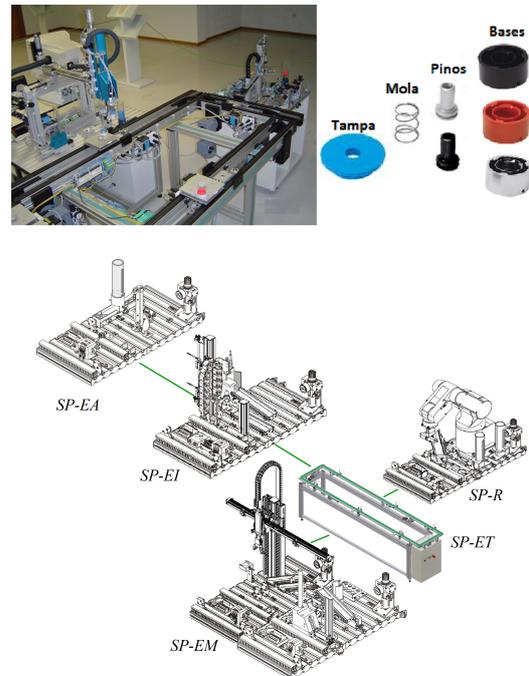


Figura 4: Sistema produtivo do LSA.

transporte, *SP-EM* – estação de montagem (formada por duas subestações) e *SP-R* – estação robô.

A Fig. 5 ilustra o detalhamento da subfase 1.1 onde é especificada a ontologia. Cada termo especifica um processo produtivo que é representado por um holon ou holarquia, por exemplo (por simplificação, aqui não são descritos todos os termos), **BusinessProcesses**<sup>2</sup> representam os processos de um SM de uma ordem de cliente (**CustomerOrder**), que contém uma ou mais ordens de produto (**ProductOrder**) para permitir a execução de certa quantidade (**ProductQuantity**) do mesmo tipo de produto (**Product**), os quais devem ser entregues nos seus devidos prazos (**Dates**).

Na Fig. 6 tem-se um exemplo de diagrama UML do SP-LSA, relacionando-o com o PFS e a RdPE das interações para a atividade [requer pallet in *OpH*] na [disponibilidade de *pallet*]. O *OpH-T* interage para execução de ordens de produção com *THs*, os quais representam estratégias de execução do produto.

A Fig. 7 ilustra o plano de produção no SP-LSA com as entradas e saídas necessárias e o inter-relacionamento entre os *PrHs*.

A identificação dos dispositivos é realizada de acordo com a especificação DIN/ISO 1219:1996-11 e usando os códigos recomendados na especificação IEC 61346-2:2000-12. Por exemplo, na nomenclatura “1S2”: 1= número do circuito, S=código do dispositivo e 2=número do dispo-

<sup>2</sup>Termos associados à ontologia estão destacados desta forma.

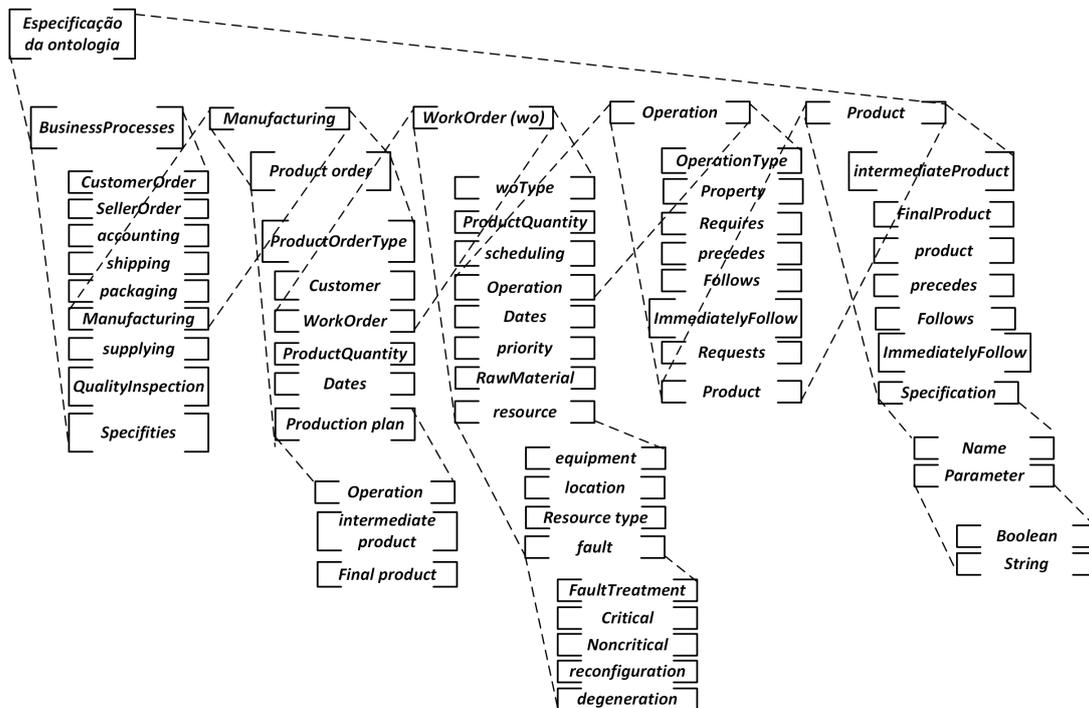


Figura 5: PFS do detalhamento da subfase 1.1 - especificação da ontologia.

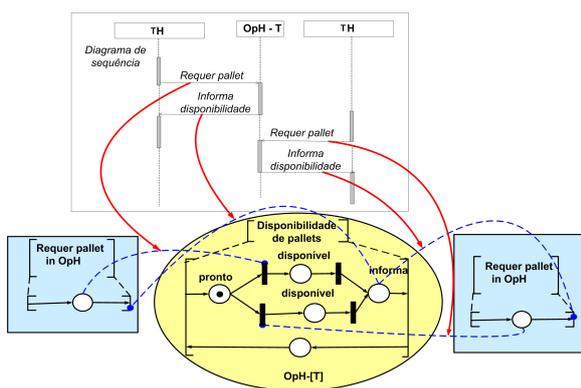


Figura 6: Exemplo de interações. A execução a partir do lugar “pronto” depende da agenda do *OpH* ou de uma prioridade pré-definida.

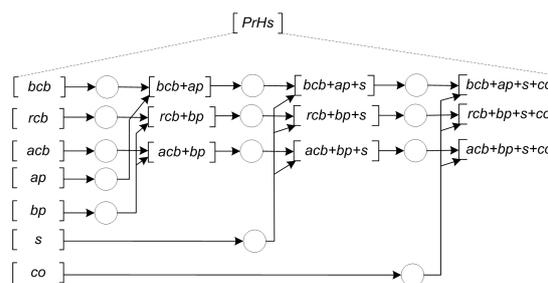


Figura 7: *PrHs* em SP-LSA.

sitivo. Na Fig. 8 tem-se o detalhamento da atividade [*identifica cor de peça*] do *PrH*-[*rcb*].

Na Fig. 9 tem-se o refinamento de alguns processos produtivos envolvidos, que são representados por *THs*.

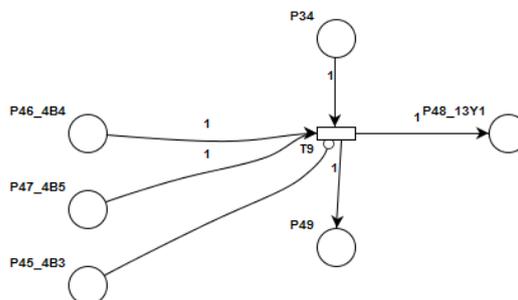
A Fig. 10 apresenta o modelo de um *OpH* do *SP-EA* do objeto de controle de válvula que ativa e desativa o gerador de vácuo e seu controlador. A disponibilidade da rede de comunicação é representada considerando se a transmissão de sinal está funcionando.

Na Fig. 11 tem-se um exemplo de monitoração da troca de mensagens entre os holons *TH*-[*customer order*] e *TH*-[*seller order*], representada pelo *SuH*-[*CO\_SO*].

A Fig. 12 ilustra um exemplo de composição de modelos onde tem-se a formação de holarquias.

*OpHs* são requisitados para acompanhamento/ liberação dos processo.

O mapeamento entre a RdPE e a linguagem JAVA é ilustrado na Fig. 13.



Legenda - B: sensor; Y: solenóide; 4B4: detecta peça; 4B5: detecta cor; 4B3: detecta se a peça é metálica; 13Y1: altera estado da válvula para carregar [rcb].

Figura 8: RdPE da atividade [*identifica cor de peça*] (*rcb*).

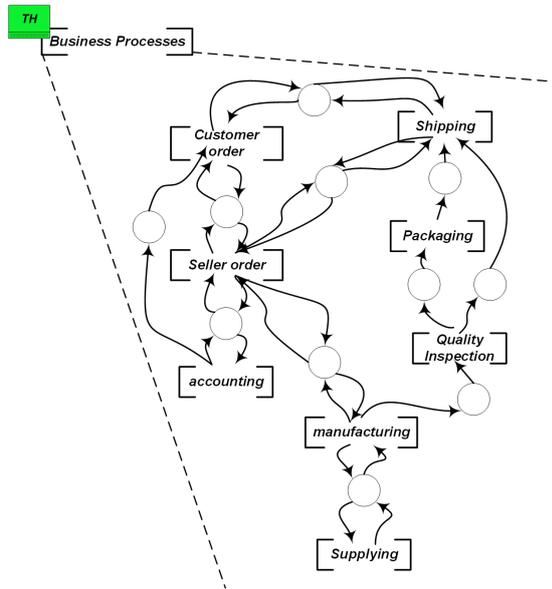


Figura 9: Detalhamento de [Business processes].

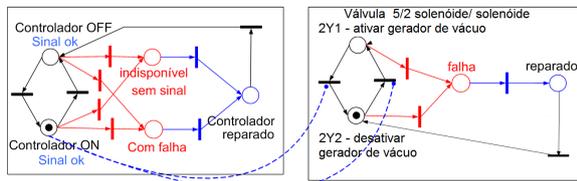


Figura 10: Modelo de objeto de controle.

## 5 Conclusões

O estudo apresenta a arquitetura de sistema de controle denominada *active holonic control system* (AHCS) e seu método de projeto, combinando a técnica de sistema de controle holônico (SCH) com as abordagens *bottom-up* e *top-down* usando uma interpretação e uma extensão da rede de Petri: *production flow schema* (PFS) e rede de Petri estendida (RdPE), respectivamente. Os modelos são utilizados tanto para estruturar o desenvolvimento das especificações dos componentes como para a apresentação do método proposto. Como exemplo de aplicação é apresentado um estudo de caso de sistema de manufatura em que é possível comprovar a efetividade das técnicas usadas em AHCS.

## Agradecimentos

Os autores agradecem ao parcial apoio financeiro das agências governamentais brasileiras: CNPq, CAPES, e FAPESP.

## Referências

ANSI/ISA (1995). *Batch Control Part 1: Models and Terminology*, American National Standard, ANSI/ISA-88.01.

Béhé, F., Galland, S., Gaud, N., Nicolle, C. e Koukam, A. (2014). An ontology-based metamodel for multiagent-based simulations, *Simulation Modelling Practice and Theory* **40**: 64–85.

Bellifemine, F. L., Caire, G. e Greenwood, D. (2007). *Developing multi-agent systems with JADE*, Vol. 7, Wiley.

Bonet, P., Lladó, C. M., Puijaner, R. e Knottenbelt, W. J. (2007). Pipe v2. 5: A petri net tool for performance modelling, *Proc. of CLEI 23rd Latin American Conference on Informatics*.

Botti, V. e Boggino, A. G. (2008). *ANEMONA: A multi-agent methodology for Holonic Manufacturing Systems*, Springer.

da Silva, R. M., Blos, M. F., Junqueira, F., Santos Filho, D. J. e Miyagi, P. E. (2014). A service-oriented and holonic control architecture to the reconfiguration of dispersed manufacturing systems, *Technological Innovation for Collective Awareness Systems, IFIP AICT*, Vol. 423, Springer, pp. 111–118.

David, R. e Alla, H. (1994). Petri nets for modeling of dynamic systems: a survey, *Automatica* **30**(2): 175–202.

Erl, T. (2005). *Service-oriented architecture: concepts, technology, and design*, Prentice Hall, New York.

Groover, M. P. (2011). *Automação industrial e sistemas de manufatura, 3.ed.*, Pearson Prentice Hall.

Hasegawa, K., Miyagi, P. E., Santos Filho, D. J., Takahashi, K., Ma, L. e Sugisawa, M. (1999). On resource arc for Petri net modelling of complex resource sharing system, *Journal of Intelligent and Robotic Systems* **26**(3-4): 423–437.

Hsieh, F.-S. e Lin, J.-B. (2013). A self-adaptation scheme for workflow management in multi-agent systems, *Journal of Intelligent Manufacturing* pp. 1–18.

Leitão, P., Marik, V. e Vrba, P. (2013). Past, present, and future of industrial agent applications, *Industrial Informatics, IEEE Transactions on* **9**(4): 2360–2372.

Mehrabi, M. G., Ulsoy, A. G. e Koren, Y. (2000). Reconfigurable manufacturing systems and their enabling technologies, *International Journal of Manufacturing Technology and Management* **1**(1): 114–131.

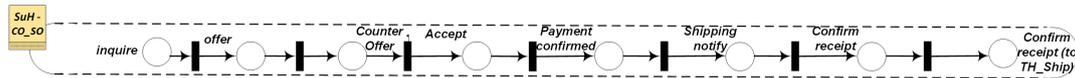


Figura 11: *SuH* monitorando a troca de mensagens entre holons.

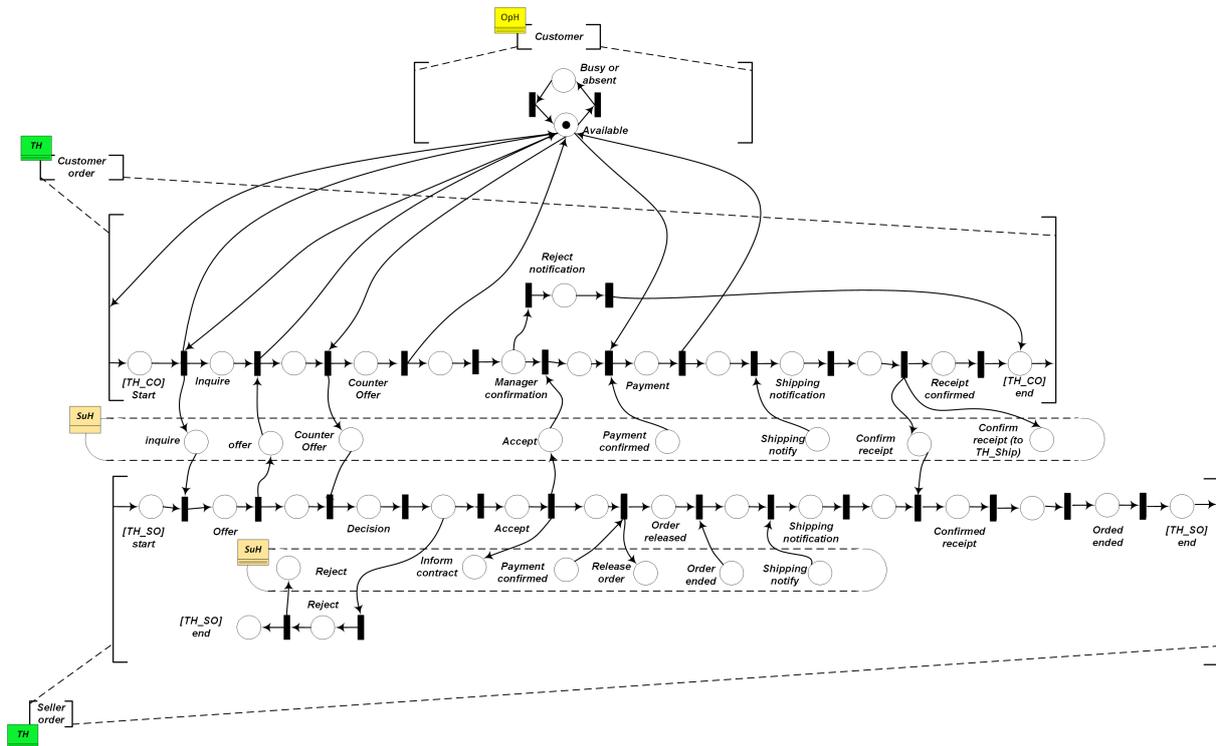


Figura 12: Exemplo da holarquia de *TH-[Customer order]*, *TH-[Seller Order]*, *SuH* e *OpH-[Customer]*.

Silva, R. M., Arakaki, J., Junqueira, F., Santos Filho, D. J. e Miyagi, P. E. (2012). Modeling of active holonic control systems for intelligent buildings, *Automation in Construction* **25**: 20–33.

Strasser, T. e Froschauer, R. (2012). Autonomous application recovery in distributed intelligent automation and control systems, *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* **42**(6): 1054–1070.

Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L. e Peeters, P. (1998). Reference architecture for holonic manufacturing systems: Prosa, *Computers in industry* **37**(3): 255–274.

Vrba, P., Tichý, P., Mařík, V., Hall, K. H., Staron, R. J., Maturana, F. P. e Kadera, P. (2011). Rockwell automation’s holonic and multiagent control systems compendium, *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* **41**(1): 14–30.

Zhang, Y. e Jiang, J. (2008). Bibliographical review on reconfigurable fault-tolerant

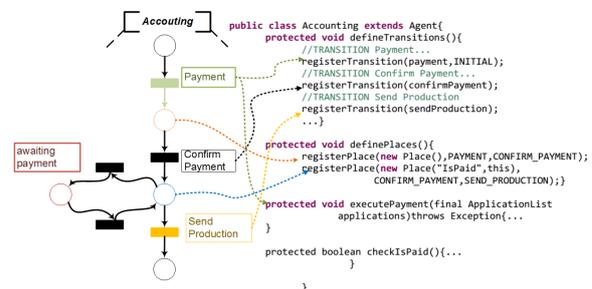


Figura 13: A transformação de modelos em RdPE em linguagem JAVA. Exemplo da transformação da atividade do *TH-[accounting]*.

control systems, *Annual Reviews in Control* **32**(2): 229–252.

Zurawski, R. e Zhou, M. (1994). Petri nets and industrial applications: A tutorial, *IEEE Transactions on Industrial Electronics* **41**(6): 567–583.