Multi-Agent Diagnosis with spatially distributed knowledge

Nico Roos [b] Annette ten Teije [c] André Bos [a]
Cees Witteveen [a]

[a] Delft University of Technology, ITS, P.O.Box 256, 2600 AJ Delft.
[b] Universiteit Maastricht, IKAT, P.O.Box 616, 6200 MD Maastricht.
[c] Utrecht University, ICS, P.O.Box 80.089, 3508 TB Utrecht.

Abstract

In a large distributed system it is often infeasible or even impossible to maintain a model of the whole system. Instead, several spatially distributed local models of the system have to be used to detect possible faults. Traditional diagnostic tools cannot handle such a set of spatially distributed local models.

A Multi-Agent System of diagnostic agents, where each agent has a model[1] of a subsystem, may offer solutions for establishing a global diagnosis of a large distributed system. Unfortunately, any protocol that establishes a global minimal diagnosis, is NP-Hard, even if an agent can determine local minimal diagnoses in polynomial time. This paper presents a protocol that enables agents to determine local minimal diagnoses that are consistent with global diagnoses. Moreover, the protocol ensures that no agent acquires knowledge of global diagnoses. The protocol does not guarantee that a combination of the agents' local minimal diagnoses is also a global minimal diagnosis. However, for every global minimal diagnosis, there is a combination of local minimal diagnosis.

1 Introduction

A classical diagnostic tool can be viewed as a *single diagnostic agent* having a model of the whole system to be diagnosed. There are, however, several reasons why in some applications such a single agent approach may be inappropriate. First of all, if the system is physically distributed and large, e.g. modern telecommunication networks, there may be not enough time to compute a diagnosis centrally and to communicate all observations. Secondly, if the structure of the system is dynamic, e.g. AGV systems driving in a platoon, it may change too fast to maintain an accurate global model of the system over time. Finally, sometimes a central model is simply undesirable. For example, if the system is distributed over different legal entities, one entity does not wish other entities to have a detailed model of its part of the system. For such systems, a *distributed* approach of multiple diagnostic agents can offer a solution, as was shown in [12].

The model (knowledge) of a system can be distributed over the agents in two principally different ways[2] (cf. [5]): *(i) spatially distributed*: knowledge of system behavior

[1] We focus on Model-Based Diagnosis.
[2] Combinations are, of course, also possible.

is distributed over the agents according to the spatial distribution of the system's components, and *(ii) semantically distributed*: knowledge of system behavior is distributed over the agents according to the type of knowledge, e.g. a separate model of the electrical and of the thermodynamical behavior of the system. For both types of distributions, a multi-agent system can establish the same global diagnoses as a single diagnostic agent having the combined knowledge of all agents [12].

In this paper we will focus on a spatial distribution of knowledge over the diagnostic agents. We will first formalize the knowledge distribution over the agent based on the well known definitions of Model Based Diagnosis [10, 7]. For simplicity, we do not consider time, though the definition can be extended to dynamic systems. We also do not consider formulations based on Discrete Event Systems [3, 9]. These formulations emphasize more the dynamical aspects of the systems on an abstract level, and especially the occurrence of failure events. As far as failure events can be related to fault modes, discrete event systems can be viewed as a special case of our approach. In section 3, we define multi-agent diagnosis and present some formal results. Section 4 describes a protocol establishing a diagnosis and section 5 concludes the paper.

2 The diagnostic setting

A system to be diagnosed is a tuple $S = (C, M, Id, Sd, Ctx, Obs)$ where C is a set of components, $M = \{M_c \mid c \in C\}$ is a specification of possible fault modes per component, Id is a set of identifiers p of connection points between components, Sd is the system description, Ctx is a specification of input values of the system that are determined outside the system by the environment and Obs is a set of observed values of the system. A component in C has a normal mode $nor \in M_c$, one general fault mode $ab \in M_c$ and possibly several specific fault modes. We assume that all components have *in-* and *outputs*.[3]

The system description $Sd = Str \cup Beh$ consists of a structural description Str and a behavioral description Beh of the components. The structural description Str consists of instances of the form $p = in(x, c)$ or $p = out(x, c)$ where x is an in- or an output identification of a component c and $p \in Id$ is a connection point identifier. Of course, a connection point $p \in Id$ is connected to at most one output of some component; i.e. if $p = out(x, c)$ and $p = out(y, c')$, then $x = y$ and $c = c'$. A connection point has a value, which is determined by the output of a component or a system input. The function $value(p)$ denotes the value of the connection point p.

The set $Beh = \bigcup_{c \in C} Beh_c$ specifies a behavior for each component $c \in C$. The behavior description Beh_c of a component describes the component's behavior for each (fault) mode in M_c, possibly with the exception of $ab \in M_c$. In this specification, the predicate $mode(c, m)$ is used to denote the mode $m \in M_c$ of a component c. For each instance $mode(c, m)$, Beh_c specifies a behavioral description of the form: $mode(c, m) \rightarrow \Phi$ where $m \in M_c$.[4] The expression Φ describes the component's behaviour given its mode $m \in M_c$.

[3]This assumption is not valid in every system. We can, however, transform most systems to a system consisting components with only inputs and outputs (see for instance [4]).

[4]Note that we may use a single description for a class of components. Instances of this description must imply the form of description give here.

The context Ctx describes the values of system inputs $Id_{in} = \{p \in Id \mid \forall x, c : (p = out(x, c)) \notin Str\}$ that are determined by the environment. Ctx consists of instances of the form $value(p) = v$ where v is the value of a connection point $p \in Id_{in}$.

Finally, the set of observations Obs describes the values of those connection points that are observed (measured) by the diagnostic agent. It therefore also consists of instances of the form $value(p) = v$ where v is a value of a connection point $p \in Id$.

A *candidate diagnosis* is a set D of instances of the predicate $mode(,)$ such that for every component $c \in C$ there is exactly one mode in $m \in M_c$ such that $mode(c, m) \in D$. A *diagnosis* is a candidate diagnosis that explains the observed behaviour of a system $S = (C, M, Id, Sd, Ctx, Obs)$ according to our diagnostic definition. In the literature two types of diagnoses are distinguished: *consistency based* [8, 10] and *abductive* [1] diagnosis. Both can be combined into the following, more general, diagnostic definition (cf [2]):

Definition 1 *Let $S = (C, M, Id, Sd, Ctx, Obs)$ be the system to be diagnosed and let $\mid\sim$ to denote the possibly limited reasoning capabilities of a diagnostic system.*[5] *Moreover, let Obs_{con}, $Obs_{\mathrm{abd}} \subseteq Obs$ be subsets of observations and let D be a candidate diagnosis. Then D is a diagnosis for S iff*

1. $D \cup Sd \cup Ctx \mid\sim \bigwedge_{\varphi \in Obs_{\mathrm{abd}}} \varphi$, *2.* $D \cup Sd \cup Ctx \cup Obs_{\mathrm{con}} \not\mid\sim \bot$.

The number of diagnoses can be quite high, exponential in the worst case. In case of consistency based diagnosis, we can characterize the set of diagnoses using a small number of *minimal diagnoses*. D is a minimal diagnosis if for no diagnosis D', $\{mode(c, nor) \mid mode(c, nor) \in D\} \subseteq \{mode(c, nor) \mid mode(c, nor) \in D'\}$.

3 Multi agent diagnosis

The knowledge distribution over multiple agents defines a division of a system into several subsystems. When knowledge is *spatially* distributed, the set of components C is partitioned over the agents. So, agent A_i has knowledge about components C_i, and $C = \biguplus_{i=1}^{m} C_i$ where m is the number of agents. This results in the following distribution of knowledge: $Beh_i = \{\xi \in Beh \mid \xi = (mode(c, m) \rightarrow \Phi), c \in C_i\}$, $Str_i = \{(p = in(x, c)) \in Str \mid c \in C_i\} \cup \{(p = out(x, c)) \in Str \mid c \in C_i\}$ and $Obs_i = \{(value(p) = v) \in Obs \mid (p = out(x, c)) \in Str, c \in C_i\}$. Note that we do not have to split up the context Ctx.

By distributing knowledge, i.e. Beh_i and Str_i over the agents, we loose the knowledge about the connections between components managed by different agents. So, we provide each agent with information about connection points that connect to components managed by other agents and we split the set connection points into relative inputs In_i and outputs Out_i of the agent's subsystem. Here, $In_i = \{p \in Id \mid \{p = in(x, c), p = out(y, c')\} \subseteq Str, c \in C_i, c' \notin C_i\}$ and $Out_i = \{p \in Id \mid \{p = out(x, c), p = in(y, c')\} \subseteq Str, c \in C_i, c' \notin C_i\}$. Hence, $S_i = (C_i, M, Id, Sd_i, Ctx, Obs_i, In_i, Out_i)$ is a subsystem to be diagnosed by the agent. A candidate diagnosis of the subsystem S_i is denoted by D_i.

[5] I.e $\{\varphi \mid \Sigma \mid\sim \varphi\} \subseteq \{\varphi \mid \Sigma \vdash \varphi\}$.

The diagnosis of one agent Each agent A_i in the multi-agent system must make a diagnosis of the subsystem $S_i = (C_i, M, Id, Sd_i, Ctx, Obs_i, In_i, Out_i)$. This can be viewed a single agent diagnosis if values of the inputs and outputs of the subsystem are known. We use the set V_i to denote value assignments $value(p) = v$ with $p \in In_i$ to the inputs. V_i is the local context of the subsystem S_i that is determined by the outputs of other subsystems. We therefore extend Definition 1 to diagnosis of subsystems.

Definition 2 *Let $S_i = (C_i, M, Id, Sd_i, Ctx, Obs_i, In_i, Out_i)$ be a subsystem to be diagnosed and let V_i be a (partial) descriptions of the values of the connection points In_i. Finally, let D_i be a candidate diagnosis.*

Then D_i is a diagnosis for S_i iff D_i is a diagnosis for $(C_i, M, Id, Sd_i, Ctx \cup V_i, Obs_i)$.

The diagnosis of multiple agents Given multiple diagnostic agents, an important question is how the diagnoses of the agents relate to the diagnoses of a single agent that has complete knowledge of the system description and the observations. When addressing this question we assume throughout the paper that *there are no conflicts between the knowledge of the different agents*. That is, there always exists a diagnosis D such that $D \cup Sd \cup Cxt \cup Obs$ is consistent.

Proposition 1 *Let $S_1, ..., S_k$ be the subsystems that make up the system S. Moreover, let D be a single agent diagnosis of S.*

Then $V_i = \{(value(p) = v) \mid p \in In_i,\ D \cup Sd \cup Ctx \mathrel{|\!\sim} (value(p) = v)\}$ is the local context of S_i that is determined by the other subsystems S_j, and $D_i = \{mode(c, s) \mid c \in C_i, mode(c, s) \in D\}$ is a diagnosis of S_i. [12]

Proposition 2 *Let $S_1, ..., S_k$ be the subsystems that make up the system S. Moreover, let the local context V_i of S_i describe the values of connection points in In_i that must be determined by the other subsystems S_j, and let D_i be a diagnosis of S_i determined by agent A_i given V_i.*

Then, $D = \bigcup_{i=1}^{k} D_i$ is a single-agent diagnosis if 1. D is a candidate diagnosis, 2. $D_i \cup Sd_i \cup Ctx \cup V_i \mathrel{|\!\sim} (value(p) = v)$ and 3. for every $p \in Out_i$, $p \in In_j$ and $(value(p) = v) \in V_j$. [12]

The above propositions show that, in principle, multi-agent diagnosis is possible.

Complexity If knowledge is spatially distributed, each agent manages a different part of the system. The behavior of a subsystem managed by an agent depends on the behavior of the other subsystems. This makes it difficult to predict the behavior of the whole system. The values of the connection points in Out_i depend on the local context V_i. The values specified by V_i, however, are determined by other subsystems S_j whose local context V_j may depend on the values of the connection points in Out_i. Because of these circular dependencies, predicting the systems behavior becomes an NP-Hard problem. To see why this is the case, consider a CSP consisting of variables, domains and constraints. We can use In_i to represent a variable, (Sd_i, D_i, Ctx, Obs_i) to represent a constraint and V_i to represent a variable assignment from the domain.

Theorem 1 *Given a global candidate diagnosis D, predicting the values of all connection point is an NP-Hard problem. [12]*

To avoid solving such a hard problem for every candidate diagnosis, consistency based diagnosis and consistency based diagnosis with abductive explanation of normal observations are preferred. These approaches do not apply to fault models. Nevertheless, we still have to solve one NP-Hard problem to predict the *normal* behavior of the system. We may avoid predicting the normal behavior if, at some abstract level, we can assume default values for the connection points. This also reduces the amount of information exchange.

Distributing the diagnostic process After observing abnormal behavior of the system, the agents must make a diagnosis. In order to do so, each agent must make a local diagnosis in which it also takes into consideration the correctness of those inputs of its subsystem that are determined by other agents. Therefore, we must extend a candidate diagnosis D_i of agent A_i with *correctness assumptions* Ca_i about the systems inputs. For every input $p \in In_i$, Ca_i contains either the proposition $correct(p)$ or $\neg correct(p)$. The *conditional context* Cc_i will be used to describe inputs of a subsystem S_i, i.e. the local context of the subsystem determined by other subsystems, conditional to these correctness assumptions, i.e., $Cc_i = \{correct(p) \leftrightarrow (value(p) = v) \mid value(p) \in V_i\}$.

If in its local diagnosis (D_i, Ca_i), agent A_i assumes that one of its inputs is incorrect, the agent must communicate this information to an agent A_j determining the input. Next, agent A_j may treat this information as an observation of one of its outputs, and adapt its local diagnosis accordingly.

A problem with this approach is the occurrence of loops. Suppose that agent A_i blames an observed anomaly on one of its inputs determined by the subsystem of an agent A_j. Agent A_j may also blame the fault in the output determining the input of S_i on one of its inputs. If this input is determined by an output of the subsystem of agent A_i, we may have a cycle of blames that supports itself. Clearly, a local diagnosis that constitutes such cycles of blames does not represent a valid diagnosis of the system. Moreover, handling such loops is a non trivial task which requires tracking dependencies between components in local diagnoses. In fact, the handling of loops causes the determination of minimal diagnoses to be an NP- hard problem[6].

Theorem 2 *Even if the agents have a polynomial algorithm for determining a local minimal diagnosis, determining a global minimal diagnosis is still an* NP-Hard *problem. [12]*

4 The protocol

We wish to design a protocol that will enable each agent to determine all its local minimal diagnoses such that each local minimal diagnosis is consistent with a global diagnosis of a single agent having the combined knowledge of all agents. None of the local agents should, however, be able to determine a global diagnosis and preferably not even be able to determine the subsystem causing the observed anomalies.

Since diagnoses can be derived from conflict sets [10, 8, 7], and since conflict sets contain the dependencies needed for handling the problem with loops described in the previous section, we propose a protocol based on determining local conflict sets. Conflict

[6]by the reduction from the Minimum Feedback Arc Set problem.

sets can be derived by determining dependency sets for predicted values of connection points. The use of dependency sets also makes it possible to apply abductive explanation of *normal* observations, thereby reducing the number of possible diagnoses.

Definition 3 *A dependency set of connection point $p \in Id$ of a subsystem S_i is defined as the* smallest *set $Dep(p) \subseteq \{mode(c, nor) \mid c \in C\} \cup \{correct(p) \mid p \in In_i\}$ such that: $Dep(p) \cup Sd \cup Cxt \cup (Obs_i - \{value(p) = v\}) \mid\sim (value(p) = v)$.*
Here, $\mid\sim$ is restricted in such a way that only output values of a component can be derived.

Note that the restriction on $\mid\sim$ is introduced in order to guarantee that only one value can be derived for a connection point. Without this restriction, it is, in principle, possible to predict an exponential number (in $|Ctx \cup Obs|$) of different values $value(p)$ for a connection point p by using the observations Obs, resulting in large communication overhead.

A dependency set $Dep(p)$ for a connection point p is called a *conflict set* if either p is observed to be incorrect or if $correct(p)$ is an element of a conflict set determined by another agent. $Dep(p)$ is called a *confirmation set* if either p is observed to be correct or if $correct(p)$ is an element of a confirmation set determined by another agent. It is not difficult to see that starting from an observed connection point and by informing an agent A_j whether the correctness assumption $correct(p)$ belongs of a connection point $p \in Out_j$ belongs to a conflict or confirmation set determined by agent A_i, the agents determine, in a distributed way, the same *global* conflict and confirmation sets as a single agent having the combined knowledge of all the agent. Note that since conflict and confirmation sets may overlap, an agent A_j may be informed that a connection point $p \in Out_j$ belongs to both a confirmation and a conflict set. These global conflict (and confirmation) sets can be used to determine the consistency based diagnoses (with abductive explanation of normal observations).

An agent can only determine local diagnoses using the local conflict and confirmation sets. These local diagnoses can be combined to form a global diagnosis. Note that when an agent A_i determines its local diagnoses, it must also consider the possibility that no assumption in a conflict set $Dep(p)$ is incorrect if an agent A_j informed A_i that $p \in Out_i$ belongs to a conflict set. A diagnosis of agent A_j need not contain $\neg correct(p)$, and therefore, no component c such that $mode(c, nor) \in Dep(p)$, needs to be broken.

An important issue is, of course, the handling of loops. Agents must detect loops in order to make sure that their local diagnoses do not result in a global diagnosis in which agents blame each other without considering a component to be broken. To detect loops, each agent A_i associates an identification[7] $rid(p)$ with each observed connection point p and with each connection point in $p \in Out_i$. For each connection point in $p \in Out_i$, agent A_i receives the status of the connection point as well as a set of identifications. This set of identifications contains an identification of an observed connection point $o \in Id$ as well as the identifications of every connection point $q \in Out_j$ on the causal path from p to o through which the value of p influences the value of o. The agent A_i uses these identifications to verify whether using the information about the connection point p results is a loop. Suppose that a loop is detected, i.e. the identifications M the agent A_i receives for $p \in Out_i$ contains the identifications associated with a connection point $q \in Out_i$. Then, in order to avoid a loop, elements form $Dep(q)$ are removed from the dependency set determined by p.

[7]The identifications are generated randomly in order to guarantee anonymity.

Protocol of agent A_i

for each connection point p that is observed or is in Out_i;
determine the dependency set $Dep(p)$;
$rid(p) :=$ randomly generated identification;
end;
$Sb := \varnothing$;
$Sk := \varnothing$;
for for each observed connection point p **do**
if value of p is correct **then**
$Sb := Sb \cup \{(Dep(p), \{rid(p)\})\}$
else $Sk := Sk \cup \{(Dep(p), \{rid(p)\})\}$;
end;
for each $(X, M) \in Sb$ **do**
for each $correct(q) \in X$ **do**
send $inform(q,$ 'correct', $M)$ to agent A_j with $q \in Out_j$;
for each $(X, M) \in Sk$ **do**
for each $correct(q) \in X$ **do**
send $inform(q,$ 'possibly incorrect', $M)$ to agent A_j with $q \in Out_j$;
repeat
for each $inform(p, status, M)$ received from agent A_j **do**
$X := Dep(p) - \bigcup_{(Y,N)\in(Sb\cup Sk), N\subseteq M} Y$;
if $status =$ 'possibly incorrect' **then**
if $X = \varnothing$ **then**
send $reject(p,$ 'possibly incorrect',M) to agent A_j
else $Sk := Sk \cup \{X, M \cup \{rid(p)\})\}$;
else if $X \neq \varnothing$ **then**
$Sb := Sb \cup \{(X, M \cup \{rid(p)\})\}$;
for each $correct(q) \in X$ **do**
send $inform(q, status, M \cup \{rid(p)\})$ to agent A_j with $q \in Out_j$;
end;
for each $reject(p,$ 'possibly incorrect',M) received form agent A_j **do**
replace $(X, M) \in Sk$ with $correct(p) \in X$ by $(X - \{correct(p)\}, M)$;
until no more changes;
determine all local minimal diagnoses and put them in Ld; (1)

end.

The correctness of the above protocol follows from the following propositions.

Proposition 3 *The combination of local dependency sets underlying the conflict and confirmation sets form the global dependency sets of the system. Moreover, a global conflict set results in local conflict sets and a global confirmation set results in local confirmation sets.*

Proposition 4 *The protocol guarantees that the agents reach a stable state in time polynomial in the number of global conflict and confirmation sets, provided that the agents have a polynomial algorithm for determining their local diagnoses; line (1) of the protocol.*

These propositions enables us to derive the following theorem, implying the correctness of the protocol.

Theorem 3 *For each global minimal diagnosis D based on global conflict (and confirmation) sets, there are local minimal diagnoses D_i based on local conflict (and confirmation) sets such that $D = \bigcup_i D_i$.*

Establishing the local minimal diagnoses, line (1) of the protocol, is an NP-Hard problem which may undermine the applicability of the above proposed protocol. Moreover, determining the global minimal diagnoses using the local minimal diagnoses is, according to Theorem 2, also an NP-Hard problem. Approaches that have been developed for handling complexity problems in the single agent [6, 11] case can be adapted to handle these complexity problems.

5 Conclusion

Multi-agent diagnosis of spatially distributed subsystems is an NP-Hard problem. This does not only imply a high time complexity, but also a high communication overhead. Especially the latter makes it impossible to apply multi-agent diagnosis when many agents are involved.

This paper has formally defined multi-agent diagnosis and has presented a protocol for determining all consistency-based diagnoses. The protocol uses a subroutine that has an exponential time complexity. Approaches that exchange diagnostic precision for speed that have been developed for the single agent case, can be used to reduce the time complexity of the proposed protocol.

References

[1] L. Console and P. Torasso. Hypothetical reasoning in causal models. *International Journal of Intelligence Systems*, 5:83–124, 1990.

[2] L. Console and P. Torasso. A spectrum of logical definitions of model-based diagnosis. *Computational Intelligence*, 7:133–141, 1991.

[3] R. Debouk, S. Lafortune, and D. Teneketzis. Coordinated decentralized protocols for failure diagnosis of discrete-event systems. *Journal of Discrete Event Dynamical Systems: Theory and Application*, 10:33–86, 2000.

[4] J. J. van Dixhoorn. Bond graphs and the challenge of a unified modelling theory of physical systems. In F. E. Cellier, editor, *Progress in Modelling & Simulation*, pages 207–245. Academic Press, 1982.

[5] P. Frohlich, I. de Almeida Mora, W. Nejdl, and M. Schroeder. Diagnostic agents for distributed systems. In J.-J. Ch. Meyer and P.-Y. Schobbens, editors, *Formal Models of Agents. LNAI 1760*, pages 173–186. Springer-Verlag, 2000.

[6] J. de Kleer. Focusing on probable diagnosis. In *AAAI-91*, pages 842–848, 1991.

[7] J. de Kleer, A.K. Mackworth, and R. Reiter. Characterizing diagnoses and systems. *Artificial Intelligence*, 56:197–222, 1992.

[8] J. de Kleer and B. C. Williams. Diagnosing multiple faults. *Artificial Intelligence*, 32:97–130, 1987.

[9] Y. Pencolé, M. Cordier, and L. Rozé. Incremental decentralized diagnosis approach for the supervision of a telecommunication network. In *DX01*, 2001.

[10] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–95, 1987.

[11] N. Roos. Efficient model-based diagnosis. *Intelligent System Engineering*, pages 107–118, 1993.

[12] N. Roos, A. ten Teije, A. Bos, and C. Witteveen. An analysis of multi-agent diagnosis. In *AAMAS 2002*, 2002.