Available online at www.sciencedirect.com
ScienceDirect
IFAC-PapersOnLine 48-3 (2015) 187–192

Motion Coordination of AGV's in FMS using Petri Nets

E.G. Hernández-Martínez *, Sergio A. Foyo-Valdés, Erika S. Puga-Velazquez.****
J.A. Meda Campaña **

**Departamento de Ingenierías, Universidad Iberoamericana Ciudad de México, C.P. 0121, México, D.F. (e-mail: eduardo.gamaliel@ibero.mx).*
*** Mechanical Engineering Department, Sección de Estudios de Posgrado e Investigación, ESIME Zacatenco, Instituto Politécnico Nacional, C.P. 07738 México, D.F (e-mail: sergiofoyo@live.com.mx, erika_selenep@hotmail.com, jmedac@ipn.mx)*

Abstract: This paper proposes a modeling strategy for automated Flexible Manufacturing Systems (FMS) that incorporates automated guided vehicles (AGV's) as material handling systems. Using the industrial standard ISA-95, the task-based coordination of equipment and storages is constructed considering process restrictions, logical precedence conditions, shared resources and the assignment of the robots to move work pieces individually or in subgroups. The Petri Net model calls in the low level to formation, marching and collision avoidance control laws, for omnidirectional robots. The hybrid architecture is implemented and validated for the case of a FMS and four mobile robots.

Keywords: Petri nets, Multi-robot systems, Formation control, Marching control, Automation.

1. INTRODUCTION

Nowadays, the FMS involves new technologies of automation in order to coordinate the assembly of different and concurrent products (Groover (2008)). The Petri Net (PN) formalism has served to represent the asynchronous firing of actions, blocking, concurrency and other dynamic behaviours frequently appeared in large FMS (Li (2014)).

The Material-Handling System (MHS) is a set of resources for the carrying of workpieces between storages and workstations. The installation of AGV's, emulating the coordinated work achieved by a group of workers, proposes a more flexible MHS, substituting some traditional fixed and non-reconfigurable transport setup, like conveyor belts or manipulators mounted in rails. The collective behaviour of groups AGV'S presents some advantages like redundancy and fault tolerance when a robot is broken and the loading of large objects by subgroups of robots in specific formation patterns (Krontiris (2013)). An industrial AGV's setup is the Kiva system (Wurman (2008) and D'Andrea (2012)) where shelves are charged and moved by small autonomous robots, sharing information about inventories and work orders (Enright (2011)).

The motion coordination of AGV's in a FMS must enable formation control, group path following or marching and inter-robot collision avoidance (Ailon (2012)). At same time, this coordination must obey process specifications, fault tolerance strategies and the routes assignment (Chen (2014)). In the context of mobile robotics control, the motion coordination of multiple robots, mainly omnidirectional and unicycle-type robots, has been studied using behavior-based control laws (Bravo (2011)), swarms stability (Fidan (2013)), bio-inspired navigation (Savkin (2013)), virtual structures (Chang (2011)), leader-followers schemes (Zhao (2010)) and artificial potential functions (Kowdiki (2012)).

Although the possibilities to implement the previous control laws in the context of FMS, the most of the previous works do not clarify how these low-level control strategies can be connected to a coordination layer of MHS within a production system. Moreover, the coordination layer must be designed according to industrial standards as ISA-95 (Instrument Society of America (1999)). However, commonly the discrete-event community studies only the high-level behavior of FMS excluding the analysis of the motion of the AGV's, for instance (Gradišar (2012)) and (Sanchez (2010)). Few works like (Hernandez-Martinez (2011)), propose some hybrid architectures of formation control and a planning level, in this case neural networks, for dispersion tasks. A supervisory control for Finite State Automata using AGV´s is obtained briefly in (Sanchez (2009)) for an FMS, which presents the drawback of the state explosion for a real case. Hybrid architectures of PN and multi-agent systems have been addressed for communication in computer systems (Celaya (2009)).

This paper proposes a procedure to class of FMS using PN according to the task-based coordination proposed by the ISA-95 standard. The PN model represents the concurrency of tasks obeying the logic of precedence between tasks, the limitation of storages and the availability of the robots. The transportation tasks selects the adequate AGV's to move the pieces implementing three basic continuous control laws to achieve formations, marching and collision avoidance of groups of robots. The approach was preliminarily presented in Hernandez-Martinez (2014) using a different case of study. Numerical simulations show the hybrid control performance. The approach links the two control levels clarifying the implementation of hybrid control in the engineering practice.

10.1016/j.ifacol.2015.06.079

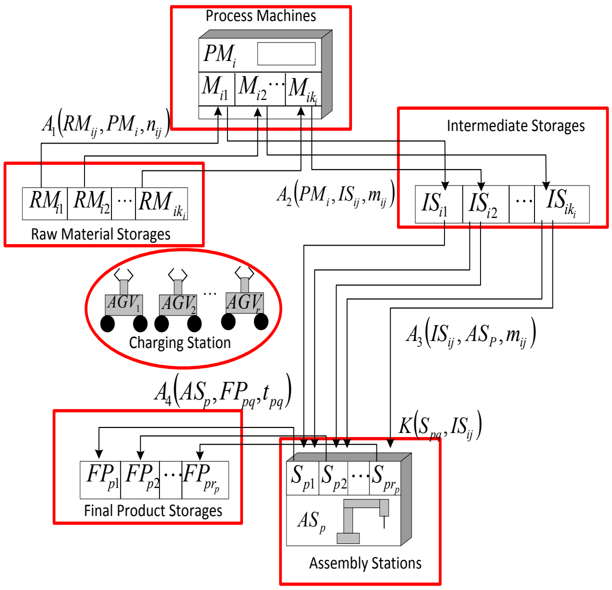

Fig. 1. FMS general scheme

2. PROBLEM DEFINITION

The general scheme of the FMS is presented in the Fig. 1 according. It is composed by Raw Material Storages, a set of Machines, for instance Computer Numerical Control (CNC) machines, programmed to machine subparts which are moved in (temporal) Intermediated Storages, like Matrix-shaped Storages, where each row contains the parts manufactured by the same machine. The adequate number of these subparts is transported to Assembly Stations to manufacture different products which are finally placed in Final Product Storages. A set of homogeneous AGV's (omnidirectional robots) is added to the system to transport the raw material, subparts or final products between the different equipment of the FMS. It is assumed that one work piece could be transported by one or many AGV's, generating the formation of the robots in specific geometric patterns and they marching in predefined routes. As shown Fig. 1, the AGV's start in a home position, where the robots could be connected to a battery charging station. When a robot finishes a task, it can return to its home position or eventually to attend another transportation task.

The main objective is to design a PN model that enables the basic equipment of the FMS including the AGV's tasks. On the other hand, the robots must receive the necessary information to implement formation, marching and collision avoidance control laws.

3. DEFINITIONS OF EQUIPMENT AND TASKS

Let $PM = \{PM_1, \dots, PM_n\}$, $AS = \{AS_1, \dots, AS_m\}$ and $AGV = \{AGV_1, \dots, AGV_r\}$ be the set of Process Machines, Assembly Stations and AGV's, respectively. It is assumed that each PM_i, realize $M_{i1}, \dots, M_{ik_i}$ machining programs. Every M_{ij} with $i = 1, \dots, n, j = 1, \dots, k_i$ requires the raw material contained in the Raw Material Storages RM_{ij} manually loaded with capacity limited to $C(RM_{ij})$ pieces. The subpart manufactured by the program M_{ij} is allocated in the Intermediated Storage IS_{ij} with capacity equal to $C(IS_{ij})$.

Denote by $A_1(RM_{ij}, PM_i, n_{ij})$ the AGV's transportation tasks a raw material from RM_{ij} to PM_i using $n_{ij} \leq r$ AGV's and denote by $A_2(PM_i, IS_{ij}, m_{ij})$ the transportation from PM_i to IS_{ij} using $m_{ij} \leq r$ AGV's. Note that $m_{ij} \leq n_{ij}$ due to the decrease of the dimension and weight of the machining action. If $n_{ij}, m_{ij} > 1$, implies that the robots achieve a formation control and path following that will be detailed in the next section.

Each Assembly Station $AS_p, p = 1, \dots, m$ realize $S_{i1}, \dots, S_{ir_p}$ assembly programs. Let $U(S_{pq}) \subseteq \{IS_{11}, \dots, IS_{nk_n}\}$ be the set of Intermediated Storages that contains the subparts needed to complete the assembly program S_{pq} and $K(S_{pq}, IS_{ij}) \in Z^+, \forall IS_{ij} \in U(S_{pq})$ the quantity of subparts of IS_{ij} to achieve the assembly program S_{pq}. Then, we define $A_3(IS_{ij}, AS_p, m_{ij})$ as the task related to the transportation of a part from IS_{ij} to AS_p. Note that the previous task must be repeated $K(S_{pq}, IS_{ij})$ times but not necessarily by the same robots. Finally, when the assembly program S_{pq} has finished, the AGV's do the $A_4(AS_p, FP_{pq}, t_{pq})$ tasks to move the final product from AS_p to the Final Product Storages FP_{pq} (with capacity $C(FP_{pq})$ and manually unloaded), using now $t_{pq} \leq r$ robots. Fig. 1 summarizes the notation of the tasks of PM, AS and the routes of the AGV's in the system. Note that some arrows are related to the number of pieces $K(S_{pq}, IS_{ij})$ moved by the robots.

Based on the task decomposition proposed in the ISA-95 standard, Fig. 2 shows the tasks of the FMS with a clear separation of the MHS (the AGV's and storages) and the process workstations (machines and assembly stations). The ISA-95 establishes that all the product sequences are reduced to the correct ordering (product recipe) of these tasks supervised by a computer system, avoiding the reprogramming of routines in the local controllers of the FMS elements when a new product is required.

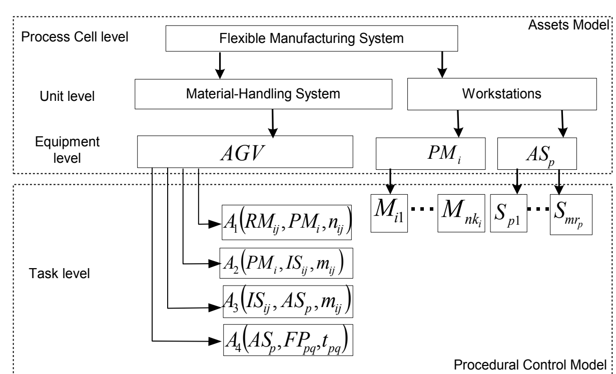

Figure 2. Task decomposition using the ISA-95 standard

Fig. 3 shows the normal process flow between tasks (rectangles) and storages (circles). The wide arrow denotes the AGV's tasks; the continuous or dotted arrows are direct or inverse precedences between a pair of tasks, respectively. The direct logical precedences, for example $A_1 - M_{ij}$ and $M_{ij} - A_2$, establishes that only when the precedent task ends, the subsequent task can be enabled. On the other hand, a inverse precedence occurs when the finish of a posterior task enables again the start of an initial task, in a normal functional flow of the FMS. For example, when the AGV's finish the task A_2, they can transport again raw materials to the machine PM_i, it becomes in an "inverse" logical precedence conditions (dotted line) between $A_2 - A_1$. The second section of the Fig. 3 shows the tasks A_3 repeated $K(S_{pq}, IS_{ij})$ times to gather the subparts to make the assembly S_{pq} and move the products to the storage FP_{pq} through a task A_4. Note an inverse precedence of $A_4 - A_3$ which ensures that the AGV's reload the assembly station only when the previous product has been stored in FP_{pq}. Next section describes the equipment models and the relationships given in Fig. 3 translated to PN.

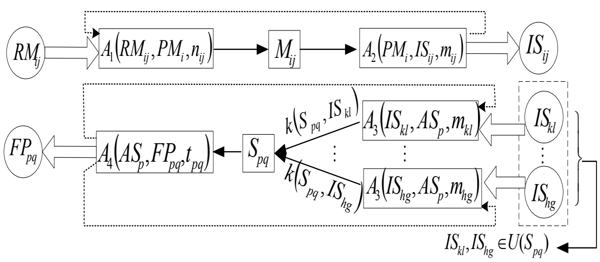

Fig. 3. Logical precedence conditions of a FMS

4. PN MODELS OF THE FMS

According to (Cassandras (2008)) a PN with finite capacity is a weighted and bipartite graph given by a 5-tuple *PN= (P, T, F, W, M0),* where $P = \{p_1, p_2, \ldots, p_m\}$ and $T = \{t_1, t_2, \ldots, t_n\}$ are the disjoint sets of nodes called places and transitions, respectively. $F \subseteq (P \times T) \cup (T \times P)$ is the set of arcs, connecting places to transitions and vice versa. $W: F \rightarrow Z^+$ is the function that assigns the weights to each arc and $M(p_i): P \rightarrow Z^+$, represents a m-entry vector with the number of tokens residing inside each $p_i \in P$ which has a finite capacity c_i of tokens, i.e. $M(p_i) \leq c_i$. Let M_0 as the initial marking.

A transition $t_i \in T$ is said to be enabled in a PN with finite capacity iff $M_k(p_j) \geq w(p_j, t_i), \forall p_j | w(p_j, t_i) \in F$, with j=1,…,m and i=1,…,n, with the restriction $w(t_i, p_j) + M_k(p_j) \leq c_j, \forall p_j \in P$. If t_i is firing in the k-th firing in some firing sequence, then the next marking state is defined by $M_{k+1}(p_j) = M_k(p_j) - w(p_j, t_i) + w(t_i, p_j)$. The set of all possible markings reachable from M_0 construct the reachability tree.

4.1 Tasks of AGV's, Machines and Assembly stations

The tasks of AGV's, machines and assembly stations are given in Figure 4. The prefixes "s" and "f" denotes the start and final of tasks, respectively. In the tasks $A_i(X, Y, w)$ $i = 1, \ldots, 4$, w is the quantity of robots needed to perform the task, according to the table 4a. The number of AGV's is the tokens of the place labelled as AGV. When a task finished, the w tokens are returned to the AGV place (available robots again). In the most simple criteria, the selection of the AGV's fro a task depends on the smallest distance of the robots with respect to the point where the robots pick up the piece. Similar to AGV's, the machining and assembly tasks are represented in Figures 4b and 4c, respectively. Note that the places PM_i and AS_p contain one token only, forcing that every machine or assembly station can make one process tasks at the same time.

4.2 Storages

The storages in the FMS are shown in Fig. 5 and classified in three types: a) Manual load-Automatic unload (RM_{ij}), b) Automatic load-Automatic unload (IS_{ij}) and c) Automatic load-Manual unload (FP_{pq}). The load of the storage RM_{ij} requires a manual input transition IN_{ij} that put tokens in the RM_{ij} place, which are extracted by the start of a task $T_a \in A_1$. The storage IS_{ij} is loaded by the end of a task $T_a \in A_2$ and unloaded by the starts of tasks T_b ... $T_c \in A_3$. Finally, the end of a task $T_a \in A_4$ load to FP_{pq} and it is unloaded by a manual transition OUT_{pq} that take tokens of the place FP_{pq}.

4.3 Logical precedence conditions between tasks

Fig. 6a shows a simple PN translation of a D-direct between two tasks where the boxes are tasks (left-side = start and right-side=end). Note that the initial marking is equal to zero and the final of the task T_a enables the start of T_b. Figure 6b shows the case of an inverse precedence, where the continuous line has been changed to a dotted line. Now, the occurrences of the end of the subsequent T_b enable the start of T_a. Note that the tokens in the place of the D-inverse is different to zero, because it is necessary to enable the begin of the task T_a in a first-time of the functional flow.

The logical precedence conditions (direct and inverse) are extended to the conjunction of multiple tasks that enables the start of multiple posterior tasks in Figure 6c and 6d, respectively. It appears, for example in the case of the gathering of transportation of subparts that enables the starting of an assembly task. Note that $K(T_c, IS_{ij})$ is a weight added to each output arc denoting the number of subparts in IS_{ij} needed to start the task S_{pq}. In the inverse case, each place contains initially $K(T_c, IS_{ij})$ tokens representing the amount of necessary subparts for IS_{ij} for each assembly S_{pq}, the input arcs to the places D also have the weight $K(T_c, IS_{ij})$.

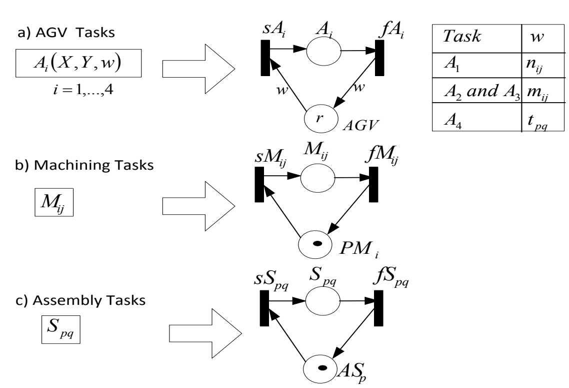

Fig. 4. Models of AGV, Machines and Assembly stations

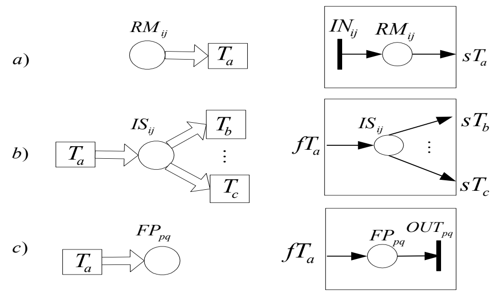

Fig. 5. Storage models

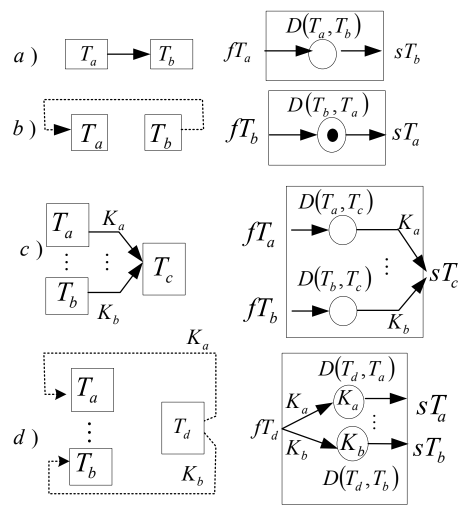

Fig. 6. Logical precedence models

4.4 Example of PN modelling of FMS

Note that the substitution of the blocks of Fig. 3 by the simple PN models described before construct a complete PN of the FMS with the possible concurrence of tasks. For example, let be $r = 4$ AGV's, 1 PM (PM_1) with 2 machine programs and 2 AS (AS_1, AS_2), with one assembly program each one, with $U(S_{11}) = \{IS_{11}, IS_{12}\}$, $U(S_{21}) = \{IS_{11}, IS_{12}\}$ in the quantities of subparts given by $K(S_{11}, IS_{11}) = 3$, $K(S_{11}, IS_{12}) = 1$, $K(S_{21}, IS_{11}) = 1$, $K(S_{21}, IS_{12}) = 2$. The precedence diagram and its translation to PN is given in the figures 7 and 8, respectively.

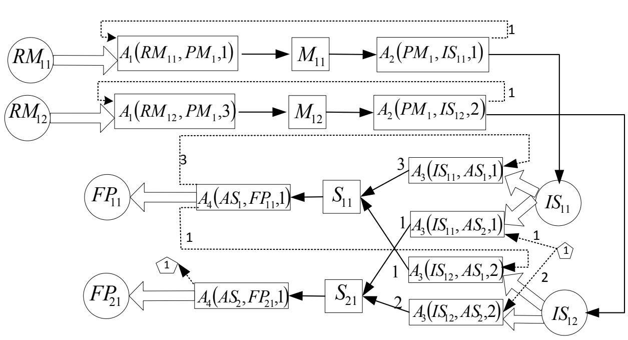

Fig 7. Diagram of precedence restrictions of the example

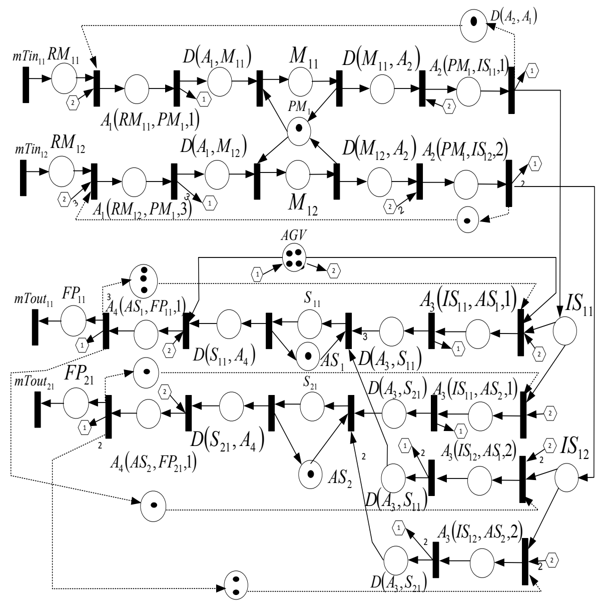

Fig. 8. Complete PN model of the example

5. COORDINATION OF THE AGV'S

In the previous section, the task of transportation alludes to the quantity of AGV's possible points to load or unload work pieces. In this subsection, the low level control laws for the AGV's are described briefly. More complete information and formal proofs are found in our previous work in (Hernandez-Martinez (2011)).

Let $N = \{R_1, \ldots, R_n\}$ be a subset of omnidirectional robots moving in the plane. Note that $n \leq r$, where r is the total of AGV's in the FMS. The kinematic model of each agent or robot R_i, as shown in Figure 10, is described by

$$\dot{z}_i = f_i \in R^2, \quad i = 1, \ldots, n \tag{1}$$

The dynamics of different non-holonomic mobile robots, like unicycle-type robots or car-like robots, can be reduced to the equation (1) using and appropriated control output and applying an input-output linearization as (Hernandez-Martinez (2011)). The main objective of the motion coordination control laws is to design the functions f_i to achieve 1) formation 2) convergence to a point and 3) marching behavior, avoiding the inter-robot collisions.

5.1 Leader-based Formation control with collision avoidance

Let $N_i \subseteq \{z_1, \ldots, z_n\}$, $N_i \neq \emptyset, i = 1, \ldots, n$ denote the subset of positions of the robots which are detectable for R_i, select arbitrarily R_n as the leader robot, then define

$$z_i^* = \frac{1}{n_i} \sum_{j \in N_i} (z + c_{ji}), i = 1, \ldots, n-1 \tag{2}$$

$$z_n^* = \frac{1}{n_n + 1} \left(\sum_{j \in N_n} (z_j + c_{jn}) + \tau \right)$$

as the combination of the desired positions of α_i with respect to the positions of all elements of N_i, where n_i is the cardinality of N_i and the vectors $c_{ji} = [h_{ji}, v_{ji}]^T$ are the desired position of z_i respect to z_j in a particular formation. In the case of the leader, z_n^* includes $\tau \in \Re^2$ that denotes a reference point (position of some storages or workstations) known by the $n-th$ robot only. According to Sanchez (2010), a formation control law with inter-robot collision avoidance is given by

$$f_i = -\frac{1}{2}k\left(\frac{\partial \gamma_i}{\partial z_i}\right) - \frac{1}{2}\eta\left(\frac{\partial V_i}{\partial z_i}\right), i = 1, \dots, n \quad (3)$$

where $k, \eta > 0$, $\gamma_i = \|z_i - z_i^*\|^2$ is and attractive potential function and $V_i = \sum_{j \in M_i}\left(\frac{1}{\|z_i - z_j\|^2} - \frac{1}{d^2}\right) i = 1, \dots, n$ with $M_i = \{z_j \mid \|z_i - z_j\| \le d\}, i = 1, \dots, n$, is a repulsive potential function to avoid inter-robot collisions, where d is the diameter of a circle centered in the coordinate z_i that circumscribes each robot. The control law (3) describes an artificial vector field where the robots are attracted to its desired position and eventually avoid the inter-robot collisions.

5.2 Convergence to a point in the plane

When a robot R_i requires only the convergence to a static point $\beta_i \in \Re^2$, (for example, its home base position), a modification of the control law (3) is reduced to

$$f = -\frac{1}{2}k\left(\frac{\partial \rho_i}{\partial z_i}\right) - \frac{1}{2}\eta\left(\frac{\partial V_i}{\partial z_i}\right), i = 1, \dots, n \quad (4)$$

where $\rho_i = \|z_i - \beta_i\|^2$.

5.3 Marching control

(Hernandez-Martinez (2010)) proposes the next marching control strategy where the leader follows a desired marching path $m(t)$, and the follower robots maintain a rigid formation respect to the leader.

$$f_i = -\frac{1}{2}k\left(\frac{\partial \gamma_i}{\partial z_i}\right) - \frac{1}{2}\eta\left(\frac{\partial V_i}{\partial z_i}\right) + \dot{m}(t), i = 1, \dots, n \quad (5)$$

$$f_n = \dot{m}(t) - k_m(z_n - m(t))$$

where $k_m > 0$ is a gain parameter. Note that the derivative of the marching path must be communicated to all the followers to ensure that the formation errors converge to zero (Hernandez-Martinez (2011)).

5.4 Example of a transportation tasks

To illustrate the use of the control laws (3)-(5), suppose that the robots must realize a transportation task $A_1(RM_{12}, PM_1, 3)$, i.e. three robots work together to move a piece from the raw material storage RM_{12} placed in the workspace coordinate $[-60,30]$ to the process machine PM_1 located in the coordinate $[0,70]$. Assume that $N_1 = \{z_2\}$, $N_2 = \{z_3\}$, $N_3 = \{z_1\}$, i.e. the robots are communicated in a cyclic pursuit configuration (Hernández-Martínez (2011)), and they require to achieve a line-shape formation pattern given by $c_{21} = [0,5.5], c_{32} = [0,5.5]$, $c_{13} = [0,-11]$. Fig. 9 shows a numerical simulation of the three robots doing the task $A_1(RM_{12}, PM_1, 3)$ with $k = 0.2, k_m = 100, \eta = 1 \times 10^7, d = 5$ and $\ell = 1$. The initial positions (home base) of the four robots are $\beta_1 = (-10.5,0), \beta_2 = (-3.5,0), \beta_3 = (3.5,0)$ and $\beta_4 = (10.5,0)$.

For $0 \le t < 250$, three robots are selected (by minimal distance) from home to the position of RM_{12} using the formation control law with collision avoidance (3). For $250 \le t \le 500$, the formed robots uses the marching control law (5), where the marching path is given by the parametric equations for straight line that begins in the position of RM_{12} and ends in the position of PM_1. Finally, for $500 < t \le 750$, the robots has finished the transportation of the work piece and they use the control law (4) breaking formation and returning to their home positions, avoiding again the inter-robot collisions. Fig. 9 shows the posture and orientation of the robots in some time instants.

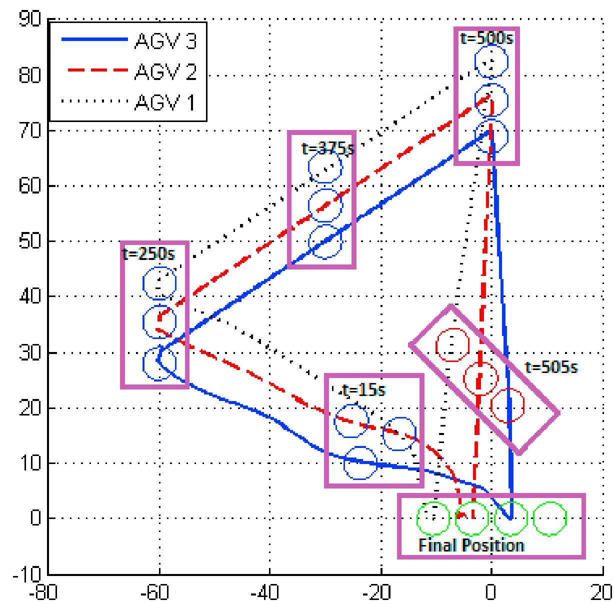

Fig. 9. Example of the motion control laws

6. CONCLUSIONS

This work presents a methodology for the discrete-event modelling of FMS based on the task decomposition proposed by the industrial standard ISA-95 and its translation to generic PN models. Thus, the coordination of the AGV's in the approach is decomposed in two levels. In the high-level, the PN model enables the transportation tasks considering the availability of the robots and the restrictions of the process. When it occurs, the AGV's are selected according to the shortest distance to the initial point of the task. On the other hand, every task in the low-level control implements the continuous control laws for the robots to achieve the desired motion behaviour. Therefore, the task assignment and the convergence to the formation, tracking and collision avoidance are solved in the hybrid architecture at same time. The approach is a systematic method that closes the concepts of Discrete-event systems in an industrial manufacturing context and clarifies the application of continuous motion control laws of AGV's in real FMS. The added value is the scalability of the methodology, which is ready to be easily programmed in industrial supervisory software. In further research, additional behaviours, as fault diagnostics, stochastic time of tasks, delays, etc., will be incorporated to complete the Petri model.

7. ACKNOWLEDGMENTS

This work was partially supported by Universidad Iberoamericana, CONACyT and IPN through research project 20150487, and scholarships EDI, COFAA, and PIFI.

8. REFERENCES

Ailon, A; Zoha, I. (2012), Control Strategies for Driving a Group of Nonholonomic Kinematic Mobile Robots in Formation Along a Time-Parameterized Path, *IEEE/ASME Transactions on Mechatronics*. Vol. 17 , pp. 326 – 336.

Bravo, F; Patiño, D., Melo, K. & Parra, C. (2011), Switching control and modeling of mobile robots formation. In Robotics Symposium, *2011 IEEE IX Latin American and IEEE Colombian Conference on Automatic Control and Industry Applications (LARC)* (pp. 1-6). IEEE.

Cassandras, C.G; Lafortune, S; (2008), *Introduction to discrete event systems*, Kluwer Academic.

Celaya J.R; Desrochers, A; Graves, R.J. (2009), Modeling and Analysis of Multi-agent Systems using Petri Nets, *Journal of Computer,* vol. 4, No. 10 pp. 981-996.

Chen, C; Liu, S; Zhong, C., & Zhang, B. (2014), Multiple robot formation control strategy in corridor environments. In Control Conference (CCC), 2014 33rd Chinese (pp. 8507-8512). IEEE.

Chang, B. L. (2011), A dynamic virtual structure formation control for fixed-wing UAVs , *9th IEEE International Conference on Control and Automation* pp 627 – 632.

D'Andrea, R. (2012), Guest editorial: A revolution in the warehouse: A retrospective on Kiva systems and the grand challenges ahead. *IEEE Transactions on Automation Science and Engineering*, vol. 9, No.4, pp.638-639.

Enright, J.J; Wurman, P.R. (2011), Optimization and Coordinated Autonomy in Mobile Fulfillment Systems, *Automated Action Planning for Autonomous Mobile Robots,* AAAI, pp 33-38.

Fidan, B; Gazi, V; Zhai, S; Cen, N; Karatas, E; (2013), Single View Distance Estimation Based Formation Control of Robotic Swarms, *IEEE Transactions on Industrial Electronics,* vol. 60 no. 12 pp. 5781 - 5791.

Gradišar, D; Mušič, G. (2012), Petri Nets - Manufacturing and Computer Science, *InTech,* edited by Pawel Pawlewski, ISBN 978-953-51-0700-2, pp. 5-26.

Groover S. M.P; (2008), Automation, *Production Systems and Computer Integrated Manufacturing*, Prentice Hall.

Hernandez-Martinez, E. G; Aranda-Bricaire, E; (2011), Convergence and Collision Avoidance in Formation Control: A Survey of the Artificial Potential Functions Approach, Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications, Dr. Faisal Alkhateeb (Ed.), ISBN: 978-953-307-174-9, InTech, DOI: 10.5772/14142.

Hernandez-Martinez, E.G; Aranda-Bricaire, E. (2010), Trajectory Tracking for Groups of Unicycles with Convergence of the Orientation Angles, *Proceedings of the 49th IEEE Conference on Decision and Control*, pp. 6323-6328.

Hernandez-Martinez, E.G; Foyo-Valdes, S.A.; Puga-Velazquez, E.S. & Meda-Campaña, J.A. (2014), Hybrid Architecture for Coordination of AGV's in FMS, *INTECH International Journal of Advanced Robotic Systems*. Vol. 11, pp. 1-12.

Hernandez-Martinez, E.G; Flores Albino, J.M; (2011), Hybrid Architecture of Multi-robot Systems based on formation Control and SOM Neural Networks, *IEEE International Conference on Control Applications* pp. 941-946.

Instrument Society of America, (1999), ISA-95.1. *Enterprise-control system integration*, Part1. Models and terminology, ISA Standards.

Krontiris, A; Luna, R; & Bekris, K. E. (2013, June), From feasibility tests to path planners for multi-agent pathfinding. *In Sixth Annual Symposium on Combinatorial Search*.

Kowdiki,, K.H; Barai, R.K; Bhattacharya, S. (2012), Leader-follower formation control using artificial potential functions: A kinematic approach, *International Conference on Advances in Engineering,* Science and Management pp. 500 – 505.

Li, J; Zhou, M.; Guo, T; Gan, Y; Dai, X. (2014),Robust control reconfiguration of resource allocation systems with Petri nets and integer programming, *Automatica* Volume 50, Issue 3, pp. 915–923.

Sanchez, A; Aranda-Bricaire, E; Hernandez-Martinez, E:G; Magallon, J; Molina, J; (2009), AGV Navigation in Flexible Manufacturing Systems using Formation Control. *Congreso Anual de la Asociación de México de Control Automático* (AMCA).

Sanchez, A; Aranda-Bricaire, E; Jaimes, F; Hernandez E; Nava A; (2010), Synthesis of product-driven coordination controllers for a class of discrete-event manufacturing systems, *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 4, pp. 361-369.

Savkin, A.V; Chao, W ; Baranzadeh, A; Zhiyu, X; (2013), Method for decentralized formation building for unicycle-like mobile robots, *9th Asian Control Conference* (ASCC) , pp. 1 – 5.

Wurman, P.R; D'Andrea, R; Mountz, M. (2008), Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses, *AI Magazine*, Vol 29, No. 1, pp 9-20.

Zhao, W; Tiauw, H. G. (2010), Robust cooperative Leader-follower formation flight control, *11th International Conference on Control Automation Robotics & Vision* , pp. 275 – 280.