Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

A New Algorithm for Translating MLD Systems into PWA Systems

J.L. Villa[1,2], M. Duque[2], A. Gauthier[2] and N. Rakoto-Ravalontsalama[1]
[1]Ecole des Mines de Nantes / IRCCyN
4, rue Alfred Kastler, F-44307 Nantes Cedex 03, France
{jvilla, rakoto}@emn.fr
[2]Universidad de los Andes
Kra 65B No 17A-11, Bogotá D.C., Colombia
{maduque, agauthie}@uniandes.edu.co

Abstract – This paper presents a new algorithm for translating Mixed Logical and Dynamical (MLD) systems into PieceWise Affine (PWA) systems. The presented algorithm uses an enumeration technique and solves several linear programming problems in order to obtain the equivalence. The obtained model is equivalent to the MLD model meaning that given an initial state and an input sequence, the trajectory of the state vector and output vector are the same. The technique is applied to three examples. The computation time and the simulation results for these examples are given.

I. INTRODUCTION

Mixed and Logical Dynamical (MLD) models introduced by Bemporad and Morari in [2] arise as a suitable representation for Hybrid Dynamical Systems (HDS), in particular for solving control-oriented problems. MLD models can be used for solving a model predictive control (MPC) problem of a particular class of HDS and it is proved that MLD models are equivalent to PieceWise Affine Models in [6]. In the paper by Heemels and co-workers, the equivalencies among PieceWise Affine (PWA) Systems, Mixed Logical and Dynamical (MLD) systems, Linear Complementarity (LC) systems, Extended Linear Complementarity (ELC) systems and Max-Min-Plus-Scaling (MMPS) systems are proved, these relations are transcribed here in Fig. 1.

This equivalences are based on some propositions (see [6] for details)

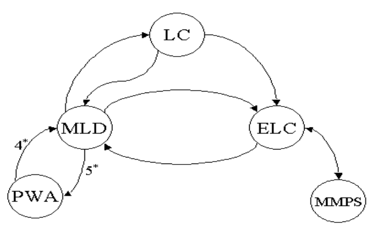


Fig. 1. Equivalence relation between hybrid systems

Every well-posed PWA system can be re-written as an MLD system assuming that the feasible states and inputs are bounded [6, proposition 4*].

A completely well-posed MLD system can be rewritten as a PWA system [6, proposition 5*].

A more formal proof can be found in [3], where an efficient technique for obtaining a PWA representation of a MLD model is proposed.

The technique in [3] describes a methodology for obtaining, in an efficient form, a partition of the state-input space. The algorithm in [3] uses some tools from polytopes theory in order to avoid the enumeration of the all possible combinations of the integer variables contained in the MLD model. However, the technique does not describe the form to obtain a suitable choice of the PWA model, even though this part is introduced in the implementation provided by the author in [4]. The objective of this paper is to propose an algorithm of the suitable choice of the PWA description and use the PWA description for obtaining some analysis and control of Hybrid Dynamical Systems.

II. MLD SYSTEMS AND PWA SYSTEMS

A. Mixed and Logical Dynamical (MLD) Systems

The idea in the MLD framework is to represent logical propositions as equivalent integer expressions. MLD form is obtained by three basic steps [5]. The first step is to associate a binary variable $\delta \in \{0,1\}$ with a proposition S, that may be true or false. δ is 1 if and only if proposition S is true. A composed proposition of elementary propositions $S_1,\ldots,S_q$ combined using the boolean operators like AND(^), OR ($\vee$), NOT(~) may be expressed like integer inequalities over corresponding binary variables δ_i, $i=1,\ldots,q$.

The second step is to replace the products of linear functions and logic variables by a new auxiliary variable $z = \delta a^T x$ where a^T is a constant vector. The z value is obtained by mixed linear inequalities evaluation.

The third step is to describe the dynamical system, binary variables and auxiliary variables in a linear time invariant (LTI) system.

A hybrid system MLD described in general form is represented by (1).

$$\begin{cases} x(k+1) = Ax(k) + B_1u(k) + B_2\delta(k) + B_3z(k) \\ y(k) = Cx(k) + D_1u(k) + D_2\delta(k) + D_3z(k) \\ E_2\delta(k) + E_3z(k) \leq E_1u(k) + E_4x(k) + E_5 \end{cases} \tag{1}$$

where $x = [x_C^T x_l^T] \in \mathbf{R}^{n_c} \times \{0,1\}^{n_l}$ are the continuous and binary states, $u = [u_C^T u_l^T] \in \mathbf{R}^{m_c} \times \{0,1\}^{m_l}$ are the inputs, $y = [y_C^T y_l^T] \in \mathbf{R}^{p_c} \times \{0,1\}^{p_l}$ the outputs, and $\delta \in \{0,1\}^{\eta}$, $z \in \mathbf{R}^{r_c}$, represent the binary and continuous auxiliary variables, respectively. The constraints over state, input, output, z and δ variables are included in the third term in (1).

B. PieceWise Affine Systems

A particular class of hybrid dynamical systems is the system described as follows,

$$\begin{cases} \dot{x} = A_i x + a_i + B_i u \\ y = C_i x + c_i + D_i u \end{cases}, \quad x \times u \in X_i,\ i \in I,\ t \in \mathbb{R}^+ \qquad (2)$$

where $\mathcal{I}$ is a set of indexes, X_i is a sub-space of the real space $\mathbb{R}^n$, and $\mathbb{R}^+$ is the set of positive real numbers including the zero element.

In addition to this equation it is necessary to define the form as the system switches among its several modes. This equation is affine in the state space x and the systems described in this form are called PieceWise Affine Systems (PWA). In the literature of hybrid dynamical systems the systems described by the autonomous version of this representation are called Switched Systems.

If the system vanishes when x brings near to zero, i.e. a_i and b_i are zero, then the representation is called PieceWise Linear (PWL) system.

The discrete-time version of this equation will be used in this work and can be described as follows,

$$\begin{cases} x(k+1) = A_i x(k) + b_i + B_i u(k) \\ y(k) = C_i x(k) + d_i + D_i u(k) \end{cases}, \quad x \times u \in X_i,\ i \in \mathcal{I},\ k \in \mathbf{Z}^+ \qquad (3)$$

where $\mathcal{I}$ is a set of indexes, X_i is a sub-space of the real space $\mathbb{R}^n$.

III. MLD SYSTEMS INTO PWA SYSTEMS

The MLD framework is a powerful structure for representing hybrid systems in an integrated form. Although E_1, E_2, E_3, E_4 and E_5 matrices are, in general, large matrices, they can be obtained automatically. An example is the HYSDEL compiler [10].

However, some analysis of the system with the MLD representation are computationally more expensive with respect to some tools developed for PWA representations. Exploiting the MLD and PWA equivalencies, it is possible to obtain analysis and control of a system using this equivalent representations. Nevertheless, as it is underlined in [3], this procedure is more complex with respect to the PWA into MLD conversion, and there exist more assumptions. To our knowledge, the only previous approach has been proposed by Bemporad [3]. We propose then a new approach of translating MLD into PWA systems.

The MLD structure can be rewritten as follows,

$$\begin{cases} x(k+1) = Ax(k) + \begin{bmatrix} B_{1c} & B_{1l} \end{bmatrix} \begin{bmatrix} u_c(k) \\ u_l(k) \end{bmatrix} + B_2 \delta(k) + B_3 z(k) \\ y(k) = Cx(k) + \begin{bmatrix} D_{1c} & D_{1l} \end{bmatrix} \begin{bmatrix} u_c(k) \\ u_l(k) \end{bmatrix} + D_2 \delta(k) + D_3 z(k) \\ E_2 \delta(k) + E_3 z(k) \le E_2 x(k) + \begin{bmatrix} E_{1c} & E_{1l} \end{bmatrix} \begin{bmatrix} u_c(k) \\ u_l(k) \end{bmatrix} + E_5 \end{cases} \qquad (4)$$

Here, the binary inputs are distinguished from the continuous inputs, because they induce switching modes in the system, in general.

Supposing that the system is well posed, $z(k)$ has only one possible value for a given $x(k)$ and $u(k)$, and can be rewritten as:

$$z(k) = k_1 x(k) + k_2 u_c(k) + k_3 \ |m[x^T, u^T]^T \le b \qquad (5)$$

Replacing this value in the original equations the system can be represented as,

$$\begin{cases} x(k+1) = (A + B_3 k_1) x(k) + (B_{1c} + B_3 k_2) u(k) + B_3 k_3 \\ y(k+1) = (C + D_3 k_1) x(k) + (D_{1c} + D_3 k_2) u(k) + D_3 k_3 \\ (-E_4 + E_3 k_1) x + (-E_1 + E_3 k_2) u \le E_5 - E_2 \delta - E_3 k_3 \end{cases} \qquad (6)$$

If an enumeration technique is used for generating all the feasible binary states of the $[u_l^T\ \delta^T]^T$ vector, the first problem is to find a value of $[x^T\ u^T]^T$ feasible for the problem, that can be obtained solving the linear programming problem,

$$\begin{cases} \min & X = [u^T\ z^T\ x^T]^T \\ s.t. & -E_{1c} u_c + E_3 z - E_4 x \le E_5 - E_2 \delta + E_{1l} u_l \end{cases} \qquad (7)$$

The solution is a feasible value $[x^{*T}\ u^{*T}]^{\mathrm{T}}$. The next problem is to find k_1, k_2 and k_3.

The inequalities can be rewritten as,

$$E_3 z \le E_4 x + E_{1c} u_c + E_{1l} u_l - E_2 \delta + E_5 = \bar{E}_4 \bar{k}_1 x + \bar{E}_{1c} \bar{k}_2 u_c + \bar{E}_5 \bar{k}_3 \qquad (8)$$

where $\bar{E}_5$ includes every constant in the problem, i.e. u_l and δ. On the other hand, the E_3 matrix reflects the interaction among the z variables, and we can write:

$$F \times z \le \bar{k}_1 x + \bar{k}_2 u + \bar{k}_3 \qquad (9)$$

The matrix F represents the interaction among the z variables, if the system is well posed F^{-1} should exist.

With this last equation, for finding $\bar{k}_3$ the next linear programming problem is solved,

$$\begin{cases} \max & \bar{k}_3 \\ s.t. & E_3 \bar{k}_3 \le E_5 - E_2 \delta^* + E_{1l} u_l^* \end{cases} \qquad (10)$$

The solution to this problem is $\bar{k}_3$, in this case we assume that all components in $\bar{E}_5$ are the maximum and minimum values of z and the only solution for the problem is $\bar{k}_3$. With $\bar{k}_3$ we can obtain the other matrices.

For obtaining $\bar{k}_1$ it is necessary to solve n_x, i.e. the length of the state vector, linear programming problems,

$$\begin{cases} \max & \bar{k}_i \\ s.t. & E_3\bar{k}_i \le E_5 - E_2\delta^* + E_{1l}u_l^* + E_{4i} \end{cases} \qquad (11)$$

where E_{4i} represents the column i of the E_4 matrix and $\bar{k}_{1i} = \bar{k}_i - \bar{k}_3$ is the column i of the matrix $\bar{k}_1$.

For obtaining $\bar{k}_2$ it is necessary to solve n_u, i.e. the length of the continuous input vector, linear programming problems,

$$\begin{cases} \max & \bar{k}_i \\ s.t. & E_3\bar{k}_i \le E_5 - E_2\delta^* + E_{1l}u_l^* + E_{1ci} \end{cases} \qquad (12)$$

where E_{1ci} represents the column i of the E_{1c} matrix and $\bar{k}_{2i} = \bar{k}_i - \bar{k}_3$ is the column i of the matrix $\bar{k}_2$.

The matrix F should be found solving n_z, i.e. the length of the z vector, linear programming problems,

$$\begin{cases} \max & \bar{k}_i \\ s.t. & E_3\bar{k}_i \le E_5 - E_2\delta^* + E_{1l}u_l^* + E_{3i} \end{cases} \qquad (13)$$

where E_{3i} represents the column i of the E_3 matrix and $F_i = \bar{k}_i - \bar{k}_3$ is the column i of the matrix F.

Finally, k_1, k_2, and k_3, can be computed as,

$$\begin{cases} k_1 = F^{-1}\bar{k}_1 \\ k_2 = F^{-1}\bar{k}_2 \\ k_3 = F^{-1}\bar{k}_3 \end{cases} \qquad (14)$$

With these equations, the algorithm for translating the MLD model into PWA model is given as follows,

Algorithm 1

1. Find a feasible point for the binary vector, composed by the binary inputs and binary auxiliary variables.
2. Compute $\bar{k}_3$ using Eq. (10).
3. Compute $\bar{k}_1$, $\bar{k}_2$ and F using Eq. (11), (12) and (13).
4. Compute k_1, k_2, and k_3 using Eq. (14).
5. Using Eq. (6), compute A_i, B_i, f_i, C_i, D_i and g_i and the valid region for this representation.
6. If there exists another feasible point go to step 1.
7. End.

Some gains in the algorithm performance can be obtained if the vector z is evaluated after step one, using a linear program for finding the maximum and the minimum in z, if the z_{min} and z_{max} solutions are the same, it is not necessary to calculate steps 3, and 4, and $z = z_{min} = z_{max}$ can be assigned directly.

IV. EXAMPLES

A. *The Three-Tank Benchmark Problem*

The three-tank benchmark problem has been proposed as an interesting hybrid dynamical system. This Benchmark was proposed in [7] and [8]. See [13] and references there in for some control results using MLD framework in this system. The algorithm described in the last section is used for obtaining a PWA representation of this system.

This system has three tanks each of them interconnected with another as depicted in Fig. 2.

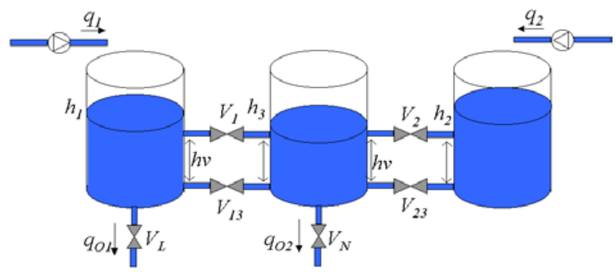


Fig. 2. Three Tank System

The model is written using binary variables (δ_i) and relational expressions,

$$\begin{cases} \delta_1 = 1 \leftrightarrow h_1 > h_v \\ \delta_2 = 1 \leftrightarrow h_2 > h_v \\ \delta_3 = 1 \leftrightarrow h_3 > h_v \end{cases} \quad \begin{cases} Z_{01} = (h_1 - h_v)\delta_1 \\ Z_{02} = (h_2 - h_v)\delta_2 \\ Z_{03} = (h_3 - h_v)\delta_3 \end{cases}$$

$$\begin{cases} Z_1 = (Z_{01} - Z_{03})V_1 \\ Z_2 = (Z_{02} - Z_{03})V_2 \\ Z_{13} = (h_1 - h_3)V_{13} \\ Z_{23} = (h_2 - h_3)V_{23} \end{cases}$$

$$h_1(k+1) = h_1(k) + T_s * \left(\frac{q_1(k)}{C_1} - \frac{h_1(k)}{R_L C_1} - \frac{Z_{13}(k)}{R_{13} C_1} - \frac{Z_1(k)}{R_1 C_1}\right)$$

$$h_2(k+1) = h_2(k) - Ts * \left(\frac{q_2(k)}{C_2} - \frac{Z_{23}(k)}{R_{23} C_2} - \frac{Z_2(k)}{R_2 C_2}\right)$$

$$h_3(k+1) = h_3(k) + Ts * \left(-\frac{h_3(k)}{R_N C_3} + \frac{Z_{13}(k)}{R_{13} C_3} + \frac{Z_{23}(k)}{R_{23} C_2} + \frac{Z_1(k)}{R_1 C_3} + \frac{Z_2(k)}{R_2 C_3}\right)$$

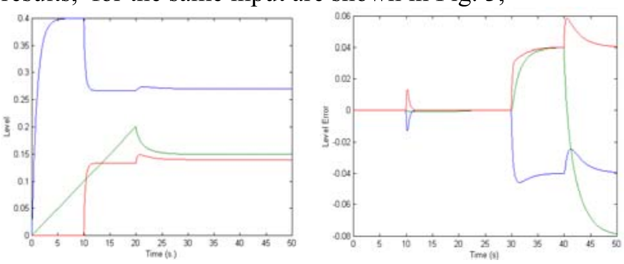
The simulation of the system using the MLD framework and a Mixed Integer Quadratic Programming MIQP algorithm running in an Intel Celeron 2GHz processor and 256MB of RAM was 592.2s, using the PWA representation the same simulation was 1.33s. The time for obtaining the PWA model using the technique described in this work is 72.90s and the algorithm found 128 regions. Using the algorithm in [4] the computation time of the PWA form was 93.88s and the total regions found was 100 and the simulation took 5.89s. These results are summarized in Table I.

Where Computation Time is the time taken by the computer for computing the PWA model based in the MLD model, and Simulation Time is the time taken by the computer for computing a trajectory given a model, an initial state and an input sequence.

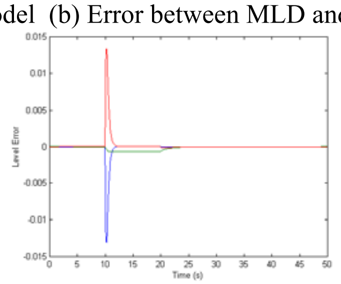
Table I. Computation and Simulation Times

Representation	Computation Time (s.)	Simulation Time (s.)
MLD	-	592.20
PWA-[4]	93.88	5.89
PWA-This work	72.90	1.33

The simulation results with MLD model and the error between PWA simulation results and MLD simulation results, for the same input are shown in Fig. 3,



(a) MLD Model (b) Error between MLD and PWA [4]



(C) Error between MLD and PWA– This Work

Fig. 3. Simulation Results for the Three-Tank System

In this case, at t=30s, the simulation with the PWA system in the Figure 3.b produces a switching to an invalid operation mode.

B. Car with Robotized Manual Gear Shift

The example of a Hybrid Model of a Car with Robotized Manual Gear Shift was reported in [9] and is used in [3] as example. The car dynamics is driven by the following equation,

$$m\ddot{x} = F_e - F_b - \beta\dot{x} \qquad (15)$$

where m is the car mass, $\dot{x}$ and $\ddot{x}$ is the car speed and acceleration, respectively, F_e is the traction force, F_b is the brake force and β is the friction coefficient. The Transmission Kinematics are given by,

$$\omega = \frac{R_g(i)}{k_s}\dot{x}$$

$$F_e = \frac{R_g(i)}{k_s}M$$

where ω is the engine speed, M the engine torque and i is the gear position.

The engine torque M is restricted to belongs between the minimum engine torque $C_e^-(\omega)$ and the maximum engine torque $C_e^+(\omega)$.

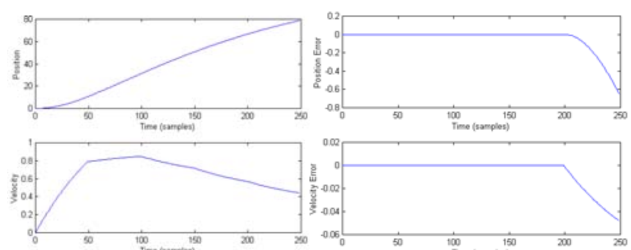
The model has two continuous states, position and velocity of car, two continuous inputs, engine torque and breaking force, and six binary inputs, the gear positions. The MLD model was obtained using the HYSDEL tool.

The translation of the MLD model took 155.73 s and the PWA model found 30 sub-models, using the algorithm proposed in this work, and the PWA model using the algorithm proposed in [3] took 115.52 s and contains 18 sub-models. The simulation time with MLD model and a MIQP algorithm for 250 iterations took 296.25s, using the PWA model obtained with the algorithm proposed here took 0.17s, and using the PWA model obtained using the algorithm in [4] the simulation took 0.35s. These results are summarized in Table II,

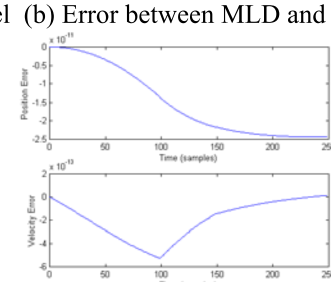
Table II. Computation and Simulation Times

Representation	Computation Time (s.)	Simulation Time (s.)
MLD	-	296.25
PWA-[4]	115.52	0.35
PWA-This work	155.73	0.17

The simulation results with MLD model and the error between PWA simulation results and MLD simulation results, for the same input are shown in Fig. 4,



(a) MLD Model (b) Error between MLD and PWA [4]



(c) Error between MLD and PWA– This Work

Fig. 4. Simulation results for robotized gear shift

C. The Drinking Water Treatment Plant

The example of a Drinking Water Treatment Plant has been reported in [11] and [12]. This plant was modeled using identification techniques for hybrid dynamical systems, and its behavior includes autonomous jumps.

The plant modeled is based in the current operation of drinking water plant Francisco Wiesner situated at the periphery of Bogotá D.C. city (Colombia), which treats on average 12m^3/s. The volume of water produced by this plant is near to 60% of consumption by the Colombian capital. In this plant, there exist two water sources: Chingaza and San Rafael reservoirs which can provide till 22m^3/s of water.

The process mixes inlet water with a chemical solution in order to generate aggregated particles that can be caught in a filter. The dynamic of the filter is governed by the differential pressure across the filter and the outlet water turbidity. An automaton associated to the filter executes a back-washing operation when the filter performance is degraded. Because of process non-linearity, the behavior of the system is different with two water sources, that is the case for the particular plant modeled.

The model for each water source includes a dynamic for the aggregation particle process which dynamical variable is called Streaming Current (SC) and is modeled using two state variables, a dynamic for the differential pressure called Head Loss (HL) with only one state variable, a dynamic for the outlet turbidity (T_o) with two state variables.

The identified model consists of four affine models, two for each water source in normal operation, one model in maintenance operation, one model representing the jump produced at the end of the maintenance operation.

$$\begin{cases} x(k+1) = A_i x(k) + B_i u(k) + f_i \\ y(k) = C_i x(k) + D_i u(k) + g_i \end{cases}$$

$$\begin{aligned} i \in \{ & 1, \text{if water source 1 and normal operation}, \\ & 2, \text{if water source 2 and normal operation}, \\ & 3, \text{if maintenance operation}, \\ & 4, \text{change from maintenance operation} \\ & \quad \text{to normal operation} \} \end{aligned}$$

where *water source* is an input variables, *maintenance operation* is executed if outlet turbidity (T_o) is greater than a predefined threshold, or, Head Loss (HL) is greater than a predefined threshold, or, operation time is greater than a predefined threshold.

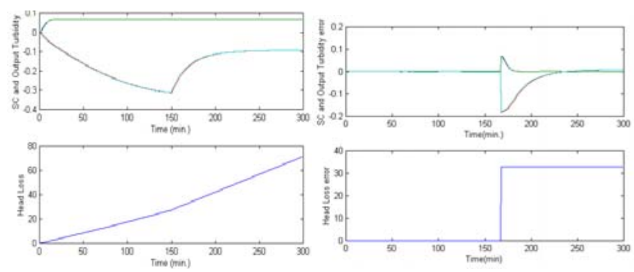
The MLD model has 7 continuous states (including two variables for two timers in the automaton), 4 continuous inputs (dosage, water flow, inlet turbidity and pH), 3 binary inputs (water source, back-washing operation and normal operation), 8 auxiliary binary variables, and 51 auxiliary variables. The complete model can be obtained by mail from the corresponding authors.

The translation from the MLD model into PWA model took 572.19 s, with the algorithm proposed here, generating 127 sub-models. The translation into PWA model took 137.37s, with the algorithm in [3], generating 14 sub-models. The simulation time for 300 iterations with the MLD model and a MIQP algorithm took 4249.301s, the same simulation with the PWA model obtained with the algorithm proposed here took 0.14s, and the same simulation with the PWA model obtained using the algorithm in [4] took 0.31s. These results are summarized in Table III,

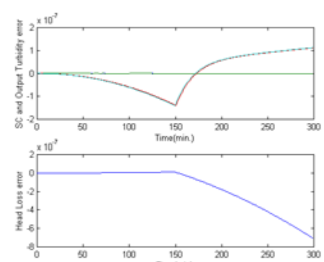
Table III. Computation and Simulation Times.

Representation	Computation Time (s.)	Simulation Time (s.)
MLD	-	4249.30
PWA-[4]	137.37	0.31
PWA-This work	572.20	0.14

The simulation results for the same input are shown in Fig. 5,



(a) MLD Model (b) Error between MLD and PWA [4]



(c) Error between MLD and PWA– This Work

Fig. 5. Simulation results for a water plant model.

In this case, at t=168min, the simulation with the PWA system in the Figure 5.b is not valid because there exist no mode in the PWA representation that belongs to the state-input vector reached in this point. Some other results can be found in [14].

V. CONCLUSIONS

This work presents new algorithm for obtaining a suitable choice of the PWA description from a MLD representation. The results are applied to the three-tank benchmark problem, to a car with robotized gear shift and to a drinking water plant, the three examples have been reported in the literature as examples of hybrid dynamical systems modeled with MLD formalism. The simulation results show that the PWA models obtained have the same behavior with respect to the MLD models. However in some cases the obtained PWA model does not have a valid solution for some state-input sub-spaces.

As a consequence of the enumeration procedure, our PWA models have more submodels/regions than the algorithm in [3], however we show that the procedure does not spent much more computation time because of the simplicity in its formulation, and it ensures the covering of all regions included in the original MLD model.

Ongoing work concerns the analysis of MLD Systems with some results from PWA systems.

VI. ACKNOWLEDGMENT

This work has been supported in part by « Contrat Etat - Région No STIC 9-18036, 2000-2006 », Nantes, France. The authors are thankful to the reviewers for their valuable comments and suggestions.

VII. REFERENCES

[1] R. Alur, C. Courcoubetis, T. A. Henzinger, and P. H. Ho. Hybrid Automata : An Algorithmic Approach To The Specification and Verification of Hybrid Systems. *In Hybrid Systems I*, Lecture Notes in Computer Science 736, Springer-Verlag, pp. 209-229, 1993

[2] A. Bemporad and M. Morari. Control of Systems Integrating Logic, Dynamics, and Constraints. Automatica 35, pp. 407-427, 1999.

[3] A. Bemporad. An Efficient Technique for Translating Mixed Logical Dynamical Systems into PieceWise Affine Systems. Proceedings of the 41st IEEE Conference on Decision and Control. Las Vegas, Nevada USA, December 2002.

[4] A. Bemporad. An Efficient Technique for Translating Mixed Logical Dynamical Systems into PieceWise Affine Systems. Implementation. http://www.dii.unisi.it/~hybrid/tools/mld2pwa

[5] G. Ferrari-Trecate, D. Mignone, D. Castagnoli, and M. Morari. Hybrid Modeling and Control of a Hydroelectric Power Plant. Technical Report AUT00-11, Automatic Control Laboratory, ETH Zurich, Switzerland, 2000.

[6] W.P.M.H. Heemels, B. de Schutter, and A. Bemporad. Equivalence of Hybrid Dynamical Models. Automatica, 37(7), pp. 1085-1091, July 2001.

[7] B. Heiming, J. Lunze Definition of the Three-Tank Benchmark Problem for Controller Reconfiguration. European Control Conference, 1999.

[8] J. Lunze. Benchmark for the Reconfiguration Problem: Laboratory 3 Tank System. Technical University Hamburg-Harbur. 1998. http://www.tu-arburg.de/rts/software/cosy

[9] F.D. Torrisi and A. Bemporad. Discrete-Time Hybrid Modeling and Verification. In *Proc. 40th IEEE Conf. On Decision and Control,* pp. 2899-2904 Orlando, Florida, 2001.

[10] F.D. Torrisi and A. Bemporad. HYSDEL Modeling and Simulation of Hybrid Dynamical Systems. Proceedings of MATHMOD Conference, Vienna University of Technology, Viena, Austria, February 5-7, 2003.

[11] J.L. Villa, M. Duque, A. Gauthier and N. Rakoto-Ravalontsalama Hybrid Modeling of Potable Water Treatment Plant. *In : Cabrera and Cabrera Jr. (Eds.) Pumps, Electromechanical Devices and Systems : Applied to Urban Water Management*. A.A. Balkema Publishers. Lisse, pp. 909-917. 2003.

[12] J.L. Villa, M. Duque, A. Gauthier and N. Rakoto-Ravalontsalama. Modeling and Control of a Water Treatment Plant. In: IEEE International Conference on Systems, Man & Cybernetics (SMC2003) October 5–8, Washington D.C., USA, pp. 171-176 (CD-ROM), 2003.

[13] J.L. Villa, M. Duque, A. Gauthier and N. Rakoto-Ravalontsalama. MLD Control of Hybrid Systems : Application to the Three-Tank Benchmark Problem. In : IEEE International Conference on Systems, Man & Cybernetics (SMC2003) October 5–8, Washington D.C., USA, pp. 666-671 (CD-ROM) 2003.

[14] J.L. Villa, M. Duque, A. Gauthier and N. Rakoto-Ravalontsalama. A New Algorithm for Translating MLD Systems into PWA Systems. Technical Report No 03/8/AUTO Ecole des Mines de Nantes, France. Sept. 2003.