

Model-Based Diagnosis of Hybrid Dynamical Networks for Fault Tolerant Control

Lachlan Blackhall^{*} and Priscilla Kan John[†]

The Australian National University,
Canberra, ACT, 0200, Australia

lachlan.blackhall, priscilla.kanjohn@anu.edu.au

Abstract

A method for determining (diagnosing) the operational mode of a hybrid dynamical network using model based diagnosis is presented. This methodology unifies discrete diagnosis capabilities with the continuous time dynamics of hybrid dynamical networks. This method is novel in that the diagnosis is not performed on the output trajectory of the network but rather on the fundamental governing dynamics of the network. This novelty allows for the diagnosis of arbitrary switching events in the network. The chosen methodology can be partially decentralised and provides optimal diagnoses even when only a portion of the network can be observed. This offers performance and implementation benefits for large distributed networks. Computational results are presented for the case where the mode changes are purely structural. That is, where only the interconnection structure of the network is changing. The results indicate that this methodology has some benefits for the fault-tolerant control of hybrid dynamical networks.

1 Introduction

Effectively controlling dynamical networks (systems where there are very many constituent components physically linked together) poses many challenges. These challenges are somewhat more significant in the presence of faults within the network. Faults drastically alter the fundamental governing dynamics (how the constituent systems or components are linked together and affect each other) of the network rendering the original, fault-free control methodology severely degraded or useless.

We thus require that an effective network controller be able to detect that a fault has occurred and change control strategy accordingly. Choosing an appropriate control strategy for a given configuration is achievable and has been the focus of extensive work in the field of control theory [Khalil,

2002]. Additionally these control strategies can usually be computed off-line. What is lacking is a systematic on-line methodology to automate the process of detecting the occurrence of a fault(s) and choosing the most appropriate control strategy. Additionally we wish to minimise the fault monitoring required to still achieve the global performance objectives. Allowing the fault detection and recovery strategy to run in a distributed manner would be an additional benefit.

The discrete diagnosis community has undertaken considerable work in the area of fault diagnosis and we propose to apply these diagnosis methods to detect and recover, in real time, from faults experienced in hybrid dynamical networks. The unification of the discrete diagnosis capability and continuous time operation of dynamical networks is discussed in depth in Section 3.

Much of the previous work focuses on detecting faults by measuring deviations of a dynamical system (a subset of a hybrid dynamical network) from the ideal or nominal trajectory. In [Blanke, 2006] a residual generating transfer function is created that forces the residual to become non-zero in the presence of a fault. In [Narasimhan *et al.*, 2000] a similar approach is used. Both works categorise faults through analysis of the system output, requiring that faults be manifest in the residual error between the system output and a nominal trajectory. In dynamical networks, where there are many interconnected systems, it is theoretically and practically challenging to determine a nominal trajectory for each system and the network as a whole. Our diagnosis methodology rather focuses on operational mode changes that can be determined by changes in the fundamental governing dynamics of the network.

In [McIlraith *et al.*, 2000] the authors approach diagnosis as a model selection problem. An initial set of qualitative candidate diagnoses are conjectured and then refined using parameter estimation and model fitting techniques. The assumptions that controller actions are responsible for all events and the inability to deal with multiple sequential faults limits the applicability of this work to dynamic networks where these assumptions are not always valid.

Other work has focussed on the prediction and selection of the dynamical mode that generates the trajectory that best describes the output of the dynamical system. In [Hofbaur and Williams, 2002] the modeling framework proposed is a probabilistic hybrid automata (PHA). These merge hidden

^{*}Lachlan Blackhall is with The Research School of Information Sciences and Engineering (RSISE).

[†]Priscilla Kan John is with National Information and Communications Technology Australia (NICTA) and The Research School of Information Sciences and Engineering (RSISE).

Markov models (HMM) with continuous dynamical system models to achieve diagnosis. In [Lerner *et al.*, 2002] systems are modelled as dynamic Bayesian networks (DBN). This representation is, in principle, able to deal with multiple simultaneous failures and is robust to parameter drift. In both approaches tracking a number of trajectories becomes intractable and they are required to use heuristics to reduce the computational burden.

Model-based diagnosis is concerned with determining the sequence of operational modes of a discrete system, where it is assumed that the occurrence of a fault could cause changes in the operational mode of the system. Model-based diagnosis assumes that the modes and events can be deterministically related via a model (*e.g.* an automata). Thus, by using observations of the system in question it is possible to make a diagnosis (determine the operational mode) that is consistent with the observations. Where multiple systems have common faults, it is possible to synchronise the diagnoses to achieve a *global diagnosis*. In a dynamic networks where there are many systems, often with interacting faults this synchronisation can become computationally intractable. Recent work by [Kan John and Grastien, 2008] allows this challenge to be overcome by providing distributed methods that achieve globally consistent diagnoses using only local diagnosis computations.

In this paper, we propose a diagnosis approach for dynamical hybrid networks with partial observations based on the detection of changes in the fundamental dynamics of the network. Our approach is both distributed and model-based, requiring only local diagnosis calculations to ensure global consistency.

2 Preliminaries

2.1 Graphs and Networks

Networks have emerged as a way of describing and analysing large numbers of individual elements or agents that interact. Understanding the interaction of these elements is of considerable interest in order to better understand the collective operation of the whole. We begin with a preliminary definition given in [Diestel, 2005].

Definition 2.1 (Directed Graph). A directed graph $G = (V, E)$ is a pair (V, E) of disjoint sets and two maps, $init(E) : E \rightarrow V$ and $ter(E) : E \rightarrow V$, that assign to each edge an initial and terminal vertex respectively, where $E \subseteq V \times V$. In this way a given edge is said to be directed from $init(E)$ to $ter(E)$. To avoid ambiguities we assume that $V \cap E = \emptyset$. We thus have that the elements of V are vertices and the elements of E are edges that represent a directed relationship between two vertices.

From these basic definitions we are now able to define a network.

Definition 2.2 (Network). A **network** is the directed graph formed by the interconnection of a set of nodes (N) through the set of links (L) by the maps $init(L) : L \rightarrow N$ and $ter(L) : L \rightarrow N$, that assign to each edge an initial and terminal vertex respectively and will be written as $G_{network} = (N, L)$.

Dynamical Networks

Having defined networks generally we now focus specifically on dynamical networks. In order to properly define a dynamical network we need a few preliminary definitions which we present now.

Definition 2.3 (Dynamical Node). A dynamical node (N_{dyn}) is a system whose operation, in isolation, is governed by a set of dynamical equations that describe implicitly how the state of the system changes with time. Dynamical nodes are given by the general form:

$$\begin{aligned}\dot{\mathbf{x}}_k(t) &= f_k(\mathbf{x}_k(t)) + g_k(\mathbf{x}_k(t), \mathbf{u}_k(t)) \\ \mathbf{y}_k(t) &= h_k(\mathbf{x}_k(t))\end{aligned}\quad (1)$$

where $\dot{\mathbf{x}}$ is the time derivative of the node state vector $\mathbf{x} \in \mathbb{R}^n$ that contains the quantities needed to describe the operating condition of the node. $\mathbf{u} \in \mathbb{R}^m$ is the local control vector that gives the values of the local control inputs used to control the operation of the dynamical node. $\mathbf{y} \in \mathbb{R}^p$ is the local output that gives the values of the measurable output of the dynamical node. $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g(\mathbf{x}) : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ and $h(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^p$ are the internal, control and output functions respectively that determine how the local states and control inputs cause the internal and output states of the dynamical node to evolve.

Definition 2.4 (Interconnection). An interconnection (L_{kj}) is a directed physical or information theoretic link between two dynamical nodes (N_k) and (N_j), where $init(L_{kj}) : L_{kj} \rightarrow N_k$ and $ter(L_{kj}) : L_{kj} \rightarrow N_j$. We define the value of an interconnection between two dynamical nodes (N_k) and (N_j) as $(l_{kj}(\mathbf{x}_k, \mathbf{y}_j) : \mathbb{R}^{n_k} \times \mathbb{R}^{p_j} \rightarrow \mathbb{R}^{n_k})$

With this framework we are now able to define a dynamical network.

Definition 2.5 (Dynamical Network). A dynamical network $G_{net_{dyn}} = (N_{dyn}, L)$ is a network where the set of nodes (N) and the set of interconnections (L) are as given previously (cf. Def 2.3 and Def 2.4 respectively). We write $G_{net} = (N, L)$ to represent the dynamical network $G_{net_{dyn}}$ where we have dropped the dynamical subscript. Mathematically a dynamical network consists of nodes (N_k) with a set of internal states ($\mathbf{x}_k \in \mathbb{R}^{n_k}$), a set of local control inputs ($\mathbf{u}_k \in \mathbb{R}^{m_k}$), and a set of measurable outputs given by ($\mathbf{y}_k \in \mathbb{R}^{p_k}$). We also have an associated set of internal dynamics ($f_k(\mathbf{x}_k) : \mathbb{R}^{n_k} \rightarrow \mathbb{R}^{n_k}$), an associated set of control dynamics ($g_k(\mathbf{x}_k, \mathbf{u}_k) : \mathbb{R}^{n_k} \times \mathbb{R}^{m_k} \rightarrow \mathbb{R}^{n_k}$) and output function given by ($h_k(\mathbf{x}_k) : \mathbb{R}^{n_k} \rightarrow \mathbb{R}^{p_k}$). Furthermore each dynamic node is connected to other dynamic nodes through non-linear interconnections ($l_{kj}(\mathbf{x}_k, \mathbf{y}_j) : \mathbb{R}^{n_k} \times \mathbb{R}^{p_j} \rightarrow \mathbb{R}^{n_k}$). This gives the dynamical model of the k -th dynamical system in a dynamical network as:

$$\begin{aligned}\dot{\mathbf{x}}_k &= f_k(\mathbf{x}_k) + g_k(\mathbf{x}_k, \mathbf{u}_k) + \sum_j l_{kj}(\mathbf{x}_k, \mathbf{y}_j) \\ \mathbf{y}_k &= h_k(\mathbf{x}_k)\end{aligned}\quad (2)$$

Hybrid Dynamical Networks

Hybrid dynamical networks allow the continuous time operation of a single node or link to be broken into discrete operating modes. An operating mode of a hybrid system is a distinct

type of behaviour which is governed by a set of dynamics that is different from other modes of the system. In this way we are able to partition the operation of the dynamical nodes into discrete modes. This provides an elegant way of abstracting the continuous time dynamics of the network into a form that allows faults and other changes in the network dynamics to be accurately modelled. We proceed using definitions similar to those in [Zhao and Hill, 2008].

Definition 2.6 (Hybrid Dynamical Network). A hybrid dynamical network is a dynamical network where the dynamical nodes (N) and interconnections (L) have a hybrid characteristic which can be represented in the form:

$$\begin{aligned}\dot{\mathbf{x}}_k &= f_{k\sigma_k}(\mathbf{x}_k) + g_{k\sigma_k}(\mathbf{x}_k, \mathbf{u}_{k\sigma_k}) + \sum_j l_{kj\sigma_k}(\mathbf{x}_k, \mathbf{y}_j) \\ \mathbf{y}_k &= h_{k\sigma_k}(\mathbf{x}_k)\end{aligned}\quad (3)$$

for $k \in 1, 2, \dots, K$ where σ_k is the switching signal taking values in:

$$M_k = \{1, 2, \dots, m_k\}$$

M_k is the set of operating modes of the system. Additionally, $x_k \in \mathbb{R}^{n_k}$ is the state vector of the system, $u_{k\sigma_k} \in \mathbb{R}^{m_{k\sigma_k}}$ is the input control vector to the system and $y_k \in \mathbb{R}^{p_k}$ is the output vector of the system. $f_{k\sigma_k}$, $g_{k\sigma_k}$ and $h_{k\sigma_k}$ are continuous.

The switching signal ($\sigma_k : \mathbb{R}_+ \rightarrow M_k$) for the k -th hybrid system can be characterised by the switching sequence:

$$\Sigma_k = \{(i_{k_0}, t_{k_0}), (i_{k_1}, t_{k_1}), \dots, (i_{k_z}, t_{k_z}) | i_{k_z} \in M_k, z \in \mathbb{Z}\}$$

where x_{k_0} , t_{k_0} are the initial system state and time respectively of the k -th system and $\mathbb{Z}_k$ is the set of nonnegative integers. We note that $t_1 = t_2 = \dots = t_K = t$, that is all systems have the same time. When $t_k \in [t_{k_z}, t_{k_{z+1}})$, $\sigma_k = i_{k_z}$, we say the i_{k_z} -th subsystem of system k is active and the trajectory of the switched system ($x_k(t)$) is defined as the trajectory of the system ($x_{k_{i_k}}(t)$) with switching signal $i_k \in M_k$.

When there is only a single system in the network we recover the usual definition for a single hybrid dynamical system. We assume that the state of the system is continuous and thus does not exhibit abrupt changes at the instant of switching. The entire internal, control and network dynamics of the network at time t can be characterised by a K -tuple $([\sigma_1(t), \sigma_2(t), \dots, \sigma_K(t)] \in \{M_1 \times M_2 \times \dots \times M_K\})$.

Hybrid Dynamical Modes

We present here some definitions and concepts relevant to the representation of hybrid dynamical nodes as graphs in order to more clearly define what we mean by the modes of a hybrid dynamical network.

Definition 2.7 (Set of States). The set of states (X) of a dynamical node is a discrete set composed of the functions ($x_i : \mathbb{R}_+ \rightarrow \mathbb{R}$) that for each $t \geq 0$ returns the value of the node state defined by the function x_i .

Definition 2.8 (Set of Outputs). The set of outputs (Y) of a dynamical node is a discrete set composed of the functions ($y_i : \mathbb{R}_+ \rightarrow \mathbb{R}$) that for each $t \geq 0$ returns the value of the node output defined by the function y_i .

Definition 2.9 (Set of Controls). The set of controls (U) of a dynamical node is a discrete set composed of the functions ($u_i : \mathbb{R}_+ \rightarrow \mathbb{R}$) that for each $t \geq 0$ returns the value of the node controls defined by the function u_i .

We now define our dynamical node as a graph progressing in a similar but slightly more general way than was presented in [Boukhobza *et al.*, 2007].

Definition 2.10 (Dynamical Node Graph). A dynamical node graph is a directed graph representing the internal interconnection of the set of states, controls and network inputs and outputs of a dynamical node ($N = X \cup U \cup Y$) through the set of links ($L = L_{XX} \cup L_{UX} \cup L_{YX} \cup L_{XY}$) by the maps $init(L) : L \rightarrow N$ and $ter(L) : L \rightarrow N$, that assign to each edge an initial and terminal vertex respectively.

The set of links defined above is formally defined as:

Definition 2.11 (Set of Links). The internal interconnection links between states for the k -th dynamical node is given by $L_{XX_k} = \{(x_{k_i}, x_{k_j}) | \frac{\partial x_{k_j}}{\partial x_{k_i}} \neq 0\}$, $k_i, k_j \in 1, \dots, n_k$. The internal interconnection links between control inputs and states for the k -th dynamical node is given by $L_{UX_k} = \{(u_{k_i}, x_{k_j}) | \frac{\partial x_{k_j}}{\partial u_{k_i}} \neq 0\}$, $k_i \in 1, \dots, m_k$, $k_j \in 1, \dots, n_k$. The internal interconnection links between states and outputs for the k -th dynamical node is given by: $L_{XY_k} = \{(x_{k_i}, y_{k_j}) | \frac{\partial y_{k_j}}{\partial x_{k_i}} \neq 0\}$, $k_i \in 1, \dots, n_k$, $k_j \in 1, \dots, p_k$ and the interconnection links between all the network outputs and internal states for the k -th dynamical node is given by $L_{YX_k} = \{(y_{k_K}, x_{k_j}) | \frac{\partial x_{k_j}}{\partial y_{k_K}} \neq 0\}$, $k_K \in 1, \dots, \sum_k m_k$, $k_j \in 1, \dots, n_k$ where (v_1, v_2) denotes a directed edge from node $v_1 \in N$ to node $v_2 \in N$.

We also define the capacity of a structural dynamic graph as a measure of the magnitude of each of the non-zero links defined in the link sets above.

Definition 2.12 (Capacity of a Dynamical Node Graph). The capacity of a structural dynamic graph is a set C_L that assigns to each link $L_i \in L$ a measure of the capacity of that link. For the link sets defined in Def. 2.11 we have the following capacity sets for the k -th dynamical node: $C_{L_{XX_k}} = \{\frac{\partial x_{k_j}}{\partial x_{k_i}} | (x_{k_i}, x_{k_j}) \in L_{XX_k}\}$, $C_{L_{UX_k}} = \{\frac{\partial x_{k_j}}{\partial u_{k_i}} | (u_{k_i}, x_{k_j}) \in L_{UX_k}\}$, $C_{L_{XY_k}} = \{\frac{\partial y_{k_j}}{\partial x_{k_i}} | (x_{k_i}, y_{k_j}) \in L_{XY_k}\}$ and $C_{L_{YX_k}} = \{\frac{\partial x_{k_j}}{\partial y_{k_i}} | (y_{k_i}, x_{k_j}) \in L_{YX_k}\}$.

From these definitions we are able to very precisely define the concept of a change in mode and this is the topic of the next section.

Mode Changes

A change in mode is a change of the fundamental governing dynamics and we can see this as follows. For the switching sequence of the k -th node in a hybrid dynamical network, $\Sigma_k = \{(i_{k_0}, t_{k_0}), (i_{k_1}, t_{k_1}), \dots, (i_{k_z}, t_{k_z}) | i_{k_z} \in M_k, z \in \mathbb{Z}\}$ we take $L_{i_{k_z}} = L_{XX_{i_{k_z}}} \cup L_{UX_{i_{k_z}}} \cup L_{YX_{i_{k_z}}} \cup L_{XY_{i_{k_z}}}$ to be the total link set in the time interval $[t_{k_z}, t_{k_{z+1}})$ and $C_{L_{i_{k_z}}} = C_{L_{XX_{i_{k_z}}}} \cup C_{L_{UX_{i_{k_z}}}} \cup C_{L_{YX_{i_{k_z}}}} \cup C_{L_{XY_{i_{k_z}}}}$ to be

the equivalent total capacity set in the same interval. A mode change implies that for any $z \in \mathbb{Z}$, $L_{i_{kz}} \neq L_{i_{kz+1}}$ and/or $C_{L_{i_{kz}}} \neq C_{L_{i_{kz+1}}}$. That is the change in mode forces the link and/or capacity sets to change, meaning a fundamental change in the fundamental governing dynamics. It should be noted that determining the link and capacity sets is possible using a variety of parameter and system identification methods, however, these methods are beyond the scope of this paper. With these concepts fully defined we are now able to understand how we can unify hybrid dynamical networks and discrete diagnosis.

2.2 Model Based Diagnosis

Diagnosis is the problem of determining what possibly happened on a system given some observations of its behaviour. In model-based diagnosis, we assume that we have a model that describes the discrete transitions between modes in the system (i.e. given an operational mode the model describes the possible transitions that will result in a new operational mode). The model is an abstraction of the system transitions and is often represented by an automaton.

2.3 Diagnosis of Discrete Event Systems

We outline the diagnosis of discrete event systems in order to apply this approach to the diagnosis of hybrid dynamical networks. Essentially, we perform an abstraction on the discrete part of a hybrid system to obtain a model on which we can perform diagnosis. More details are given in section 3.2.

We consider the discrete evolution of a system. The set of all possible behaviours of the system is a language denoted Mod over the set of events Σ that could possibly occur on the system. In our work, we assume that observations we make are on the modes of a given system (in contrast to more commonly used event-based observations [Largouet and Cordier, 2000]). We denote these observations (a set of possible modes) on the system by a language Obs . The diagnosis of the system can be computed as

$$\Delta = Mod \otimes Obs. \quad (4)$$

where $\otimes$ is the projection of the model onto the observations. The diagnosis Δ essentially determines the current operational mode consistent with the observations. In this way, we are able to refine the original mode estimates in Obs . The strict definition of languages in this context and operations on them are presented in [Kan John and Grastien, 2008].

2.4 Distributed Diagnosis

It is possible to use diagnosis techniques to overcome the problem of diagnosing large and complex systems which is of considerable interest. We make use of distributed diagnosis algorithms on which one of the authors is working [Kan John and Grastien, 2008] to minimise the computational complexity in large diagnosis problems.

In a distributed diagnosis approach [Pencol  and Cordier, 2005], [Cordier and Grastien, 2007], [Su and Wonham, 2005], [Kan John and Grastien, 2008] we only consider local computations. Rather than considering the diagnosis on the whole network, we only compute diagnoses locally (on

each node). The problem is then to make sure that the local sets of diagnoses are globally consistent.

Global Consistency

We consider a network S of interconnected nodes $S = \{S_1, S_2, \dots, S_k\}$. The diagnosis of S_i is given by

$$\Delta_i = Mod_i \otimes Obs_i. \quad (5)$$

The diagnosis Δ of S is said to be *globally consistent* if the overall diagnosis agrees with the diagnosis at each node. A strict definition of global consistency is presented in [Kan John and Grastien, 2008].

Local (pairwise) consistency does not normally ensure global consistency. However, we can transform the topological graph of the whole network into a form (in our case a junction tree) which will ensure that local consistency leads to global consistency [Kan John and Grastien, 2008]. We then have groups of nodes as clusters of the tree. Diagnosis is performed locally on each cluster and local consistency is applied until a fixpoint is reached.

3 Unifying Discrete Diagnosis and Hybrid Dynamical Networks

We explain in this section the method by which hybrid dynamical networks and discrete diagnosis may be unified. We first discuss some of the assumptions that we are making during this analysis.

3.1 Analysis Assumptions and Discussion

The assumptions used in this analysis are ideally kept to a minimum to maximise the real-world applications of this approach. Importantly these assumptions do not restrict the method by which a mode changes. It is reasonable to assume that an event may be due to a fault, a change in environmental conditions, the effects of a control choice in a given mode or a variety of other reasons. By looking at the outcome of the event rather than the source of the event we are better placed to use this methodology for dealing with many of the challenges faced in modern dynamical networks. Our assumptions also allow faults to occur in succession provided they do not occur instantaneously. This is also advantageous as it allows us to deal with cascading type faults which are of significant concern in modern hybrid dynamical networks. The following assumptions are made during the analysis.

Mutual Exclusivity of Modes

We assume that the modes (M_k) of a system are mutually exclusive. That is a system (S_k) in the dynamic network can only be in one mode at any time. That is if $\sigma_k = m_i \in M_k$ for $t_k \in [t_{k_i}, t_{k_{i+1}})$ and $\sigma_k = m_j \in M_k$ for $t_k \in [t_{k_i}, t_{k_{i+1}})$ then $m_i = m_j$. This is a natural assumption that ensures any events cause the system to move into one of a discrete set of modes and that one and only one mode is responsible for describing the dynamic evolution of a system in the network at a given time.

Timing of Events

We assume that between two observations only one event can occur. This further implies that for any finite $T > t_0$, there

exists a positive integer K_T , which may depend on T , such that during the time interval $[t_0, T]$ each dynamical node (Eq. 1) switches no more than K_T times. This ensures that it is practically possible to diagnose a system as the sequence of observations will have information about all the faults that occur in the system.

Controller Laws

We are assuming that for a given operational mode it is possible to compute an appropriate controller off-line [Khalil, 2002]. The methods for achieving this are too many and varied to enumerate and cover the span of classical and modern control for a period of almost a century. For this reason we focus purely on the method of choosing an appropriate control strategy when the fault occurs. A simple justification of this can be seen in that if the operating mode of each dynamical node can be chosen independently there are:

$$\prod_k m_k \quad (6)$$

K-tuples that characterise the operating condition of the hybrid dynamical network. In networks with large numbers of systems and internal modes this rapidly becomes computationally intractable.

In the majority of dynamical networks a single event or fault will effect a number of systems in the network and thus using diagnosis methodologies allows this number to be drastically reduced so that the off-line computation of controllers is actually feasible. Furthermore if we consider networks where only a subset of the systems are directly observed then without the relationships stored in the diagnosis algorithm it would be impossible to diagnose and control for the effects of those unobserved systems.

Mode Determination

Again, the ability to determine the modes is beyond the scope of this paper and covers methodologies in system and parameter identification [Ljung, 1999] [Blackhall and Rotkowitz, 2008] over the last half century or so. The authors believe strongly that it is a more than valid assumption that modes can be determined with high accuracy in the majority of scenarios.

3.2 Diagnosis of Hybrid Dynamical Networks

In diagnosis terms we consider a hybrid dynamical network S of interconnected dynamical nodes $S = \{S_1, S_2, \dots, S_K\}$. This means that each node has a defined operating mode and each transition causes a change in mode that redefines the continuous behaviour of the system.

We can think of each system in S as having a continuous behaviour captured by a tuple $S_{k_{cont}} = \langle \mathbf{x}_k, u_{k\sigma_k}, \mathbf{y}_k, f_{k\sigma_k}, g_{k\sigma_k}, l_{k1\sigma_k}, \dots, l_{kK\sigma_k} \rangle$ where the elements of the tuple have been defined previously.

We also make an abstraction on the discrete behaviour of each dynamical node using an automaton. Thus the automaton representing the discrete behaviour of a system can be given by the tuple $S_{k_{disc}} = \langle \mathcal{M}_k, \Omega_k, \mathcal{T}_k, \sigma_{k_0} \rangle$.

- $\mathcal{M}_k$ is the finite set of system modes $(m_1, m_2, \dots, m_K)$.

- Ω_k is the set of events $(\omega_1, \omega_2, \dots, \omega_l)$ which determine switching between modes.
- $\mathcal{T}_k$ is the transition function that maps an event and mode into a new mode, $\mathcal{T} : \Omega_k \times \mathcal{M}_k \rightarrow \mathcal{M}_k$.
- σ_{k_0} is the initial mode of each system.

Using the diagnosis methods defined previously in sections 2.3 and 2.4 it is possible to compute a diagnosis of the discrete behaviour of each dynamical node. As each discrete behaviour has a corresponding continuous behaviour we have thus shown that it is possible to perform diagnosis of a continuous hybrid dynamical network using the discrete model based diagnosis methodology.

3.3 Algorithmic Considerations

We outline two methods in which distributed diagnosis can be used in maintaining a fault tolerant network. Algorithm 1 presents a global optimal approach. Algorithm 2 utilises the power of the junction tree technique to address the time criticality issue that often arises in large hybrid dynamical networks to try and avoid cascading failures.

We consider a hybrid dynamical network $S = \{S_1, \dots, S_K\}$ each of which has an automaton model. We thus have a set of models $Mod_1, \dots, Mod_K$ available. We have a set $\mathcal{C} = \{C_1, \dots, C_r\}$ of controllers that can be used on the system. At a given time t , we have observations $Obs_1, \dots, Obs_K$ on the system where each Obs_k is a set of estimates of the mode of a given component S_k at t obtained from parameter estimation and system identification. In Al-

Algorithm 1 Global Optimal Algorithm for diagnosis and control of a hybrid dynamical network

- 1: At time t_{obs}
 - 2: **input** $\{Mod_1, \dots, Mod_K\}, \{Obs_1, \dots, Obs_K\}, \mathcal{C}$
 - 3: **for all** $t_{obs} \in t_0, \dots, t_{final}$ **do**
 - 4: **do** $\Delta_{t_{obs}} := Mod_{t_{obs}} \otimes Obs_{t_{obs}}$
 - 5: **end for**
 - 6: Run consistency algorithm (returns globally consistent network)
 - 7: **for all** $t_{obs} \in t_0, \dots, t_{final}$ **do**
 - 8: pick a control strategy C from $\mathcal{C}$
 - 9: **end for**
-

gorithm 1, we pick the best control strategy over a globally consistent network. Provided that the diagnosis and global consistency can be performed fast enough, this provides the global optimal solution. However, in a large network, time is often a critical factor in the occurrence of cascading type faults. If we can choose a locally optimal controller after performing local consistency, this could help prevent other dynamical nodes in the system from failing as a potential fault is captured and dealt with as soon as possible. This approach is outlined in Algorithm 2.

3.4 Diagnosis under partial observations

The problem with large networks is that it is not always possible or practical to obtain observations on all dynamical nodes. However, because components share common events, it is

Algorithm 2 Optimal Time Critical Algorithm for diagnosis and control of a hybrid dynamical network

```

1: At time  $t$ 
2: input  $\{Mod_1, \dots, Mod_K\}, \{Obs_1, \dots, Obs_K\}, \mathcal{C}$ 
3: for all  $t_{obs} \in t_0, \dots, t_{final}$  do
4:   do  $\Delta_i := Mod_i \otimes Obs_i$ 
5: end for
6: while not globally consistent do
7:   Perform local consistency with neighbours
8:   pick a control strategy  $C$  from  $\mathcal{C}$ 
9: end while

```

still possible to diagnose unobserved nodes by using the diagnosis of the observed ones. For an unobserved component, we take Obs to be the set of all possible modes. We then refine the diagnosis when performing local consistency with other components.

4 Numerical Example

We will use the fault diagnosis methodology developed to diagnose and recover from failures on a four node complex network (Fig. 1). Complex networks (described for example in [Li *et al.*, 2004]) have emerged out of the biological and social sciences and are an attempt to explain the behaviour of large interconnected groups. From a dynamical and systems theoretic perspective complex networks consist of independent dynamical nodes connected together. We use the following representation:

$$\dot{\mathbf{x}}_i = f(\mathbf{x}_i) + c \sum_j a_{ij} \Gamma(\mathbf{x}_j - \mathbf{x}_i) + \epsilon_i(\mathbf{s}(t) - \mathbf{x}_i) \quad (7)$$

where it is clear that this complex network is in the dynamical network form presented in Eq. 2. We have $\mathbf{x} \in \mathbb{R}^n$ is the state vector, $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is the unique internal dynamics, a_{ij} determines the interconnections in the network and is symmetric. As such $a_{ij} = a_{ji} = 1$ if node i and node j are connected together and zero otherwise. Γ is the feedback connectivity matrix that determines how the difference vector $(\mathbf{x}_j - \mathbf{x}_i)$ effects the internal state and c is the scalar gain that determines the magnitude of this effect. We also have the control input $\epsilon_i(\mathbf{s}(t) - \mathbf{x}_i)$ where $\mathbf{s}(t)$ is the desired equilibrium state of the dynamical node and the network as a whole and $\epsilon_i \in \{0, \epsilon\}$. That is only some nodes have a local feedback control applied with control gain ϵ . This is referred to in the literature as pinning control [Li *et al.*, 2004]. In complex networks the control problem becomes ensuring that the network will synchronise, where synchrony is defined as:

$$\mathbf{x}_1 = \mathbf{x}_2 = \dots = \mathbf{x}_n \rightarrow \mathbf{s}(t)$$

such that $\dot{\mathbf{x}}(s(t)) = f(s(t))$ is the solution of the isolated nodes.

It is well known in the complex networks community that the ability of the network to synchronise is dependent on the value of c chosen as well as the location and magnitude of the pinning control that is applied to the network. Whilst an arbitrarily large c and pinning control gain (ϵ) can always be

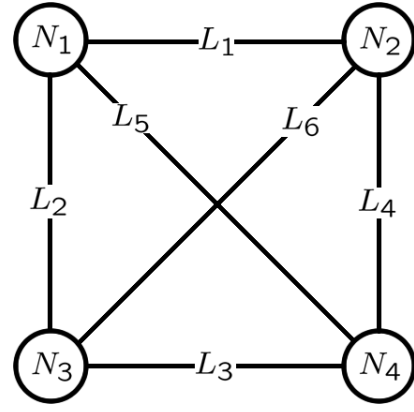


Figure 1: The four node, six link complex network analysed in this example.

chosen this is often physically impossible as well as being inefficient as much more control authority is being commanded than is necessary to achieve the desired synchrony.

In our example we choose $f(x_i) = -x_i^2 + 5x_i$, $\Gamma = I$, the four dimensional identity matrix and the network is globally connected. With $c = 1.5$ we can see the resultant state trajectory in Fig. 2, where synchrony is rapidly achieved.

We consider faults that effect the structure of the network (similar to [Kim and Hill, 2008]). That is we consider a network where the value of $a_{ij} = a_{ji}$ changes when a fault occurs, thus representing the hybrid component of the dynamical network presented in Eq. 7. This further implies that only the link sets change when a fault occurs ($L_{i_{k_z}} \neq L_{i_{k_z+1}}$). That is the fault causes a fundamental change in the network dynamical structure. The capacity sets do not change ($C_{L_{i_{k_z}}} \cap C_{L_{i_{k_z+1}}} \subseteq C_{L_{i_{k_z+1}}}$) and thus networks of this type are said to be capacity invariant. Even in this simple network the possible structural dynamical fault modes are substantial. We consider ten possible events, four due to node removal and six due to link removal. In the case of node removal we assume that all the links between the removed node and the network are also removed simultaneously. These fault modes correspond to the typical faults found in complex network operation. We consider two possible fault scenarios in this example:

- The network link L_5 becomes disconnected.
- The network system N_4 becomes disconnected.

We assume that we only have partial observations on the network and thus we are only able to monitor the modes of systems N_1 and N_2 . The two faults outlined are important because it is only when we use the diagnosis algorithm, and the observations from both nodes N_1 and N_2 , that we are able to exactly determine the fault that occurred and implement an appropriate control strategy.

While we can always assume that we can increase the control effect on the nodes in the case of a fault, this is an inappropriate action as it is known that when network link L_5 becomes disconnected no additional control action is required to ensure network synchrony. Conversely, when network sys-

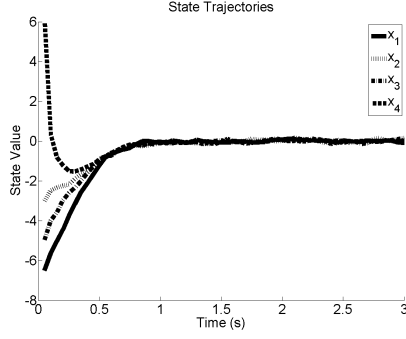


Figure 2: The state trajectories of each system in the network when the network is globally connected.

tem N_4 becomes disconnected we must apply local control to system N_1 to ensure the network achieves the desired synchrony. In the latter case we apply the local controller described previously using a controller gain of $\epsilon_1 = 2.5$ about the equilibrium point $\mathbf{x}_1 = \mathbf{x}_3 = \mathbf{x}_4 = 0$.

The dynamics are simulated in MATLAB and the system is started each time at the states $(x_1, x_2, x_3, x_4) = [-6.5, -3, -4.95, 6]$ and the fault occurs at $t = 0.1s$. We make observations of the system every $\Delta t_{observations} = 0.4s$. We assume that the internal dynamics of the system are well characterised and thus we are able to measure the network dynamics in isolation. We assume that any inaccuracies in our measurement of the network dynamics can be modelled as zero mean, i.i.d. gaussian noise (η). On this basis our measurement model is:

$$y_{i_{observation}} = c \sum_j a_{ij} \Gamma(\mathbf{x}_i - \mathbf{x}_j) + \eta \quad (8)$$

To determine the dynamical structure of the system we are interested in determining all the values of a_{ij} . This is a special case of regression known as sparse regression, more details of which can be found in [Blackhall and Rotkowitz, 2008]. We use the algorithm from [Blackhall and Rotkowitz, 2008] to determine the values of all the a_{ij} , thus allowing us to determine the link set and thus the modes of the dynamical nodes N_1 and N_2 . We re-initialise the algorithm after every observation to ensure we can determine when the dynamical structure of a node changes, thus allowing rapid determination of fault occurrence in the hybrid dynamical network.

We always assume the initial mode to be fault free and when we induce the first of our faults and the network link L_5 becomes disconnected the diagnosis algorithm is capable of determining that this is the fault that has occurred by synchronising the observations of system N_1 and system N_2 . We observe in the resultant state trajectories (Fig. 3) that the systems take longer to achieve synchrony but are still able to do so. Although the performance of the network is reduced (in terms of achieving synchrony) there is no need to inject additional control as the objective is still achieved in a reasonable time as compared to the normal network operation (Fig. 2).

When the second of the faults is introduced and the network system N_4 becomes disconnected we can see the resultant trajectory that would occur in the absence of diagnosis

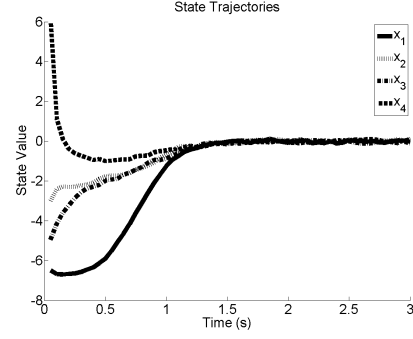


Figure 3: The state trajectories of each system in the network when the network link L_5 becomes disconnected due to a fault. The diagnosis algorithm is used here to determine the fault that has occurred and that no additional control input is required to achieve network synchrony.

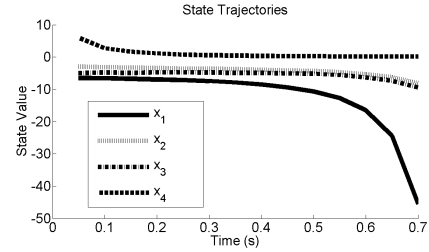


Figure 4: The state trajectories of each system in the network when the system N_4 becomes disconnected from the network due to a fault. The diagnosis algorithm is not running and we do not initiate any external control.

and thus external control being applied (Fig. 4). Alternatively when we use the diagnosis approach, the algorithm again accurately determines the fault that has occurred through the observations of the two nodes N_1 and N_2 . In this case the network begins to move rapidly away from equilibrium but is arrested by a local controller that has been switched in when the diagnosis algorithm determines that the node N_4 has been completely disconnected from the network, removing a major stabilising component of the network.

We can take a number of key insights from this example. Firstly the importance of being able to accurately and rapidly determine dynamical structure. If this were not possible then it is unlikely we would be able to accurately determine the fault in such a short period of time and switch in appropriate control strategies to ensure synchrony is achieved. Secondly, although this may appear like a simple example there are many possible faults that can occur in the network and it is only by having a diagnosis algorithm that we are able to accurately identify and recover from a chosen fault. Although in both cases the fault that occurs is related to system N_4 through observing and controlling system N_1 and system N_2 we are able to achieve the desired results. This is a major accomplishment as it removes the necessity for local observations and control at every node in the network and still allows

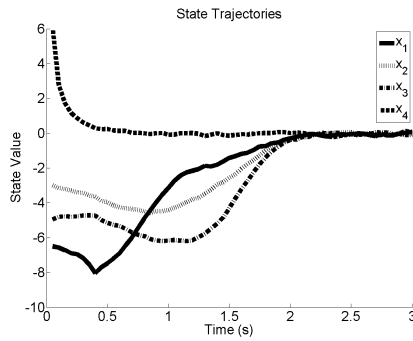


Figure 5: The state trajectories when the node N_4 becomes disconnected from the network due to a fault. The diagnosis algorithm detects the fault and switches in a local control input on node N_1 to ensure network synchrony.

the network performance objectives to be achieved. The optimal location of observation and control is still an area of active research.

5 Conclusion and Future Work

In this paper we have shown how it is possible to use model based diagnosis to create a fault tolerant dynamical network. We have outlined the method by which the two concepts can be unified and motivated the development of this algorithm by showing how this algorithm will offer a substantial improvements in the fault diagnosis and recovery of dynamical networks. Furthermore we have shown how this particular approach will result in the ability to do globally consistent distributed diagnosis. We believe that this methodology is an important advance in the unification of the study of hybrid dynamical networks and advanced, computationally feasible analysis tools. This paper is intended to lay the ground work for the development of advanced algorithms capable of being used in real world scenarios.

The most important area of future work is the implementation of this algorithm with real systems and data. It is likely that implementations will focus on electricity networks as this represents the most urgent need for algorithms and methodologies of this type.

6 Acknowledgements

The authors gratefully acknowledge the guidance, support and advice of Dr Alban Grastien and Professor David Hill.

References

[Blackhall and Rotkowitz, 2008] Lachlan Blackhall and Michael Rotkowitz. Recursive sparse estimation using a gaussian sum filter. In *IFAC 2008*, 2008.

[Blanke, 2006] M. et al Blanke. *Diagnosis and Fault-Tolerant Control (2nd Ed.)*. Springer, 2006.

[Boukhobza et al., 2007] T. Boukhobza, F. Hamelin, and S. Martinez-Martinez. State and input observability for structured linear systems: A graph-theoretic approach. *Automatica*, 43:1204–1210, 2007.

[Cordier and Grastien, 2007] M.-O. Cordier and A. Grastien. Exploiting independence in a decentralised and incremental approach of diagnosis. In M. Veloso, editor, *Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 292–297. AAAI press, 2007.

[Diestel, 2005] Reinhard Diestel. *Graph Theory*. Springer-Verlag Heidelberg, New York, electronic edition, 2005.

[Hofbaur and Williams, 2002] M. Hofbaur and B.C Williams. Mode estimation of probabilistic hybrid systems. In *Proceedings of the 5th International Workshop on Hybrid Systems: Computation and Control*, pages 253–266, 2002.

[Kan John and Grastien, 2008] P. Kan John and A. Grastien. Local consistency and junction tree for diagnosis of discrete-event systems. In *European Conference on Artificial Intelligence (ECAI-08) – to appear*, 2008.

[Khalil, 2002] Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 3rd edition, 2002.

[Kim and Hill, 2008] Sejong Kim and David J. Hill. Synchronization of a complex network with switched coupling. In *IFAC 2008*, 2008.

[Largouet and Cordier, 2000] C. Largouet and M. Cordier. Timed automata model to improve the classification of a sequence of images, 2000.

[Lerner et al., 2002] U. Lerner, R. Parr, D. Koller, and G. Biswas. Bayesian fault detection and diagnosis in dynamic systems. In *Proceedings of the 7th National Conference on Artificial Intelligence (AAAI-00)*, 2002.

[Li et al., 2004] Xiang Li, Xiaofan Wang, and Guanrong Chen. Pinning a complex dynamical network to its equilibrium. *IEEE Transactions on Circuits and Systems-I: Regular Papers*, 51(10):2074–2087, October 2004.

[Ljung, 1999] Lennart Ljung. *System Identification - Theory For the User*. PTR Prentice Hall, Upper Saddle River, N.J., 1999.

[McIlraith et al., 2000] S. McIlraith, G. Biswas, D. Clancy, and V. Gupta. Hybrid systems diagnosis. In *Third International Workshop on hybrid systems*, pages 282–295, 2000.

[Narasimhan et al., 2000] S. Narasimhan, F. Zhao, G. Biswas, and E. Hung. Fault isolation in hybrid systems combining model based diagnosis and signal processing. In *Third International Workshop on hybrid systems*, 2000.

[Pencolé and Cordier, 2005] Y. Pencolé and M.-O. Cordier. A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks. *Artificial Intelligence (AIJ)*, 164:121–170, 2005.

[Su and Wonham, 2005] R. Su and W. M. Wonham. Global and local consistencies in distributed fault diagnosis for discrete-event systems. *Transactions on Automatic Control*, 50(12):1923–1935, 2005.

[Zhao and Hill, 2008] J. Zhao and D. J. Hill. Dissipativity theory for switched systems. *IEEE Transactions on Automatic Control*, 53(4):941 – 953, May 2008.