

Considerations on Holonic Control of Manufacturing Systems

Valentin Vlad and Adrian Graur

Abstract—This paper presents some considerations on the next generation of manufacturing control systems, from the holonic concepts' perspective. Unlike the classical manufacturing control systems, designed for high-volume, low-variety production, the new control systems are required to be characterized by high agility and adaptability to the environment changes, and robustness against the occurrence of disturbances. Holonic manufacturing control systems provide a high potential in dealing with these problems, bringing the advantages of modularity, decentralization, autonomy and scalability.

Holonic architectures combine the Function Block based programming, for the low-level real-time control, and the multi-agent technologies, for the decision-making level. The IEC 61499 standard, FIPA standardization, and Petri net based modeling are examples of topics covered in this paper.

Index Terms—Distributed control, Holonic manufacturing systems, IEC 61499, Intelligent agents, Petri nets, Reconfigurable systems

I. INTRODUCERE

ÎN ultimele decenii mediul industrial a fost marcat de o serie de schimbări importante cum ar fi trecerea de la o economie locală la o economie globală, orientarea cerințelor pieții spre produse de calitate superioară și cost redus, încercarea companiilor de a satisface cât mai fidel cerințele clienților, fabricarea de produse puternic particularizate și în general cu un ciclu de viață redus. În paralel cu aceasta, evoluția continuă a tehnologiei face necesară adesea actualizarea și integrarea sistemelor existente în sisteme noi cu o arhitectură de control evoluată, pentru a se evita depășirea lor din punct de vedere tehnologic.

Sistemele tradiționale de manufacturare prezintă o optimizare bună a producției dar un răspuns slab la schimbări, datorat în principal rigidității și arhitecturii centralizate a *structurii de control*. În aceste circumstanțe provocarea este de a dezvolta sisteme pentru controlul procesului de manufacturare caracterizate de *autonomie* și *intelență*, agile și adaptabile la schimbările ce pot să apară, robuste în prezența perturbațiilor și care să permită integrarea cu ușurință atât a resurselor actuale cât și a sistemelor

tradiționale.

În acest sens au fost propuse mai multe soluții pentru realizarea sistemelor de manufacturare, dintre care cele mai cunoscute sunt: **sistemele de manufacturare biologice** (*biological manufacturing systems - BMS*) [24], **sistemele de manufacturare holonice** (*holonic manufacturing systems - HMS*) [HMS, 2008], **sistemele de manufacturare fractale** (*fractal manufacturing systems*) [26] și **manufacturarea bazată pe celule virtuale** (*virtual cellular manufacturing - VCM*) [6].

II. SISTEME DE FABRICAȚIE HOLONICE

Conceptul *holonic* a fost definit de către scriitorul și filozoful ungar Arthur Koestler, care a propus termenul de *holon* pentru a descrie o unitate de bază organizatorică în sistemele biologice și sociale [11]. Două proprietăți importante ale holonului fac din conceptul holonic un instrument de modelare adecvat în proiectarea atât a sistemelor de manufacturare cât și a sistemelor de manipulare a materialelor. În primul rând, autonomia oferă holonilor dreptul de a lua decizii fără a consulta vreo entitate superioară. În al doilea rând, cooperarea permite holonilor să comunice cu alți holoni asemenea pentru dezvoltarea de secvențe de planificare mutual acceptate și executarea acestora.

Aplicabilitatea acestor concepte în sistemele de manufacturare a fost menționată prima dată de către Suda [22]. Observațiile lui au condus în 1993 la formarea proiectului HMS (Holonic Manufacturing Systems), cu un studiu de fezabilitate înainte de începerea Fazei 1 a proiectului. Proiectul HMS [9] a fost dezvoltat în cadrul programului IMS (Intelligent Manufacturing Systems) [10]. Conceptul programului IMS a fost formulat în Japonia în 1988 și reprezintă o inițiativă internațională de cercetare și dezvoltare a următoarei generații de tehnologii și norme din sistemele de manufacturare [27]. Programul HMS este unul din primele și cele mai ample proiecte ale programului IMS, implicând peste 50 de organizații din Canada, USA, Europa, Japonia și Australia.

Aplicarea conceptelor holonice în sistemele de manufacturare conduce la o nouă abordare a problemelor din mediul industrial, dat fiind avantajele aduse de acestea: **modularitate**, **descentralizare**, **autonomie** și **scalabilitate**.

III. EXEMPLE DE ARHITECTURI HOLONICE

A. Arhitectura de referință PROSA

PROSA (Product-Resource-Order-Staff Architecture) [5] a fost definită ca o *arhitectură holonică* de referință pentru sistemele de manufacturare, dezvoltată de un grup de cercetători de la Universitatea Catolică din Leuven, Belgia. În cadrul arhitecturii se utilizează holonii pentru a reprezenta produsele, resursele, comenzile și activitățile logice. Această arhitectură definește trei tipuri principale de holoni: holoni produs (*product holons*) holoni resursă (*resource holons*) și holoni de comandă (*order holons*), ca în Fig. 1.

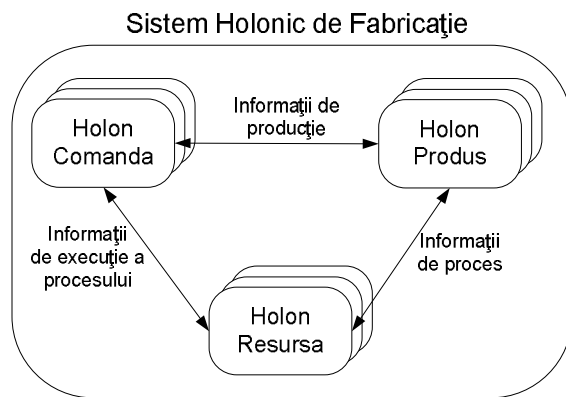


Fig. 1. Holonii de bază propuși în arhitectura PROSA.

Un *holon resursă* este format dintr-o parte fizică, reprezentând una din resursele fizice ale sistemului de manufacturare, și o parte procesoare de informații, având rolul de a controla resursa respectivă. Acest tip de holon conține metodele pentru alocarea resurselor disponibile, precum și cunoștințe de organizare, utilizare și control a resurselor, în scopul realizării producției. Un holon resursă este o abstractizare a unui mijloc de producție, cum ar fi o fabrică, un atelier de producție, mașini, cuptoare, conveiere, componente, materii prime, unelte, dispozitive de prindere a uneltelor, operatori umani, etc.

Holonii produs cuprind cunoștințe despre proces și produs, necesare pentru a asigura realizarea corectă a produsului și cu o calitate acceptabilă. Un holon produs conține informații consistente și actualizate despre ciclul de viață al produsului, cerințele utilizatorilor, proiectare, planul de relizare, lista de materiale necesare (*bill of materials - BOM*), proceduri de asigurare a calității, etc. acționând ca un server de informații pentru ceilalți holoni din sistemul de manufacturare.

Un *holon de comandă* reprezintă un task în sistemul de fabricație, fiind responsabil de realizarea corectă și la timp a activităților asociate. Un holon de comandă poate reprezenta comenzile clienților, comenzi de realizare produse pe stoc, comenzi pentru realizare prototip, comenzi pentru întreținerea și repararea resurselor, etc. Sarcinile realizate sunt cele asociate în mod tradițional unui dispecer, unui monitor al progresului și unui planificator pe termen scurt.

Pe lângă aceste tipuri de bază sunt prevăzuți o serie de *holoni auxiliari* (*staff holons*), cum ar fi planificatorul execuției (*scheduler*), ce au misiunea de a asista și de a “consilia” holonii de bază. Ei analizează diverse aspecte ale problemelor holonilor de bază furnizând acestora informații în scopul luării de decizii corecte pentru rezolvarea problemelor. Responsabilitatea luării deciziei rămâne holonului de bază, holonul auxiliar fiind considerat ca un expert extern ce dă anumite sfaturi.

Această arhitectură combină robustețea și predictibilitatea controlului ierarhic cu reacția rapidă la perturbații, datorată controlului hetero-ierarhic.

B. Arhitectura holonică ADACOR

ADACOR (*ADaptive holonic COntrol aRchitecture for distributed manufacturing systems*) este o arhitectură holonică pentru sistemele de manufacturare distribuite, propusă de Paulo Leitão [12].

Arhitectura ADACOR este intenționată a avea o structură de control cât mai descentralizată posibil și în același timp centralizată atât cât este necesar, pentru a se putea asigura optimizarea globală a sistemului. Ea se bazează pe un set de entități autonome și cooperative, reprezentate de *holoni*, ce permit o distribuție a abilităților și cunoștințelor și asigură îmbunătățirea capacității de adaptare la schimbările mediului. Fiecare holon este o reprezentare a unei componente de fabricație ce poate fi fie o resursă fizică (mașini comandate numeric - CNC, roboți, controlere programabile, paleți, etc.), fie o entitate logică (comenzi, produse).

Holonii ce formează arhitectura sunt grupați în următoarele clase: holoni produs, holonii task, holoni operaționali și holoni supervisor, după cum se poate observa și din Fig. 2.

Holonii supervisor sunt inspirați din sistemele biologice și prezintă caracteristici diferite față de holonii auxiliari (*staff holons*) definiți în cadrul arhitecturii de referință PROSA. Ei reprezintă un aspect inovativ al arhitecturii propuse și au rolul de a introduce o coordonare și optimizare globală în cadrul soluțiilor cu control descentralizat. Un holon supervisor poate realiza coordonarea câtorva holoni operaționali, sau a altor holoni supervisor.

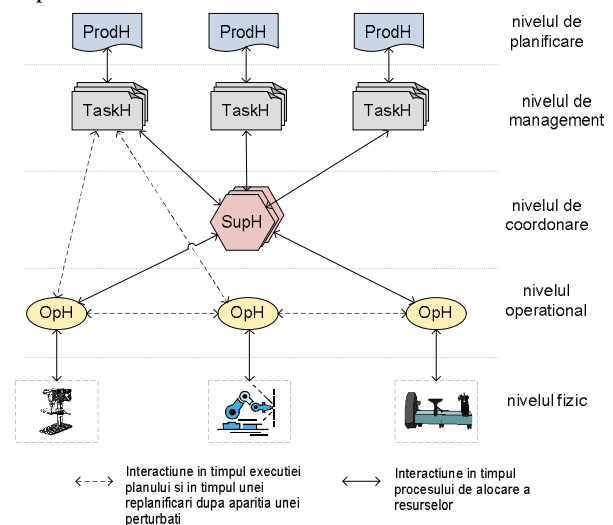


Fig. 2. Clasele de holoni din cadrul arhitecturii ADACOR (adaptat din [12]).

Un *holon operațional* poate fi format dintr-un set de holoni operaționali sau holoni supervisor, ceea ce permite construirea de holarhii *fractale*, așa cum este ilustrat în Fig. 3.

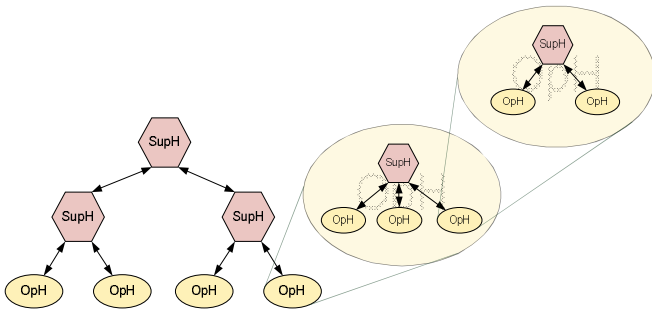


Fig. 3. Organizarea fractală a holonilor.

Holonii din cadrul arhitecturii ADACOR se bazează pe principiul *plug and produce* putându-se adăuga un nou element în sistem fără a fi nevoie de reinițializarea sau de reprogramarea sistemului, ceea ce conduce la o flexibilitate ridicată în reconfigurarea sistemului. La intrarea unui holon în sistem acesta anunță și oferă serviciile sale celorlalți holoni. La părăsirea sistemului de către un holon, holonii rămași trebuie să fie capabili să găsească soluții alternative pentru a acoperi lipsa produsă de holonul plecat.

Conform [12], îmbunătățirile aduse de introducerea arhitecturii ADACOR pot fi observate pe două niveluri diferite: nivelul de proiectare și nivelul operațional.

La *nivelul de proiectare*, dat fiind avantajul utilizării sistemelor descentralizate, proiectarea, întreținerea, expandarea și reutilizarea aplicațiilor din controlul manufacturării este realizată mult mai simplu decât în cazul soluțiilor tradiționale. La *nivelul operațional*, arhitectura propusă furnizează un mecanism de adaptare a structurii sistemului de control al fabricației în cazul apariției perturbațiilor și de revenire la modul normal de operare după dispariția perturbației. În acest fel, optimizarea controlului centralizat, în absența perturbațiilor, este combinată cu agilitatea controlului hetero-ierarhic în situații instabile determinate de apariția frecventă a perturbațiilor.

IV. PROIECTAREA ȘI MODELAREA ARHITECTURILOR HOLONICE

Acest capitol prezintă succint două instrumente de modelare, folosite în proiectarea sistemelor de manufacturare holonice: *limbajul standard UML* (*Unified Modelling Language*) și *rețelele Petri*.

A. Limbajul UML

UML un limbaj standard universal de modelare, orientat obiect, ce permite specificarea, vizualizarea și documentarea de modele pentru sisteme software precum și pentru alte tipuri de aplicații [17]. *UML* este succesorul propriu-zis al

celor mai bune trei limbaje de modelare anterioare orientate pe obiecte: Booch [2], OMT [19] și OOSE [14]. El se constituie din unirea acestor limbaje de modelare și în plus deține o expresivitate care ajută la rezolvarea problemelor de modelare pe care vechile limbaje nu o aveau.

Deși este foarte adecvat pentru modelarea aspectelor statice ale sistemelor software, limbajul *UML* prezintă unele neajunsuri referitor la modelarea comportamentală a sistemelor, făcând uneori imposibilă dezvoltarea de sisteme cu un comportament complex.

B. Rețelele Petri

Rețelele Petri reprezintă un instrument matematic și grafic pentru modelarea sistemelor informaționale concurente, asincrone, distribuite, paralele, indeterminate sau stocastice [16]. Ele asigură un compromis între ușurința în utilizare și puterea de reprezentare, permițând descrierea într-un mod relativ simplu a proceselor și a relațiilor dintre ele. Spre deosebire de limbajul *UML*, formalismul rețelelor Petri permite atât reprezentarea comportamentală a sistemului de control cât și validarea și verificarea comportamentului acestuia.

Din punct de vedere istoric, *Rețelele Petri* își au originea în teza de doctorat a cercetătorului german Carl Adam Petri, intitulată "Kommunikation mit Automaten", scrisă în 1962 [18].

O rețea Petri este alcătuită din *locuri*, *tranziții* și *arce orientate*. Arcele asociază o tranziție unui loc și viceversa. Nu exista arce între două locuri și nici între două tranziții. Locurile pot conține un număr variabil de jetoane. Tranzițiile se produc consumând jetoane din locurile de intrare și producându-le în cele de ieșire. O tranziție se numește *activă* în momentul în care conține jetoane în fiecare din intrările sale.

Jetoanele într-o rețea Petri standard nu se disting între ele ca și aparență vizuală. Pentru acest lucru se folosesc *rețelele Petri Colorate* (*Colored Petri Nets*), deseori regăsite sub acronimul CPN. Activarea unei tranziții este determinată în totalitate de prezența jetoanelor în locurile de intrare.

Matematic, o rețea Petri poate fi reprezentată printr-un cvintuplu de forma $PN = (P, T, F, W, M_0)$, unde:

- $P = \{p_1, p_2, \dots, p_m\}$ - reprezintă o mulțime finită de locații;
- $T = \{t_1, t_2, \dots, t_n\}$ - reprezintă o mulțime finită de tranziții;
- $F \subseteq (P \times T) \cup (T \times P)$ - reprezintă o mulțime de arce;
- $W : F \rightarrow \{1, 2, 3, \dots, \mathbf{L}\}$ - reprezintă funcția pondere;
- $M_0 : P \rightarrow \{0, 1, 2, 3, \dots, \mathbf{L}\}$ - reprezintă marcajul inițial.

Există o largă varietate de extensii ale rețelelor Petri, dintre care amintim:

- **RP cu priorități** – sunt utilizate atunci când se dorește impunerea unei alegeri între mai multe tranziții validate. Sunt compuse dintr-o rețea Petri și o relație între tranzițiile rețelei
 - **RP neautonome** – permit modelarea sistemelor la care execuțiile sunt sincronizate cu evenimente externe și/sau la care evoluțiile sunt funcție de timp;
 - **RP sincronizate** – în cazul acestor rețele fiecarei tranziții îi este asociat un eveniment și execuția acestei tranziții se va efectua dacă tranziția este validată și când evenimentul asociat se va produce.
- RP temporizate** – permit descrierea unui sistem a cărui funcționare depinde de timp;

Având în vedere caracteristicile rețelelor Petri prezentate, se poate observa că ele reprezintă un instrument adecvat pentru modelarea, analiza și validarea sistemelor de control holonice.

Fig. 4 prezintă un exemplu de modelare a unui *holon produs*, din cadrul unui sistem holonic de manufacturare, preluat din [13].

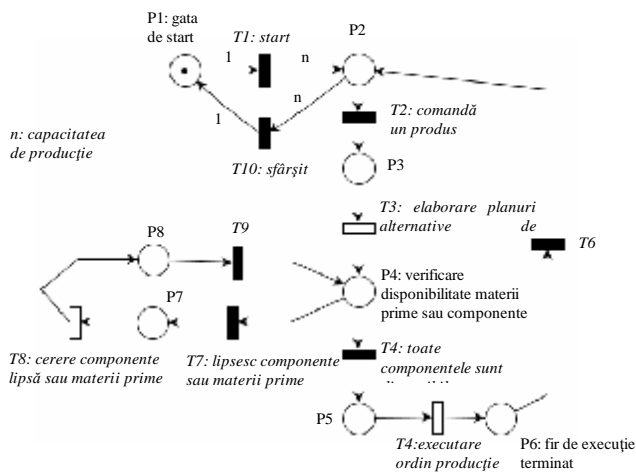


Fig. 4. Modelul comportamental al unui holon produs.

Holonul produs conține toate cunoștințele necesare realizării unui anumit produs. Fiecare nou ordin de realizare a unui produs generează un fir de execuție care se ocupă de planificarea pe termen scurt a procesului și de managementul execuției ordinului de producție. Holonul produs așteaptă în continuare noi ordine, fiind capabil să proceseze simultan mai multe astfel de ordine, funcție de capacitatea de producție n .

În cadrul modelului prezentat, holonul produs interacționează cu un *holon operațional*, responsabil cu managementul componentelor stocate, pentru a verifica disponibilitatea componentelor sau a materialelor prime necesare. Dacă acestea nu sunt disponibile, holonul produs interacționează cu alți holoni produs, pentru a cere execuția subproduselor, invocând în mod recursiv modelul holonului produs.

V. AGENȚII INTELIGENȚI ÎN SISTEMELE HOLONICE

În sens larg, un *agent* reprezintă un sistem computațional situat într-un mediu de execuție și capabil de acțiune autonomă în acel mediu pentru a-și îndeplini obiectivele propuse. Un *agent inteligent* reprezintă un sistem hardware sau mai ales software care se bucură de următoarele proprietăți: (a) autonomie, (b) reactivitate, (c) proactivitate și (d) abilitate socială.

Un *sistem multiagent* este un sistem care constă din cel puțin doi agenți inteligenți capabili să interacționeze între ei în vederea realizării unor scopuri individuale sau comune (scopuri globale) și care partajează același mediu de lucru. Sistemele multiagent (*Multiagent Systems - MAS*) constituie un subcâmp al Inteligenței Artificiale, bazat în mare parte pe domeniul Inteligenței Artificiale Distribuite (DAI) [1].

Discuțiile din cadrul consorțiului HMS au condus în final la identificarea unei arhitecturi generice pentru dispozitivele de control holonice, numită *HCD Architecture* [3], bazată pe două niveluri: un *nivel deliberativ*, reprezentat de o componentă software agent de nivel înalt, și *nivelul fizic*, ce conține senzori și dispozitive de acționare și este supus constrângerilor de timp real. Comunicația între cele două niveluri se realizează prin intermediul unui bloc funcțional special numit *nivelul funcțiilor de control (control functions layer)*. Un holon definit în acest mod încapsulează astfel într-o singură structură atât tehnologiile bazate pe agenți inteligenți cât și aplicațiile de control clasice și poartă numele de *agenți holonici*.

Conform [15] în cadrul acestei arhitecturii pot fi identificate trei canale de comunicație:

- comunicația *intra-holon* între componenta de control de nivel jos și componenta *agent*, de nivel înalt;
- comunicația *inter-holon* ce se ocupă de transmiterea de mesaje între componentele agent ale holonilor;
- comunicația *directă*, realizată între componentele destinate realizării controlului de timp real ale holonilor.

Pentru comunicația inter-holon se utilizează în general standardizarea FIPA (Foundation for Intelligent Physical Agents) [8], comunicația directă realizându-se pe baza altor standarde, cum ar fi IEC 61499, ce asigură suport pentru constrângeri de timp real.

VI. STANDARDUL IEC 61499

Standardul internațional IEC 61499 definește o arhitectură deschisă (*open architecture*) destinată următoarei generație de sisteme din domeniul controlului distribuit și automatizării. Elementele cheie considerate în cadrul unui sistem distribuit sunt *aplicația*, *dispozitivul* și *resursa*.

Aplicația este constituită dintr-un set de funcții care comunică între ele pentru îndeplinirea unui task de control. Un *dispozitiv* reprezintă o unitate de control ce conține unul

sau mai multe procesoare, o interfață I/O, și deasemenea comunică cu alte dispozitive din rețea. O *resursă* este în esență un procesor pe care rulează o parte din aplicația distribuită.

Unitatea programabilă în cadrul arhitecturii IEC 61499 este *blocul funcțional (function block)*, pe care se bazează construirea întregii aplicații. Există trei tipuri de blocuri funcționale: blocuri funcționale de bază (*basic FB*), blocuri funcționale compuse (*composite FB*) și blocuri funcționale de interfațare (*service interface FB*).

Un bloc funcțional de bază (Fig. 5a) execută o funcție elementară de control, cum ar fi citirea unui senzor sau setarea stării unui actuator, și conține *algoritmi* și o *diagramă de control a execuției (execution control chart - ECC)*. Blocurile funcționale de bază pot fi combinate împreună într-un *bloc funcțional compus* (Fig. 5b), pentru a încapsula o funcție de control de nivel mai înalt. Blocurile funcționale de interfațare furnizează servicii pentru comunicația dintre dispozitive.

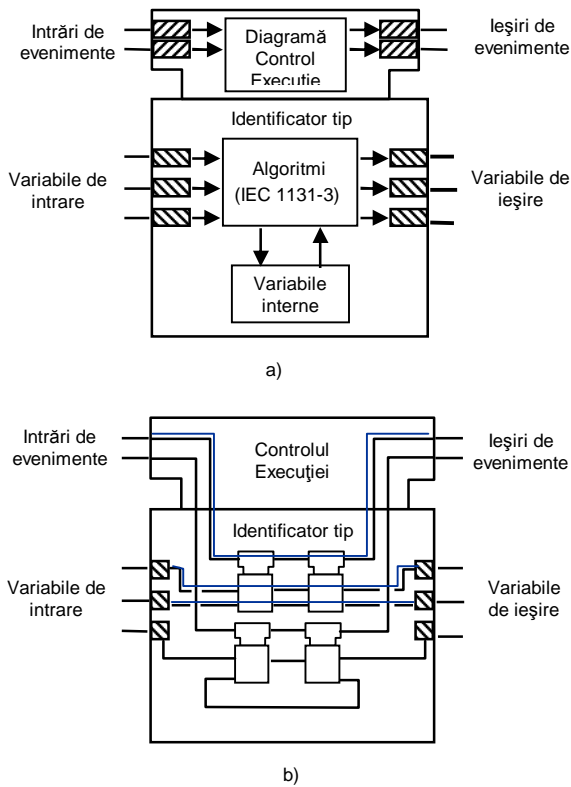


Fig. 5. a) bloc funcțional de bază; bloc funcțional compus (adaptat din [7]).

Standardul utilizează un *model de execuție bazat pe evenimente*. Fiecare bloc funcțional are intrări și ieșiri pentru evenimente, precum și intrări și ieșiri pentru date. Apariția unui eveniment determină execuția unuia sau a mai multor algoritmi din cadrul blocului. În cadrul algoritmului datele de la intrare sunt preluate și prelucrate în scopul obținerii datelor de ieșire. La finalizarea execuției algoritmului este generat un eveniment la ieșire, eveniment ce poate fi utilizat ca eveniment de intrare pentru un alt bloc funcțional [25].

VII. RECONFIGURAREA SISTEMELOR DE FABRICAȚIE

Reconfigurarea în cadrul unui sistem se referă la posibilitatea acestuia de a răspunde la anumite schimbări prin reorganizarea resurselor disponibile, așa încât performanța sistemului să fie afectată cât mai puțin sau chiar deloc.

Pentru a satisface noile provocări în sistemele moderne de manufacturare a fost introdus conceptul de *sistem reconfigurabil*. Un sistem de acest tip poate să se reintegreze la o nouă producție într-un timp foarte scurt folosind module hardware și software. Reconfigurabilitatea permite adăugarea, excluderea sau modificarea unor unități de producție specifice, control, software, sau modificarea structurii mașinii pentru o adaptare optimă la cerințele pieței. Pentru ca sistemul să dețină calitatea de reconfigurabilitate trebuie să îndeplinească anumite condiții. Acestea includ: *modularitatea* pentru componentele hardware cât și pentru componentele software, *integrabilitatea* (proiectarea sistemului și a componentelor atât pentru integrarea rapidă cât și pentru introducerea noilor tehnologii), *diagnozabilitate* (capacitatea de identificare rapidă a problemelor de calitate și defectare din sistem) și *personalizare* (proiectarea pentru adaptarea la producția unui anumit produs prin intermediul componentelor hardware și software) [20].

Reconfigurarea unui *sistem distribuit* poate fi privită pe trei niveluri de complexitate [4]: simplă, dinamică și inteligentă. *Configurarea simplă* vizează utilizarea unui model cum ar fi IEC 61499 în scopul evitării problemelor datorate cuplării software, probleme ce ar putea să apară în timpul reconfigurării sistemului. *Reconfigurarea dinamică* utilizează tehnici pentru sincronizarea optimă a software-ului în timpul reconfigurării, iar *reconfigurarea inteligentă* exploatează tehnici multi-agent pentru a permite reconfigurarea automată a sistemului, ca răspuns la schimbările apărute.

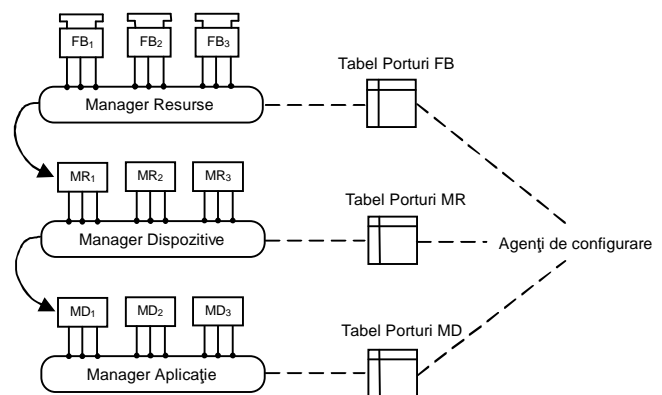


Fig. 6. Model de reconfigurare (adaptat din [4]).

În lucrarea [4] este prezentat un model de reconfigurare pentru un sistem holonic, la nivelul controlului de timp real. În cadrul modelului propus (Fig. 6) porturile blocurilor funcționale (conexiunile pentru date și evenimente) sunt conectate la un manager de resurse care se ocupă de

interconexiunile dintre blocuri și păstrează într-un tabel o înregistrare a tuturor porturilor.

Modelul IEC 61499 definește o resursă ca fiind formată din mai multe blocuri funcționale interconectate, iar un dispozitiv din una sau mai multe resurse. Ca urmare relația dintre resurse și dispozitive este asemănătoare relației dintre blocurile funcționale și resurse. În acest sens, *managerul de dispozitive* se ocupă de conexiunile dintre porturile resurselor, înregistrate în *Tabel Porturi MR*. Similar, *managerul de aplicații* se ocupă de conexiunile porturilor managerilor de dispozitive, păstrând informațiile aferente în *Tabel Porturi MD*.

Folosind acest model, un manager de dispozitive, de exemplu, poate realiza reconfigurarea porturilor managerilor de resurse conectate la el pentru a corespunde oricărei noi configurații dorite. Avantajul soluției prezentate constă în faptul că reconfigurarea poate fi realizată la diferite niveluri (blocuri funcționale, resurse, dispozitive, aplicații), singurul lucru necesar fiind o *hartă* a interconexiunilor noii configurații.

O altă abordare a problemei reconfigurării sistemelor de manufacturare este prezentată în lucrarea [21] și are la bază conceptul *celulelor virtuale de manufacturare*, prin care se urmărește creșterea performanțelor sistemelor bazate pe *celule de manufacturare*. Acest concept presupune analiza performanțelor sistemului și regruparea resurselor disponibile în celule virtuale, dacă este nevoie, oricâteori apare un nou ordin de producție. Formarea celulelor virtuale urmărește minimizarea sau maximizarea unor indici de performanță: interacțiunea dintre celule, media întârzierilor în sistem, media utilizării sistemului, capacitatea de producție medie. Pentru *interacțiunea dintre celule* se utilizează o ecuație analitică, valorile pentru ceilalți trei indicatori de performanță fiind determinate cu ajutorul unui modul de simulare/planificare.

Un aspect important al reconfigurării este conceptul *Plug and Produce* prezentat în [23]. Similar conceptului *Plug and Play* folosit pentru dispozitivele din componența unui computer, acest concept se referă la posibilitatea adăugării sau eliminării cu ușurință a dispozitivelor din componența unui sistem distribuit, prin actualizarea automată a informațiilor din cadrul sistemului, reducând în acest fel costurile reconfigurării. Aplicarea conceptului *Plug and Produce*, împreună cu o arhitectură autonomă descentralizată conduce la obținerea unui grad de flexibilitate mult mai ridicat pentru sistemele de manufacturare.

REFERENCES

- [1] Alexandru, L., *Agenți inteligenți. Sisteme multiagent*, Universitatea Tehnică "Gh. Asachi" Iași, 2004.
- [2] Booch, G. *Object Oriented Design with Applications*. Benjamin/Cummings, Redwood City, CA, USA, 1991.
- [3] Brennan, R. W., Hall, K., Marik, V., Maturana, F. P., & Norrie, D. H. *A real-time interface for holonic control devices*. In *Holonic and multi-agent systems for manufacturing*. Lecture notes in artificial intelligence (Vol. 2744) (pp. 25–34). Heidelberg: Springer, 2003.
- [4] Brennan, R. W., Fletcher, M., Norrie, D.H., *Reconfiguring Real-time Holonic Manufacturing Systems*. University of Calgary, 2001.
- [5] Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L. Peeters, P. *Reference architecture for holonic manufacturing systems: PROSA*. Computers in Industry, Vol. 37, 1998
- [6] Drolet, J.R., Moodie, C. L., Montreuil B. *Scheduling factories of the future*, J. Mech. Work. Technol. 20 (1989) 183–194.
- [7] Christensen, J.H. *The IEC 61499 Standard: Concepts and R&D Resources*, in Tutorial at 1st IEEE Intl. Conf. Industrial Informatics INDIN'03, Canada.
- [8] FIPA. Foundation for Intelligent Physical Agents – website. <http://www.fipa.org/>, 2008
- [9] HMS. Holonic Manufacturing Systems Consortium - web site. <http://hms.ifw.uni-hannover.de>, 2008.
- [10] IMS. Intelligent Manufacturing Systems - web site. <http://www.ims.org>, 2008.
- [11] Koestler, A. *The Ghost in the Machine*. Arkana Books, London, 1969
- [12] Leitão, P. *An Agile and Adaptive Holonic Architecture for Manufacturing Control*. Phd Thesis, University of Porto, 2004
- [13] Leitão, P., Colombo, A.W. and Restivo, F. *A formal specification approach for Holonic Control Systems: the ADACOR case*. Int. J. Manufacturing Technology and Management, Vol. 8, Nos. 1/2/3, pp.37-57.
- [14] Jacobson, I. *Object-Oriented Software Engineering*, Addison-Wesley, 1992.
- [15] Marik, V., Lazansky, Jiri. *Industrial applications of agent technologies*. Control Engineering Practice 15, 2007.
- [16] Murata, T., *Petri Nets: Properties, Analysis and Applications*. Proceedings of the IEEE, Vol. 77, No. 4, pages 541-580. April 1989.
- [17] Object Management Group – web site. http://www.omg.org/gettingstarted/what_is_uml.htm
- [18] C.A. Petri, "Kommunikation mit Automaten." Institut fur Instrumentelle Mathematik, Schriften des IIM Nr.3, 1962. Traducere în engleză, "Communication with Automata." New York: Griffiss Air Force Base. Tech.Rep.RADC-TR-65-377, vol. 1, Suppl.1, 1966.
- [19] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, S., Lorensen, W. *Object Oriented Modelling and Design*. Prentice Hall, Englewood Cliffs, NJ, USA, 1991.
- [20] Proiect RECONFIG. *Metode de simulare, modelare si productie virtuala sau digitala bazate pe tehnologia informatiei si comunicarii (ICT) si dedicate noii generatii de sisteme de prelucrare reconfigurabile*, Raport de cercetare, PROGRAM CEEX Nr. 22-I03/10.10.2005 – INFOSOC, 2005.
- [21] Saad, S.M., *The reconfiguration issues in manufacturing systems*, Journal of Material Processing Technology 138 (2003) 277-283.
- [22] Suda, H. *Future factory systems formulated in Japan*. Techno Jpn 1990; 23(3).
- [23] Sugii, M., Maeda, Y., Aiyama, Y., Harada, T., Arai, T. *A Holonic Architecture for Easy Reconfiguration of Robotic Assembly Systems*, IEEE Transactions on Robotics and Automation, Vol. 19, No. 3, June 2003.
- [24] Ueda, K., Vaario, J., Ohkura, K. *Modelling of biological manufacturing systems for dynamic reconfiguration*, Ann. CIRP 46 (1997) 343–346.
- [25] Vyatkin, V. *IEC 61499 Function Blocks for Embedded and Distributed Control Systems Design*, Instrumentation Society of America, USA, July, 2006.
- [26] H.J. Warnecke, *The Fractal Company*, Springer, Berlin, 1993.
- [27] Yoshikawa, H. *Intelligent manufacturing systems*. In: Yoshikawa, Goossenaerts, editors. Information infrastructure systems for manufacturing (IFIP Transactions B-14). London: Elsevier; 1993.