Journal of the Nigerian Association of Mathematical Physics
Volume **19 (November, 2011), pp 69 – 76**
© *J. of NAMP*

Reliability modelling and simulation of switched linear system control using temporal databases

C. N. Njoku

Dept. of Computer Science
Federal University of Technology, Owerri, Nigeria.

Abstract

In reliability modelling with redundancies, it is assumed that the system would have sets of failed, standby, and working subsystems at any time. This study deals with the case of stochastic switched linear system comprising series-parallel interconnected subsystems for which operational safety conditions are specified. The subsystems are assumed to admit the same control inputs and undergo the same state transitions. Thus, constructing a subsystem Markov model and matching its parameters with the specified safety factors provides the basis for the entire system analysis. For the system simulation, temporal databases and predictive control algorithm are designed. The simulation results are analyzed to assess the reliability of the system behaviour. Graphs of the system behaviour indicate cases of highly oscillatory and fairly stable trajectories from which optimal-time controls are deduced. The study demonstrates use of temporal databases in practical analysis of such stochastic system, highlighting the dynamics beyond results of theoretical analysis. The study has applications, which include optimal design of fault-tolerant real-time switching systems control and modelling embedded micro-schedulers for complex systems maintenance.

Keywords: Switched linear system, Markov model, predictive control, reliability, safety factors.

1.0 Introduction

Reliable switching control problems occur in the design of many complex physical artefacts. The switching devices may be electronic, hydraulic, mechanical, etc. Reliability modelling of switching control of some complex systems assumes that at any time, there are sets of failed, standby, and working subsystems. As the system dynamics vary with switching instants, control rules dealing with subsystems states and timed switching sequences are designed to regulate the system behaviour [1–4]. This study deals with reliability modelling and simulation of an intuitively specified stochastic switched linear system control using temporal databases. The major objective is to determine if the specifications would result to reliable, optimal-time control, and stable system trajectories.

1.1 Previous study

In a previous study reported in [5], a strategy board game called Jonda was redefined and used as experimental object in switched linear system identification. The aim was to explore the feasibility of non-classical methods (namely, game-theoretic strategies and neural computing) in system identification from first principles. The study resulted to construction of mathematical models of the system and its controller, estimation of parameters and identification of stabilizing switching control rules. A simulation test that considered few fixed temporal parameters of the identified system was carried out to assess its behaviour, which was found to be reasonable.

With the results of the previous study, the present study adopts a more pragmatic approach to the reliability modelling based on detailed system specifications. Significant in the approach are the construction of subsystem Markov model, temporal databases, enhanced switching control rules, and extensive simulation tests aimed at determining the regions of reliable control.

1.2 Basic state space model of a discrete-time switched linear system

As our study is concerned with a discrete-time system, we define the basic model equations as follows:

$$x(t+1) = A(t)\,x(t) + B(t)u(t) \qquad (1.1)$$

$$y(t) = C(t)x(t) + D(t)u(t) \qquad (1.2)$$

Corresponding authors: E-mail: cnnodimnjoku@yahoo.com, Tel. +2348059910428 or +2348060965934

where $x \in R^n$ is the state variable vector, u is the forcing input, $y \in R^n$ is the system output vector or configuration, A, B, C, and D are the switching state, input coefficient, output, and gain/feed-through matrices, respectively.

For a discrete-time transitive system, the switching behaviour can be described as a 5-tuple:

$$S = (x, q, u, \Phi, t) \tag{1.3}$$

where x is a set of state variables, q is the set of switching modes, u is set of control inputs, t is the switching time, and Φ is the switching control function or rule. Thus, for the switching rule, we may write

$$q(t+1) = \Phi(x(t), q(t), u(t)) \tag{1.4}$$

Usually, the switching control function specification depends on the control performance criteria such as stability, reliability, minimum trajectory tracking error, etc. Thus, equation (1.4) can be used to generate and test switching sequences that meet desired performance criteria. We note that for switching control functions that study system stability or reliability, the switched system can be viewed as a special case of a time-varying linear system [6 – 8].

2.0 System specifications and statement of the problem

2.1 System description

Consider a discrete-time switched linear system with subsystems (or nodes) arranged as shown in Figure 2.1. This figure corresponds to the board structure of a tic-tac-toe game called Jonda, which we have developed for advanced learning tool in control system theory and computer science [5,9]. In this abstracted case, Figure 2.1 represents control panel display of a switched linear system.

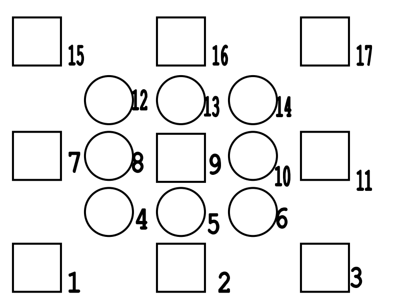

Figure 2.1: Sample switched linear system layout

In the figure, each horizontal, vertical, and diagonal row is a set of operationally related nodes. The square nodes have greater operational capacity than the circular nodes. In terms of operational safety, the critical sets are: failed-nodes $F(f_1, f_2,\ldots,f_k)$, standby nodes $S(s_1, s_2,\ldots,s_m)$, and working (active) nodes $W(w_1, w_2, \ldots, w_n)$. The nodes are transitive elements with the relations T(S, W), T(W, F), etc, meaning transitions from standby to working, working to fail, respectively. Working node failure occurs randomly, while other node transitions may be due to human operator or controller actions. The nodes are assumed to admit the same set of control inputs and all node state transitions are to be automatically recognized by the controller.

2.2 System specifications

For the reliability modelling with redundancies, appropriate failed-standby-working ratio policy (e.g. 4-4-9) is to be adopted. The node temporal parameters are defined as downtime t_d, standby time t_s, working time t_w, and the maximum and minimum values as t_{dmax}, t_{smin}, and t_{wmax}. The system safety conditions that depend on the sets of node states and temporal parameters are specified as presented in Table 2.1.

Table 2.1: System safety condition and temporal specifications

S/N	System safety condition	Node states and explanations	Temporal relations
1	Fatal system failure	Complete row failure	With all $t_d < t_{dmax}$
2	Many row-failure chance	For 3-node row: W = 1, F = 2; For 5-node rows: W = 1, F = 4.	$t_w \le t_{wmax}$, and there are some $t_d < t_{dmax}$.
3	One row-failure chance	One row with W defined as for many row-failure above	At least one F node with $t_d \ge t_{dmax}$, and/or one W node with $t_w < t_{wmax}$.
4	Working square and circle nodes ratio	The ratio N_{sq}/N_{cir} should be in the range of 0.50 …1.00	The case of $N_{sq}/N_{cir} < 0.50$ is tolerable if some N_{sq} $(t_d) \ge t_{dmax}$ or N_{sq} $(t_s) \ge t_{smin}$,

5	Minimal safe configuration	For 3-node row: $W \geq 1$, and $S \geq 1$. For 5-node rows: $W \geq 2$ and $S \geq 1$.	For some W and S nodes $t_w < t_{wmax}$, and $t_s \geq t_{rmin}$
6	Optimal safe configuration	For 3-node row: $W \geq 2$. For 5-node rows: $W \geq 3$.	$t_w < t_{wmax}$ for most W nodes; $t_d \geq t_{dmax}$ for most F nodes; $t_s \geq t_{smin}$ for most S nodes.

In particular, fatal failure is considered as an undesirable incident whereby the controller shuts down system without options for operator's intervention. Single or multiple row failure chance is a condition whereby the system may continue to operate for some time interval within which stabilization might be possible. We note that some of the temporal relations imply specifications for delay in state or control in the system. The switching controller is to switch nodes to maintain safe and reliable operation by checking immediate and near-future system failure chances. The entire system is required to operate safely for a reasonable time horizon before failure.

2.3 Statement of the problem

With the system description and specifications, the problem to be addressed is summarized as follows:

(1) Develop a stochastic simulation model of the abstracted switching control system for testing the workability and reliability of its specifications;

(2) Construct appropriate temporal databases and control algorithm to automate the model simulation;

(3) Evaluate the simulation results to ascertain the operational reliability in terms of optimal-time control and stable system trajectories.

3.0 System modelling

3.1 Subsystem Markov model state transition diagram and temporal relations

Each subsystem (or node) in our system is a transitive element that may be in failed (F), standby (S), or working (W) state. The state transition diagram is as presented in Figure 3.1.

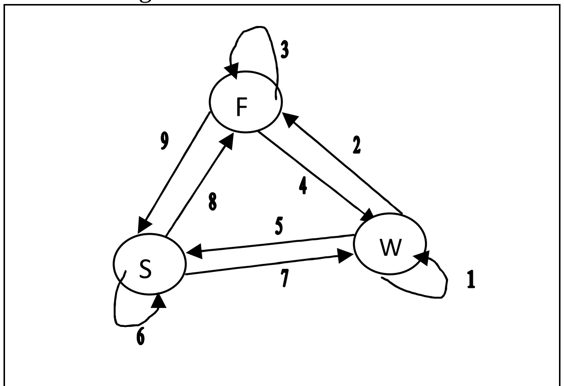

Figure 3.1: Node state transition diagram

The node links 4 and 8 need further explanation. In practice, link 4 may be due to human operator corrective action for minor faults of short duration. For instance, the operator may temporarily stop a working node, quickly make some adjustments, and switch it back to working mode within the specified limit of delayed control. Further, suppose a failed node is completely replaced by a new one, it may also be switched to working mode immediately, leaving the controller with system stabilization. Link 8 is for conditions when standby node fails while being switching to working mode. The transition links and temporal relations are as explained briefly in Table 3.1, while Table 3.2 shows the possible node transitions and controller actions. It is assumed that within the maximum downtime t_{dmax}, a failed node must have been restored to normalcy and can be switched to standby or working mode.

Table 3.1: Node state transitions and temporal relations

Node Link	Explanations	Temporal relations
1	Continuous working mode	$t_w < t_{wmax}$
2	Working node failure or operator action	Random incidental fault
3	Downtime duration	$t_d \leq t_{dmax}$
4	Repaired F node switched to W node	$t_d \geq t_{dmax}$
5	Working node switched to rest	$t_w \geq t_{wmax}$
6	Node standby or rest duration	$t_s \geq t_{smin}$
7	Switching from S-node to W node	$T_s \geq t_{smin}$
8	Faulty S node switched to F	Occasional S node failure
9	Repaired F node switched to S node	$t_d \geq t_{dmax}$

Table 3.2: Effects of node state transitions and controller actions		
Node Transitions	**Possible effects**	**Possible controller actions**
$T_1(W, F)$	Instability due to node breakdown	Switch nodes using T_2, or T_5
$T_2(F, W)$	Repairs increases W nodes	Perform T_6
$T_3(F, S)$	Repairs increase S nodes	Perform T_5
$T_4(S, F)$	Failed S nodes increases F nodes	Perform T_3 and T_5
$T_5(S, W)$	Switched S nodes increases W nodes	Perform T_3 or T_6
$T_6(W, S)$	Switched W nodes increases S nodes	Perform T_2 or T_5

3.2 The Markov model parameters

The applications of Markov models include studies of system performance improvement, and stochastic transition system modelling such as complex queuing systems [10,11]. Some of the parameters associated with a Markov model are state transition probabilities ρ_{ij}, equilibrium probabilities P_1, P_2, …, P_n, mean time duration of each state, mean recurrent time of a recurrent state, etc. While the other parameters are calculated, the state transitions probabilities are determined from analysis of the system's past operational data or the model builder may assign the empirical values based on preferences, intuition, game-theoretic payoffs, or technical feasibilities [7,12–15].

For the case under consideration, empirical values are assigned to the state transition probabilities and are related to the temporal safety factors of the system. Table 3.3 shows a sample of the possible state transition probability matrix assigned after some trial sampling, and what may be considered as worst–case values.

Table 3.3: Sample node state transition probability matrix			
Node state	**F**	**S**	**W**
F	0.25	0.60	0.15
S	0.05	0.25	0.70
W	0.20	0.20	0.60

Let the equilibrium probabilities for F, S, and W node states be P_1, P_2, and P_3, respectively. Then, these probabilities are determined by solving the simultaneous equations

$$\begin{pmatrix} 0.25 & 0.05 & 0.20 \\ 0.60 & 0.25 & 0.20 \\ 0.15 & 0.70 & 0.60 \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \\ P_3 \end{pmatrix} = \begin{pmatrix} P_1 \\ P_2 \\ P_3 \end{pmatrix} \qquad (3.1)$$

and $$P_1 + P_2 + P_3 = 1 \qquad (3.2)$$

The solution of the equations gives $P_1 = 0.166$, $P_2 = 0.281$, and $P_3 = 0.553$. This means that at equilibrium, 16.6% of a node's productive time is spent in failure mode, 28.1% in standby mode, and 55.3% in active work. Note that these are due to the assigned worst–case values in Table 3.3.

Since the equilibrium probabilities are constants, they can be related with the temporal parameter ratios: t_d/t_{dmax} t_s/t_{smin}, and t_w/t_{wmax}. Thus, for simulation of the system steady state behaviour, we may write the rules covering switching mode transitions T, as:

$$\text{If } (t_d/t_{dmax} \geq P_1) \text{ then } T(x_d, x_s) \qquad (3.3a)$$
$$\text{If } (t_s/t_{smin} \geq P_2) \text{ then } T(x_s, x_w) \qquad (3.3b)$$
$$\text{If } (t_w/t_{wmax} \geq P_3) \text{ then } T(x_w, x_s) \qquad (3.3c)$$

In a situation where a standby subsystem may be switched to maintenance state (M) if, considering the total working time Σt_w, it is due for maintenance, we may write

$$\text{If } (\Sigma t_w/t_{maintdue} \geq 1.0) \text{ then } T(x_s, x_m) \qquad (3.3d)$$

Equations (3.3a) – (3.3d) correspond to some part of the switching rule defined by equation (1.4). The above basic rules have to be combined with other records of the temporal databases and system safety factors to design the switching control algorithm [16– 19].

The meant time duration of the node states can be computed to determine the temporal parameters t_{dmax}, t_{smin}, and t_{wmax} as follows:

$$N_{dur} = t/(1.0 - \rho_{jj}) \qquad (3.4)$$

where t is the time it takes to enter a state, and ρ_{jj} is the diagonal elements of the state transition probability matrix defined in Table 3.3. Hence $t_{dmax} = t_{smin} = 1.333$ unit time, $t_{wmax} = 2.5$ unit time, which may be seconds, minutes, hours, etc.

Further, the mean recurrent times t_{rj} of the F, S, and W node states are calculated using the following formula (where t_{rj} is the mean time between successive visits to a recurrent state):

$$t_{rj} = 1.0/P_j \qquad (3.5)$$

where P_j is the equilibrium probability of the j^{th} state. Hence, for the F, S, and W node states we have $F(t_{rj}) = 1.0/0.166 = 6.0241$; $S(t_{rj}) = 1.0/0.281 = 3.5587$; $W(t_{rj}) = 1.0/0.553 = 1.8083$.

3.3 System temporal databases and predictive switching control rules

In section 3.2, our calculations have been on Markov model of a single subsystem. This section deals with aggregation of subsystem parameters to construct the entire system temporal databases and specify its other features.

Three complementary temporal databases are required for the system simulation. These are described as follows:

(1) ***System static database***. This contains basic ground facts about the system; the file structure is defined as
SystemStaticData(Fmax, Smax, Wmax, Tdmax, Tsmin; Twmax, Tmaintdue)
Fmax, Smax, and Wmax are maximum allowable F, S, and W nodes, respectively..

(2) ***Node transition database***: This keeps track of node state transition times; the file structure is defined as:
NodeTransitTime(NodeId, NodeCond, PrevState, CurrState, PrevTime, CurrTime)

(3) ***System dynamic database:*** This contains instances of time-stamped records of the system dynamic information; the file structure is defined as
SystemDynamicData((NodeId, CurrState, NodeCond, Td, Tr, Tw, Twtotal, MTBF)
NodeCond (node condition) may be failed, standby, working or maintenance due. MTBF is working node mean time before failure.

3.3.1 System temporal parameters and safety factors specifications

Suppose as part of the model builder's decisions, the following data are specified:

(a) Content of system static database: Fmax = 4, Smax = 4, and Wmax = 9;

(b) Time scale: Node state unit times may be in seconds, minutes, hours, etc. For instance, we may let Tdmax = 60, Tsmin = 4, Twmax = 30, and Tmaintdue = 720. Usually, such temporal parameters are not fixed arbitrarily; rather they are determined by some analysis as shown in section 3.3.2.

(c) The system safety factors are as specified in Table 3.4, where N_{sq}, denoted square nodes, and N_{cir}, circular nodes.

Table 3.4: System conditions and safety factors

S/N	System safety conditions	Safety factors
1	Fatal system failure	0.00
2	Many row-failure chance	0.00
3	One row-failure chance	0.20
4	N_{sq}/N_{cir} ratio < min. value	0.40
5	N_{sq}/N_{cir} ratio ≥ min. value	0.60
6	Minimal safe configuration	0.80
7	Optimal safe configuration	1.00

3.3.2 Relating the equilibrium probabilities and safety factors

When the values of Tdmax, Tsmin, and Twmax have been predefined, the range of temporal variables Td, Ts, and Tw can be determined as follows (based on the equilibrium probabilities and maximum time parameters specified in section 3.3.1):

(1) $t_d/t_{dmax} \geq P_1$ => $Td/60 \geq 0.166$ or $Td \geq 9.96$;
(2) $t_s/t_{smin} \geq P_2$ => $Ts/4 \geq 0.281$ or $Ts \geq 1.124$;
(3) $t_w/t_{wmax} \geq P_3$ => $Tw/30 \geq 0.553$ or $Tw \geq 16.59$;

Further, we assume that working node random failure occurs at the mean recurrent time of failed node (F) state. That is,

(4) Working node mean time before failure (MTBF) $\geq 1/P_1$ or $1/0.166 = 6.024$.

Reducing MTBF, Td, Ts, and Tw to unit time scale, we have 0.363, 0.60, 0.068, and 1.00, respectively. Thus, based on P_1, P_2, and P_3, the minimum values of MTBF, Td, Ts, and Tw that would allow node switching have been determined. Then, the simulation may be based on the time constraints $0.60 \leq t_d \leq t_{dmax}$, $t_s \geq 0.068$, and $1.00 \leq t_w \leq t_{wmax}$. Note that $t_s \geq t_{smin}$ is preferred.

The above analysis leads to enhancing the switching rules defined by equations (3.3a) – (3.3d) by incorporating the safety factors specified in Table 3.4. For instance, we can write

$$\text{If } (Ts \geq 0.068 \ \& \ Tw \geq 1.00 \ \& \ \text{Safety\_factor}(x_s, x_w) \geq 0.2) \text{ then } T(x_s, x_w) \qquad (3.6)$$

Predictive node switching control algorithm comprising rules in the form of equation (3.6) is designed. The algorithm is to use the databases to determine switchable nodes that satisfy temporal constraints and safety factors defined in Table 3.4. Determining the safety factors of switchable nodes is combinatorial aspect of the reliability modelling problem. As indicated by equation (3.6), it involves iterative analysis of switching standby and working nodes, considering system operational safety conditions. Reported in the next section are simulation tests of the entire system based on the specifications and switching control rules developed in this section.

4.0 Simulation results

Approach to model testing depends on the nature and application of the system [20–22]. For our experimental model, possible simulation tests include feasibility of the policy on tolerable number of failed, standby, and working nodes, optimal-time control, stability, and reliability. Considering system reliability in the context of availability and resilience to failure, the parameters t_{dmax}, t_{smin}, and t_{wmax} are applied as specified in the analysis of section 3.3.2, the unit of time being in seconds. Node failure is induced by randomization and the failure time interval is varied from 0.01 to 1.00 second. The number of switching controls and time intervals of entire system failure are recorded. It is observed that the number of system switching controls tends to infinity when node failure time interval is greater than 0.61 second, which is comparable to the unit downtime (T_d = 0.60 unit time) computed in section 3.3.2.

Twenty simulation runs are performed for each value of node meantime before failure. This results to a large numerical data output. Based on the computed system switching control speed, the output is sorted into eight (8) data classes and the behavioural graphs plotted. Sample graphs indicating oscillatory and fairly stable system trajectories are presented in figures 4.1a, 4.1b, 4.2a and 4.2b. It can be deduced from the graphs that node failure time interval from 0.40 to 0.61 second gives fairly stable system switching speed trajectory for the system specifications.

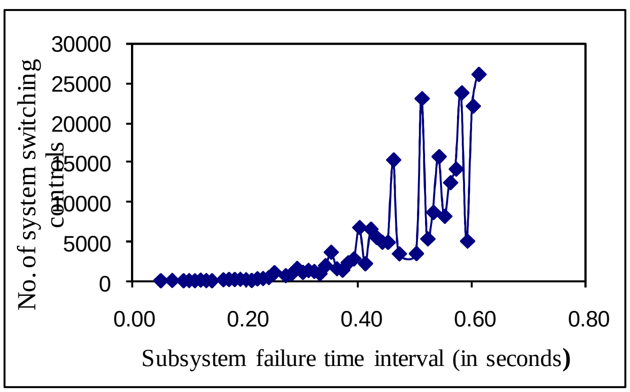

Figure 4.1a: System switching controls versus subsystem failure time

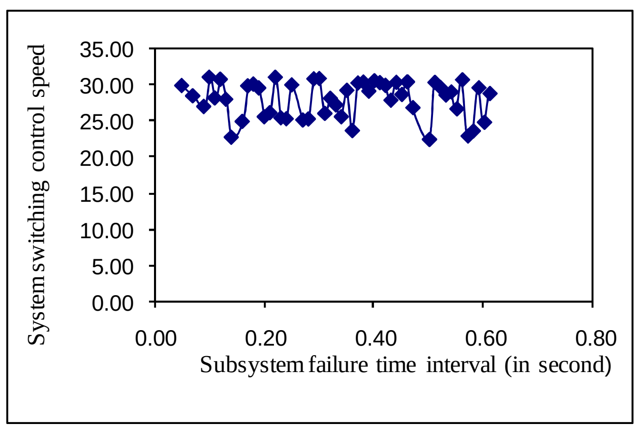

Figure 4.1b: System control speed versus subsystem failure time

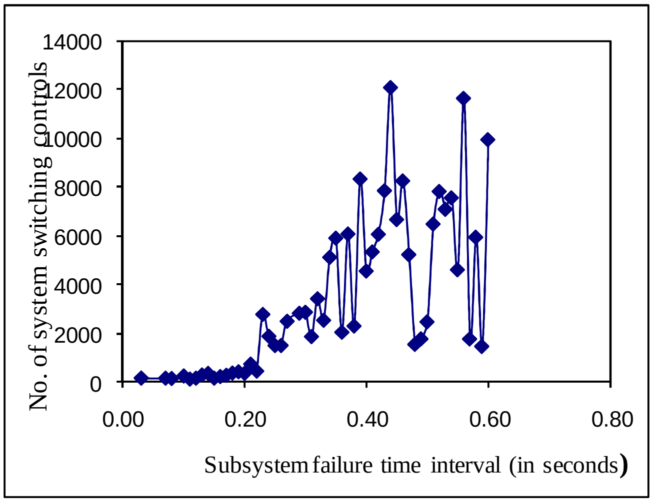

Figure 4.2a: System switching controls versus subsystem failure time

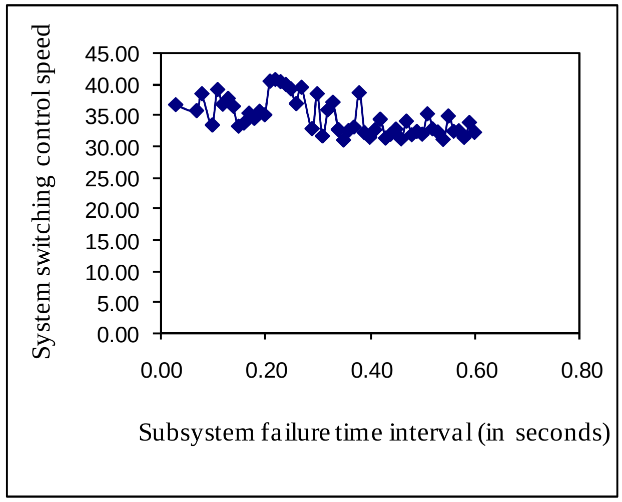

Figure 4.2b: System control speed versus subsystem failure time

5.0 Conclusion

In this study we have practically dealt with reliability modelling and simulation of switched linear system control using temporal databases. The experimental study is based on the abstraction of a tic-tac-toe board game called Jonda as a stochastic switching control problem, where the nodes are categorized into sets of failed, standby, and working subsystems. The controller is to predict switching sequences that would maintain specified operational safety conditions. To provide the basis for the entire system analysis, a subsystem Markov model is constructed and the equilibrium state transition probabilities matched with the temporal safety parameters. Then, temporal databases and switching control algorithm are designed for the system simulation. The simulation results indicate cases of highly oscillatory and fairly stable system trajectories from which optimal-time controls are deduced.

Finally, our methodology can be applied in optimal design of fault-tolerant real-time switching systems control and embedded micro-schedulers for complex systems maintenance. Moreover, the model developed can be adapted to study the dynamics of some switched linear systems with delays in state and control, most of which are treated theoretically in the literature.

References

[1] Henriques, J., Cardoso A. and Dourado A. (1999): Supervision and cmeans clustering of PID controllers for a solar power plant, *International Journal of Approximate Reasoning*, 22, pp. 73-91.

[2] Rato, L., Borelli D., Mosca E., Lemos J. and Balsa P. (1997): Musmar based switching control of a solar collector field, ECC97-European Control Conference.

[3] Tripakis S. and Yovine S. (1996): Analysis of timed systems based on time abstracting bisimulations. In Proc. CAV'96, Vol. 1102 of Lect. Notes in Comp. Sci. Springer-Verlag.

[4] Zhivoglyadov P. V., Middleton R. H. and Fu M. (2002): Localization based switching adaptive control for time-varying discrete-time systems," IEEE Trans. Automatic Control 45(4), pp. 752-755.

[5] Njoku C. N. (2011): A study on switched linear system identification using game-theoretic strategies and neural computing; To appear in the Journal of the Nigerian Association of Mathematical Physics.

[6] De Alfaro L. (1999): Computing minimum and maximum reachability times in probabilistic systems. In Proc. CONCUR '99, Vol, 1664 of Lect. Notes in Comp. Sci. Springer Verlag.

[7] Hansson H. and Johnson B. (1994): A logic for reasoning about time and reliability. Formal Aspects of Computing, 6(4): pp. 512-535.

[8] Niamsup P. and Phat V. N. (2000): Asymptotic stability of nonlinear system described by difference equations with multiple delays; In Electronic Journal of Differential Equations, Vol. 2000, No. 11, pp 1-17.
[9] Njoku C. N. (2004): "Towards the theory and applications of Jonda game"; In The Journal of Computer Science and Its Applications; An International Journal of Nigeria Computer Society. Vol 10, No. 1; pp 103 – 116.
[10] Shahabuddin P. (1994): Importance sampling for the simulation of highly reliable Markovian systems; Management Science, 40(3); pp. 333–352.
[11] Stewart. W (1994): Introduction to the Numerical Solution of Markov Chains. Princeton University Press.
[12] Basar T. and Bernhard P.(1995): H_∞ optimal control and related minimax design problems: A dynamic game approach, 2nd Ed., Birkhauser, Boston, 1995.
[13] Chen H., Scherer C., and Allgöwer F. (1997): A game-theoretic approach to nonlinear robust receding horizon control of constrained systems; Proceedings of the American Control Conference, Albuquerque New Mexico.
[14] Lygeros J., Godbole D. N, and Sastry S. (1996): A game-theoretic approach to hybrid system design. In R. Alur and T. Henzinger, editors, Hybrid Systems III, volume 1066 of Lecture Notes in Computer Science, pp. 1–12. Springer-Verlag.
[15] Tomlin C. J., Lygeros J. L. and Sastry, S. S. (2000): A game-theoretic approach to controller design for hybrid systems", IEEE Proc., Vol. 88, pp. 949-970.
[16] Coito F., Lemos J., Silva R. and Mosca E (1997): Adaptive control of a solar energy plant: Exploiting accessible disturbances, *IIJAC and Signal Processing*, 11, 4, 327-342.
[17] Lin, C. and Sun H. (2000): Intelligent control theory in the guidance and control system design: An overview, Proc. National Science Council, 24, 1, pp. 15-30.
[18] Hocherman-Frommer J., Kulkarni S. R., and Ramadge P. (1995): Supervised switched control based on output prediction errors; roc. 34th IEEE Conf. Decision and Control.
[19] Kushner H. and Yin G. (1997): Stochastic Approximation Algorithms and Applications. Springer-Verlag, New York.
[20] Alur R., Courcoubetis C., and Dill D (1993): Model-checking in dense real-time. Information and Computation, 104 (1).
[21] Baier C. and Kwiatkowska. M. (1998): Model checking for a probabilistic branching time logic with fairness. Distributed Computing 11.
[22] Kwiatkowska M., Norman G., Segala R., and Sproston J. (2000): Automatic verification of real-time systems with discrete probability distributions. Technical Report CSR-00-2, University of Birmingham.