Model Based Fault Diagnosis of Hybrid Systems Based on Hybrid Structure Hypothesis Testing

G. K. FOURLAS*, K. J. KYRIAKOPOULOS and N. J KRIKELIS

Abstract-- **Fault detection and isolation is a challenging task in the control of Hybrid Systems. In this work we focus on the design of a diagnoser for fault diagnosis of Hybrid Systems, in the framework of Hybrid Input Output Automata (HIOA) using hybrid structure hypothesis tests. We present a methodology for detection and isolation of faults using a diagnoser, which combines a set of different hypothesis tests. This approach is applicable to a wide range of systems since Hybrid Systems involve both continuous and discrete dynamics. The states of the Hybrid System model reflect the normal and the failed status of the system components. The faults in our setting are modeled as either discrete or continuous (detrimental) state changes.**

Index Terms-- **Fault detection, fault diagnosis, diagnoser, hybrid systems, hypothesis testing.**

I. Introduction

THE increasing requirements to achieve more reliable performance on complex systems such as air traffic management systems [11, 18], automated highway systems [12, 19], manufacturing systems [3], power systems [8, 10] have necessitated the development of fault diagnosis schemes for accurate diagnosis of system failures. Such systems can be viewed as hybrid systems and therefore fault diagnosis is a challenging task in the control of hybrid systems. Hybrid systems are systems including both continuous and discrete dynamics influencing each other [1], and therefore the global dynamics. The issues of safe operation for such systems are of major importance and require their supervision in order to timely handle the occurrence of faults or failures [15]. In fault detection, we have to answer whether a transition from the normal to a faulted state has occurred.

At this paper, as in our previous contributions [5, 6, 7], we are interested in the problem of failure diagnosis for hybrid systems. We have introduced the notion of diagnosability of hybrid systems presenting a methodology for detection of faults imposing the conditions for a hybrid system to be diagnosable [7]. In this work based on the previous definition of diagnosability and the conditions of diagnosability we proceed with the design and reconstruction of the diagnoser using hybrid structure hypothesis tests.

To perform diagnosis a diagnostic system is required. The basic idea is to use the framework of structured hypothesis tests proposed in [14], with appropriate modifications, to the fault diagnosis of hybrid systems. This new framework called hybrid structured hypothesis test has as the original, no restrictions about any type of faults. Also has a mathematical foundation since it is theoretically grounded in hypothesis testing and mathematical logic. Although we share some ideas with [14] and [16, 17], our approach is different in the sense that it addresses hybrid systems and not discrete event or continuous systems. In this framework both discrete and continuous dynamics are formally described.

In our work the detection system appears in figure 1, and the general structure of diagnoser is shown in figure 3.

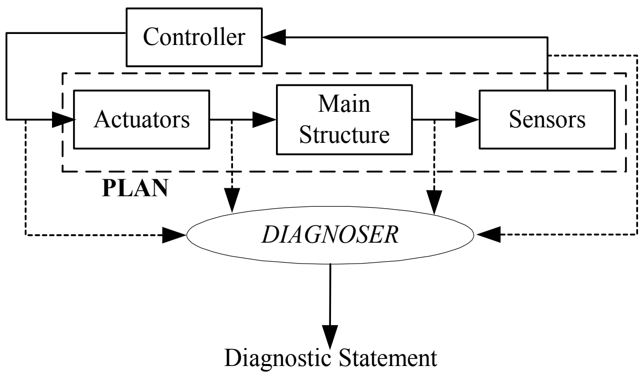

Figure 1. Control and Detection systems

The behavior of the system is modeled by a HIOA (Hybrid Input/Output Automaton) [13] since this is capable of describing both the continuous and the discrete behavior, with modest extensions of the original framework, so as to capture all interesting phenomena.

II. Fault Modeling

A hybrid input output automaton $A = \left(U, X, Y, \Sigma^{in}, \Sigma^{\text{int}}, \Sigma^{out}, \Theta, D, W\right)$ consists of:

– Three disjoint sets U, X and Y of variables, called *input*, *internal* and *output variables*, respectively. We set $V = U \cup X \cup Y$.

– Three disjoint sets Σ^{in}, Σ^{int} and Σ^{out} of actions called *input*, *internal* and *output actions*, respectively. We set $\Sigma = \Sigma^{in} \cup \Sigma^{\text{int}} \cup \Sigma^{out}$.

Manuscript received January 31, 2003.

*Partially supported by the ARCHIMEDES Basic Research Initiative of the Institute of Communication and Computer Systems at NTUA.

Authors are with the Control Systems Laboratory, Mechanical Eng. Dept., National Technical University of Athens (NTUA), Athens 15700, (tel: ++30-210-7723595, e-mail: {gfourlas,kkyria,nkrik}@central.ntua.gr).

– A non-empty set $\Theta \subseteq \boldsymbol{V}$ of *initial states*.

– A set $D \subseteq \boldsymbol{V} \times \Sigma \times \boldsymbol{V}$ of *discrete transitions*.

– A set W of *trajectories* over V.

Due to space limitations, in this paper we only present the fault modeling while the construction of the system model can be found in our previous work [7].

Consider a fault and assume that the same automaton models both the normal and the faulty behavior. We consider that the faults do not affect the system output, i.e. $Y_N = Y_F$ where the subscripts N and F indicate whether the system is normal or faulty. When a fault occurs there is some kind of internal action. This means that $\Sigma^{\text{int}} = \varnothing$ if the main structure operates in normal mode and $\Sigma^{\text{int}} \neq \varnothing$ if the main structure malfunctions.

According to the definition of HIOA the states may change either continuously or discreetly. Thus the variables will evolve either continuously as functions of time or be subject to instantaneous "jumps". The continuous state evolution is modelled by trajectories while the discrete state evolution is representing by the actions.

Consider $s \in \boldsymbol{V}$ a state of the hybrid system. This state can keep evolving continuously, as long as $\forall s_t \in \boldsymbol{V_M},\ s_t \in \omega$ then $s_{t+\Delta t} \in \omega$ where s_t is the state of hybrid system the moment t and Δt is the time interval at which the state evolves continuously at the trajectory ω.

Whenever an input action occurs to the hybrid system its state will either jump to another state or remain to its current state and evolve continuously. The second case will take place whenever the main structure's output variables coincide with the desired ones. In our approach the information about the occurrence of a fault will be provided in the following stages.

A. Continuous Stage

The set W describes the continuous behavior of the HIOA. The information about the fault occurrence from this set will be based on a standard technique of analytical redundancy and more specifically at the new model based diagnosis framework suggested in [14]. According to this method and if disturbances affecting the system are ignored, the system model consists of a plant $G(f_G)$ and a vector valued signal $z(t, f_z)$, where the parameters f_G and f_z are used to describe possible faults. Therefore the set U of input variables is partitioned into two subsets U_N and U_F corresponding respectively to known inputs e.g. control signals and other unknown signals describing faults. Thus we have $U = U_N \cup U_F$.

Likewise the set X of internal variables is partition into two subsets X_N and X_F describing the normally and faulty operation respectively. Thus we have $X = X_N \cup X_F$.

According to the above partition the valuation of the vector $V_F = \begin{bmatrix} U_F & X_F \end{bmatrix}$ is called the *fault state* representing the faulty behavior of the system, while the vector $V_N = \begin{bmatrix} U_N & X_N \end{bmatrix}$ is called the *no fault state* representing the normal behavior of the system. The *fault state space* of V_F will be denoted $\boldsymbol{V_F}$ and the *no fault state space* will be denoted $\boldsymbol{V_N}$. The different faults can be classified into different faulty modes. This classification corresponds to a partition of the fault space $\boldsymbol{V_F}$ into subsets $\boldsymbol{V_{F_i}} \subseteq \boldsymbol{V_F}$ where i is the fault modes. All sets $\boldsymbol{V_{F_i}}$ are pair-wise disjointed which means that only one fault mode can be present at the same time. The fault-free case appertains to $\boldsymbol{V_N}$. Thus the total state space is divided into different subsets as illustrated in fig. 2.

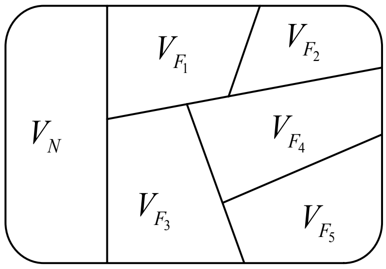

Figure 2. The total state space

The partition of state space can be expressed as a map $\Psi_v : V \to M$ where M is finite set. Then the total state space can be expressed as $\boldsymbol{V} = \boldsymbol{V_N} \cup \boldsymbol{V_F}$.

B. Discrete Stage

The set D determines the discrete evolution of the state. From all news states after the jumping only a certain number of them correspond to the commands and so they represent a normal behavior of the hybrid system. Therefore the set D of discrete transitions is partition into two subsets D_N and D_F respectively for the transitions, which correspond to the normally operation and faulty operation. Then

$$D = D_N \cup D_F$$

The two aforementioned sets are defined as follow:

$$D_N = \bigcup \left\{ (s, \alpha, s') \mid (s, s') \in V_N, \alpha \in \Sigma^{in} \right\} \subset D$$

is the set of transitions for which the hybrid system transits from normal to normal operation, while

$$D_F = \bigcup \left\{ (s, \alpha, s'') \mid s \in V_N, s'' \in V_F, \alpha \in \Sigma^{in} \right\} \subset D$$

is the set of transitions for which the hybrid system transits from normal to fault operation.

Based on earlier definitions the transitions D_F guide the system to the *fault state space* $\boldsymbol{V_F}$. The classification of different faults into fault modes allows us to associate to every subset $\boldsymbol{V_F}$ a transition or a set of transitions of D_F. This means that the transitions can be classified into different *transition types*, each one for each fault mode. Consequently we have a partition of set D_F,

$$D_F = \bigcup_{i \in E} D_{F_i}$$

where E denotes the set of all faults modes.

As we said the hybrid system is modeled as an automaton.

Then the model with a fixed value of $\boldsymbol{V_F}$ or/and transition type D_{F_i} specifies exactly the system situation when a specific fault or no fault is present.

III. Hybrid Structure Hypothesis Test

The problem of fault diagnosis of hybrid systems using hypothesis test can be defined as follow. Suppose that we have a number of possible faults, occurs either to discrete or to continuous dynamics that can be classified into different faulty modes. For each fault we construct a hypothesis test. Then using a certain mechanism we should test and decide about which fault (or fault mode) has occurred.

Due to the hybrid nature of the diagnostic system, there are two different types of faults. Faults which occurs to discrete dynamics and faults to continuous dynamics. According this there are two different kinds of hypothesis tests (their construction is explained formally to subsequent section). First there is a bank of *discrete hypothesis tests* associated to discrete event dynamics. Their objective is to provide the necessary information about the fault occurrence at this level of dynamics. Second there is a bank of *continuous hypothesis tests* corresponded to lower level continuous dynamics of hybrid system. These tests perform the continuous fault diagnosis process.

The hypothesis tests that we use are classical statistical binary test, which means that we make only two mutually disjoint hypothesis.

The hypothesis that the actual fault belong to the set B_i (set of faults modes), the truth of which we test, is called *null hypothesis* and is denoted by H_0.

Any hypothesis which differs from a given hypothesis is called an *alternative hypothesis* and is denoted by H_1.

Next we should decide which fault has occurred. That means that we should conclude if the fault f_a (means the actual fault) belongs to B_i or to B_i^C (where B_i^C is the complement of B_i). The decision if the fault f_a (means the actual fault) belongs to B_i or to B_i^C will be based on testing with a certain criterion if the correspondence hypothesis should be accepted or rejected.

The name hybrid structure is originated from the hybrid nature of the diagnostic system.

A. Construction of the Hybrid Hypothesis Tests

The development of hypothesis tests requires that we have decided the set of hypothesis to testing. This decision will be based to the partition of the state space into different fault modes.

As was aforementioned, due to the hybrid nature of system we construct two different banks of hypothesis tests, one for the discrete behavior and one for the continuous.

1) Discrete behavior

At this level the decision rule which is a function from the observations to diagnosis statement [2] can be written

$$\delta_D : \left(u, \Sigma, y_D\right) \to P\left(E\right)$$

where *D* refers to the discrete part, and $P\left(E\right)$ is the power set of all faults modes.

Each hypothesis test generates a sub-statement $\delta_{D_i} = S_{D_i}$. The final diagnosis statement is produced by combining all the sub-statements. Let f_{D_a} denote the actual fault. Then the two hypothesis, the null and the alternative can be written

$H_{D_i}^0 : f_{D_a} \in B_i$ "some fault mode in B_i can explain the discrete measurements data"

$H_{D_i}^1 : f_{D_a} \in B_i^C$ "no fault mode in B_i can explain the discrete measurements data"

where $B_i \subseteq E$ is a specific set of faults modes and B_i^C is the complement of B_i.

So the decision statement is

$$S_D = \begin{cases} S_{D_i}^1 = B_i^C, if \quad H_{D_i}^1 \quad is \quad accepted (H_{D_i}^0 rejected) \\ S_{D_i}^0 \subseteq E, \; if \quad H_{D_i}^0 \quad is \quad accepted \end{cases}$$

2) Continuous behavior

At this level the decision rule which is a function from the observations to the diagnosis statement can be written

$$\delta_C : \left(u, S_D, y_C\right) \to P\left(E\right)$$

where *C* refers to continuous dynamics, and $P\left(E\right)$ is the power set of all faults modes.

Each hypothesis test generates a sub-statement $\delta_{C_i} = S_{C_i}$. The final diagnosis statement is produced by combining all the sub-statements.

Let f_{C_a} denote the actual fault. Then the two hypothesis, the null and the alternative can be written

$H_{C_j}^0 : f_{C_a} \in B_j$ "some fault mode in B_j can explain the continuous measurements data"

$H_{C_j}^1 : f_{C_a} \in B_j^C$ "no fault mode in B_j can explain the continuous measurements data"

where $B_j \subseteq E$ is a specific set of faults modes and B_j^C is the complement of B_j. So the decision statement is

$$S_C = \begin{cases} S_{C_j}^1 = B_j^C, if \quad H_{C_j}^1 \quad is \quad accepted (H_{C_j}^0 rejected) \\ S_{C_j}^0 \subseteq E, \; if \quad H_{C_j}^0 \quad is \quad accepted \end{cases}$$

IV. Construction of Test Quantities

Until now we have discus the construction of hybrid structure hypothesis tests. For each one of individual hypothesis test we must find a region of rejection, which means a subset of all possible observations where the null hypothesis is rejected. For this purpose usually is used a test quantity $C_D\left(x_D, \Sigma, y_D\right)$ (or $C_C\left(x_C, S_D, y_C\right)$) which is a function of the observations to a scalar value, respectively for the discrete and continuous case. This value is to be compared with a threshold T_{D_i} or T_{C_j}.

Generally the construction of the test quantity is based to the following principle according [14]: *The test quantity* $C_D(x_D,\Sigma,y_D)$ (or $C_C(x_C,S_D,y_C)$) *should be small if the data matches any of the system models and large otherwise.*

In other words the test quantity can be seen as a measure of the validity of some models.

How the test quantities are designed depends on the actual case. There exist several principles that can be used such as the *prediction principle*, the *estimate principle*, and the *likelihood principle*.

In our work we will use *residual generation* (r) [9], which is a special case of the prediction principle. Then the hypothesis tests can be written for the discrete and the continuous case respectively

$$f_{D_a} \in \begin{cases} B_i^C, & if \quad r_{D_i}(x_D,\Sigma,y_D) \geq T_{D_i} \\ E, & if \quad r_{D_i}(x_D,\Sigma,y_D) < T_{D_i} \end{cases}$$

$$f_{C_a} \in \begin{cases} B_j^C, & if \quad r_{c_j}(x_C,S_D,y_C) \geq T_{C_i} \\ E, & if \quad r_{c_j}(x_C,S_D,y_C) < T_{C_i} \end{cases}$$

where $r_{D_i} = C_{D_i}$ and $r_{C_i} = C_{C_i}$.

V. Diagnoser Design

The diagnoser is a hybrid automaton that generates a signal whenever a fault occurs. Its role is to observe and check the behavior of the system automaton (and to compare its evolution with the predefined acceptable behavior). Moreover, whenever it detects a fault it should generates a diagnostic statement *S*, indicating the malfunctioning component, and contains information about which fault mode can explain the behavior of the process.

The diagnoser is considered to be *passive*, which means that does not affect the system to be diagnosed. Also it is assumed to be *static* that is, the same observations will always give the same diagnostic result (statement).

Inputs of diagnoser are the signals $u(t)$, Σ and $y(t)$, and output the signal of decision statement. In terms of decision theory the diagnosis system is a decision rule, where the output (decision statement) is a function of the diagnoser inputs.

The diagnoser consists from three parts: the discrete diagnoser, the continuous diagnoser and the mechanism of decision logic. The discrete component offers estimation for the discrete state of hybrid system as well as the fault diagnosis at the level of discrete event. The continuous part provides the fault diagnosis of continuous behavior of the hybrid system. Least the decision logic mechanism generates the fault statement.

A. Discrete part of Diagnoser

The discrete part of diagnoser is served for two reasons. First it offers the diagnosis of fault at the level of discrete event, and second it estimates the discrete state of the hybrid system. This operation is executed by the following steps:

1. Upon the occurrence of an action the diagnoser compute all the possible next discrete transitions of the hybrid system, given by $D = \{(s,\alpha,s') \mid (s,s') \in V, \alpha \in \Sigma\}$
2. For each one of the discrete transitions is associated the corresponded guard [7], and the transition is enable when the guard is satisfied.
3. For each transition is checked if the corresponding guard belongs to the set of measurements guards.
4. For each one of the discrete transitions performed hypothesis test and generates the corresponding sub-diagnosis statement.

B. Continuous part of Diagnoser

The continuous diagnoser is a system whose dynamics depends on the current state of the hybrid system and it serves to provide the diagnosis of faults to the level of continuous dynamics. We assume that the initial state of the system is known and the diagnoser function in parallel with the system from the start of operation.

In this paper we restrict our attention to linear hybrid system. Then the state space form representation is

$$\dot{x}(t) = A_i x(t) + B_i\left(u(t) + f_a(t)\right) + H_i f_c(t) + E_i d(t)$$

$$y(t) = C_i x(t) + D_i u(t) + f_s(t)$$

where $A_i \in R^{n\times n}, B_i \in R^{n\times m}, C_i \in R^{p\times n}$ depending on the current discrete state q_i of the plant and $f_a(t)$ denotes actuator faults, $f_c(t)$ component faults, $f_s(t)$ sensor faults and $d(t)$ disturbances acting upon the system. H_i and E_i are the distribution matrices for $f_c(t)$ and $d(t)$ depending also on the current discrete state q_i.

Definition: *A continuous diagnoser is a system that takes process input and output signals and discrete diagnoser output as inputs and generates sub-diagnosis statement.*

When a discrete transition takes place and the new discrete state is estimate by the discrete part of diagnoser the corresponded set of hypothesis test is energized (triggered). Therefore the discrete state is a function of the discrete diagnoser output, $\varphi : S_D \to \boldsymbol{V}$.

To each discrete state all possible faults are determined and thus we have a number of hypothesis tests corresponding to all possible faults of that state.

As test quantities are used the residuals designed for each fault. Then each hypothesis test generates the corresponding sub-diagnosis statement, feeding the decision logic.

C. Decision Logic

The decision logic unit is used to produce the final diagnosis statement which expressed as $S = \bigcap_i S_{D_i} \bigcap_j S_{C_j}$ i.e. is a combination of discrete and continuous sub-statements.

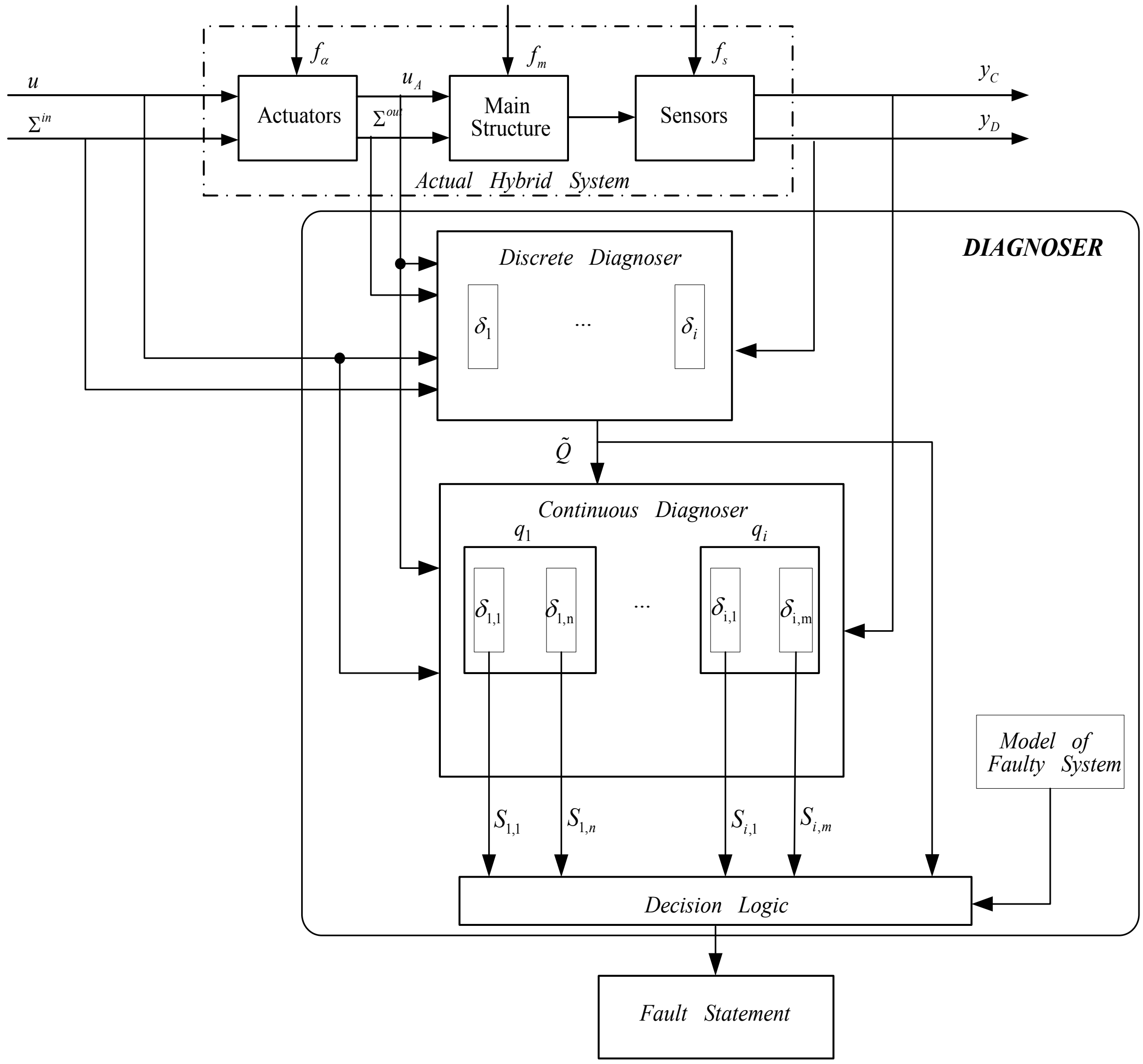

Figure 3. General structure of Diagnoser

VI. Application to an Electric Power Transmission System

Power systems often exhibit complex behavior in response to large disturbances. Such behavior is characterized by interactions between continuous dynamics and discrete events. Components such as loads drive the continuous dynamic while others components such as protection devices exhibit event-driven discrete dynamics. Therefore power systems are an important example of hybrid systems. In our example (Fig. 4) a simple power system consisting of a voltage source, two-transmission lines equipped with inverse time relay overcurrent protection and one load is built and simulated (Fig. 2) with, the Simulink™ environment, and the Stateflow™, all running on top of Matlab™.

The hybrid behavior of this system is due:

- to the close/open position of the circuit breakers (CB)
- to continuous dynamics of the inverse time relay and the load.

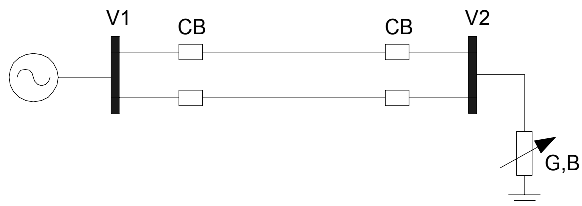

Figure 4: Single line diagram

A power system is considered to exhibit different states, which are *normal state*, *emergency state* and *restorative state* [4]. When the system starts functioning is in its normal mode. The load dynamics are represented by a ramp and because of line characteristics (Table I) the second line is overloaded first. The inverse time relay integrates the overcurrent value and when the output of the integrator becomes positive a trip signal is given. The CB is considered a perfect switch instantaneously interrupting current. As a consequence the second line is turned off and therefore all the current passes through the first line which is in turn overloaded. The overload of line 1 results (after approximately 10sec) is an overcurrent tripping signal given by its relay and a blackout occurs (Fig. 6).

The diagnoser is a hybrid automaton that generates a signal whenever a significant change occurs. Its role is to observe and check the behavior of the system automaton (and to compare its evolution with the predefined acceptable behavior). Moreover whenever the system is found in any state the diagnoser generate an ALARM signal, indicating the malfunctioning of the power system (Fig. 5).

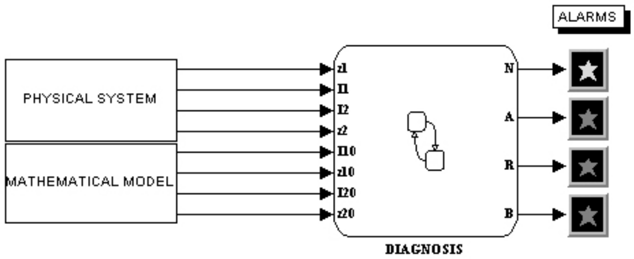

Figure 5: Diagnosis Simulink model

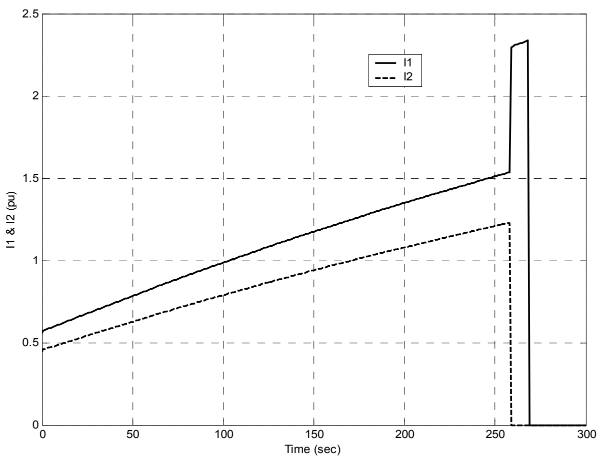

Figure 6: System response

TABLE I
LINE DATA

Line	Ampacity (pu)	Resistance (pu)	Reactance (pu)
1	1.5	0.05	0.2
2	1.0	0.0625	0.25

VII. CONCLUSION

The handling of faults for large-scale systems is one of the major problems faced by control engineers today. The most significant challenge arises from the complexity of the system, which forces designers to develop more sophisticated diagnosis schemes. In this work we concentrated on the design and reconstruction of a diagnoser using hybrid structure hypothesis tests, which can be used for fault diagnosis of hybrid systems. This approach was illustrated via a simple application to an electric power transmission system.

Our current directions include the study of multiple faults occurring at different system components as well as the interaction of different faults.

ACKNOWLEDGMENT

The authors want to thank Prof. K.Vournas of the Electrical & Computer Eng. Dept. at NTUA for his help with the Electric Power model.

REFERENCES

[1] M.S. Branicky, Studies in Hybrid Systems: Modeling, Analysis and Control, PhD thesis, Massachusetts Institute of Technology, Dept. of Electrical Eng. and Computer Science, June 1995.
[2] L.O. Berger, Statistical Decision Theory and Bayesian Analysis, Springer Verlag. 1985.
[3] C. G. Cassandras D.L. Pepyne, "Optimal control of a class of hybrid systems", in *IEEE Conference on Decision and Control*, San Diego, California, pp. 133-138, December 1997.
[4] O.I. Elgerd, "Electric Energy Systems Theory", 1982, McGraw-Hill.
[5] G.K. Fourlas, K.J. Kyriakopoulos, N.J. Krikelis, "Contribution to the Fault Detection for Hybrid Systems", *Proceedings of the 8th IEEE Mediterranean Conference on Control and Automation*, Rio, Patras, Greece, July 2000.
[6] G.K. Fourlas, K.J. Kyriakopoulos, N.J. Krikelis, "A Framework for Fault Detection of Hybrid Systems", *Proceedings of the 9th IEEE Mediterranean Conference on Control and Automation*, Dubrovnik, Croatia, 2001.
[7] G.K. Fourlas, K.J. Kyriakopoulos, N.J. Krikelis, "Diagnosability of Hybrid Systems", *Proceedings of the 10th IEEE Mediterranean Conference on Control and Automation*, Lisbon, Portugal, 2002a.
[8] G.K. Fourlas, K.D. Vournas, K.J. Kyriakopoulos, "Hybrid Systems Modeling for Power Systems", *Proceedings of the 3rd IEE Mediterranean Conference on Power Generation*, Athens, Greece, 2002b.
[9] P.M. Frank, "Fault Diagnosis in dynamic Systems Using Analytical and Knowledge-based Redundancy, A Survey and Some New Results", *Automatica*, vol. 26, no. 3, 1990, pp. 459-474.
[10] I.A. Hiskens, M.A. Pai, "Hybrid systems view of power system modeling" ISCAS 2000, May 28-31, 2000, Geneva, Switzerland.
[11] J. Lygeros, G. J. Pappas, S. Sastry, "An approach to the verification of the Center-TRACON Automation System", in *Hybrid Systems: Computation and Control*, vol. 1386 of LNCS, 1998, pp. 289-304, Springer Verlag.
[12] J. Lygeros, D.N. Godbole, S. Sastry, "Verified hybrid controllers for automated vehicles", *IEEE Transactions on Automatic Control*, 43(4) pp. 522-539, April 1998.
[13] N. Lynch, R. Segala, F. Vaandrager, H. Weinberg, "Hybrid I/O automata", *Hybrid Systems III*, no. 1066 in LNCS, 1996, pp. 496-510, Springer Verlag.
[14] M. Nyberg, Model Based Fault Diagnosis: Methods, Theory and Automotive Engine Applications, PhD thesis, Department of Electrical Engineering, Linkoping Univ., Linkoping, Sweden, 1999.
[15] R. Patton, P. Frank, R. Clark, "Fault Diagnosis in Dynamic Systems – Theory and Application", 1989, Prentice Hall.
[16] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, D.C. Teneketzis, "Failure Diagnosis using discrete-event models", *Trans. On Control System Technology*, vol. 4, no.2 pp.105-124, 1996.
[17] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, D.C. Teneketzis, "Diagnosability of Discrete-Event Systems", *Trans. On Control System Technology*, vol. 40, no.9 pp.1555-1575, 1995.
[18] C. Tomlin, G. J. Pappas, S. Sastry, "Conflict resolution for air traffic management: A study in multi-agent hybrid systems", *IEEE Transactions on Automatic Control*, 42(4) pp. 509-521, April 1998.
[19] P. Varaiya, "Smart cars on smart roads: problems of control", *IEEE Transactions on Automatic Control*, 38(2) pp. 195-207, 1993G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *Phil. Trans. Roy. Soc.* London, vol. A247, pp. 529-551, Apr. 1955.