IPN Progress Report 42-149 May 15, 2002

New Approaches for Solving the Diagnosis Problem

A. Fijany,[1] F. Vatan,[1] A. Barrett,[1] and R. Mackey[1]

Over the past decade, the number of Earth orbiters and deep-space probes has grown dramatically and is expected to continue to do so in the future as miniaturization technologies drive spacecraft to become more numerous and more complex. This rate of growth has brought a new focus on autonomous and self-preserving systems that depend on fault diagnosis. Although diagnosis is needed for any autonomous system, current approaches are almost uniformly ad hoc, inefficient, and incomplete. Systematic methods of general diagnosis exist in literature, but they all suffer from two major drawbacks that severely limit their practical applications. First, they tend to be large and complex and hence difficult to apply. Second and more importantly, in order to find the minimal diagnosis set, i.e., the minimal set of faulty components, they rely on algorithms with exponential computational cost and hence are highly impractical for application to many systems of interest.

In this article, we propose a two-fold approach to overcoming these two limitations and to developing a new and powerful diagnosis engine. First, we propose a novel and compact reconstruction of the general diagnosis engine (GDE) as one of the most fundamental approaches to model-based diagnosis. We then present a novel algorithmic approach for calculation of the minimal diagnosis set. Using a powerful yet simple representation of the calculation of the minimal diagnosis set, we map the problem onto two well-known problems—that is, the Boolean satisfiability and 0/1 integer programming problems. The mapping onto the Boolean satisfiability problem enables the use of very efficient algorithms with a superpolynomial rather than an exponential complexity for the problem. The mapping onto the 0/1 integer programming problem enables the use of a variety of algorithms that can efficiently solve the problem for up to several thousand components. These new algorithms are a significant improvement over the existing ones, enabling efficient diagnosis of large, complex systems. In addition, the latter mapping allows one, for the first time, to determine the bound on the solution, i.e., the minimum number of faulty components, before solving the problem. This is a powerful insight that can be exploited to develop yet more efficient algorithms for the problem.

---

[1] Exploration Systems Autonomy Section.

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

I. Introduction

The diagnosis of a system is the task of identifying faulty components that cause the system not to function as intended. The diagnosis problem arises when some symptoms are observed, that is, when the system's actual behavior is in contradiction with the expected behavior. The solution to the diagnosis problem is then determination of the set of faulty components that fully explains all the observed symptoms. Of course, the meaningful solution should be a minimal set of faulty components since the trivial solution, which assumes all components are faulty, always explains all inconsistencies.

Model-based diagnosis, first suggested by Reiter [1] and later expanded more by de Kleer, Mackworth, and Reiter [2], is the most disciplined technique for diagnosis of a variety of systems. This technique, which reasons from first principles, employs knowledge of how devices work and their connectivity in the form of models. As an example, Fig. 1 illustrates a model of a hydrazine propulsion subsystem—in this case for Cassini's attitude thrusters.[2] This model consists of a set of connected components, where the connectors are pipes. To diagnose this device, the pressure sensors are monitored for discrepancies while valves open and close during normal operations.

In model-based diagnosis, the focus is on the logical relations between components of a complex system. So the function of each component and the interconnections between components all are represented as a logical system, called the system description (SD). The expected behavior of the system is then a logical consequence of the SD. This means that the existence of faulty components leads to inconsistency between the observed behavior of the system and the SD. Therefore, the determination of the faulty components (or, diagnosis) is reduced to finding the components for which assumption of their abnormality could explain all inconsistencies.

In summary, the diagnosis process starts with identifying symptoms that represent inconsistencies (discrepancies) between the system's model (description) and the system's actual behavior. Each symptom identifies a set of conflicting components as initial candidates. Minimal diagnosis is the smallest set of components that intersects all candidates sets. Therefore, finding the minimal diagnosis set is accomplished in two steps: first generating candidate sets from symptoms, and then calculating a minimal set of faulty components.

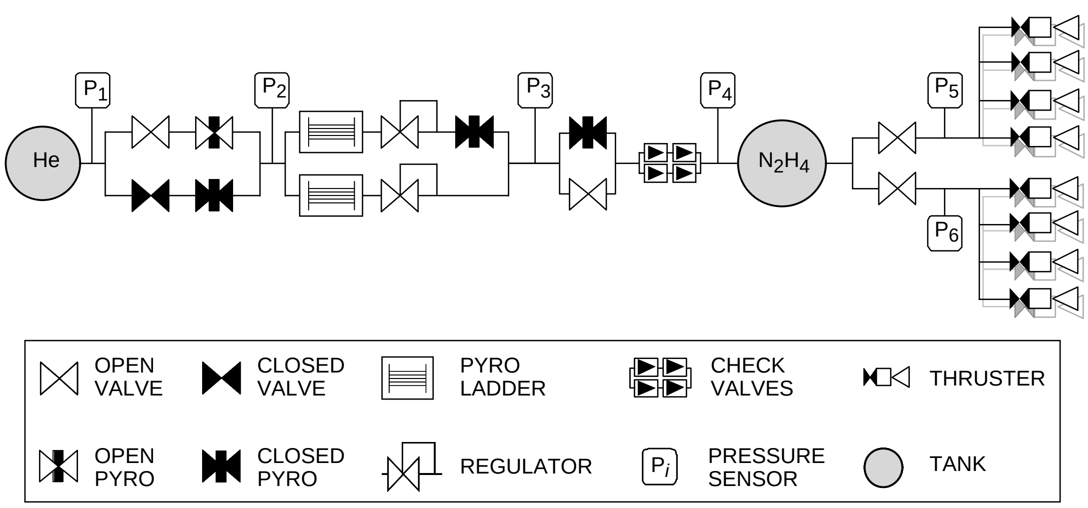

Fig. 1. The hydrazine propulsion subsystem for Cassini's 16 attitude thrusters.

---

[2] *Cassini Program Environmental Impact Statement Supporting Study, Volume 3: Cassini Earth Swingby Plan*, JPL D-10178-3 (internal document), Jet Propulsion Laboratory, Pasadena, California, November 18, 1993.

However, there are two major drawbacks in the current model-based diagnosis techniques that severely limit their practical applications (see also Section II). First, current model-based techniques tend to be large and complex and hence difficult to apply. Second and more importantly, in order to find the minimal diagnosis set, they rely on algorithms with exponential computational cost and hence are highly impractical for application to many systems of interest.

It should be mentioned that a widely employed approach for overcoming limitations of model-based diagnosis techniques is to develop a set of fault protection modes, that is, a set of symptoms-to-cause rules. In this approach, the human experts, by relying on their detailed knowledge of the target system, attempt to predict and analyze all possible faults and to determine their causes. Obviously, in this approach human knowledge is used to overcome the exponential computational complexity. As a result, this approach is time consuming, costly, and prone to human errors (since it is impossible to predict all possible faults in advance). As an example, the development of fault protection modes for Cassini took more than 20 work years.[3]

In this article, we present a novel and two-fold approach for model-based diagnosis to overcome the two above-mentioned limitations and to achieve a powerful engine that can be used for fast diagnosis of large and complex systems. This approach starts with a novel and compact reconstruction of the general diagnosis engine (GDE) as one of the most fundamental approaches to model-based diagnosis. More importantly, we present a novel algorithmic approach for calculation of a minimal diagnosis set. We first discuss the relationship between this calculation and solution of the well-known hitting-set problem. We then discuss a powerful yet simple representation of the calculation of a minimal diagnosis set. This representation enables mapping onto two well-known problems—that is, the Boolean satisfiability and 0/1 integer programming problems. The mapping onto the Boolean satisfiability problem enables the use of very efficient algorithms with superpolynomial complexity for the problem (see Section IV). The mapping onto the 0/1 integer programming problem enables the use of a variety of algorithms that can efficiently solve the problem for up to several thousand components. Therefore, these new algorithms are significant improvements over the existing ones, enabling efficient diagnosis of large, complex systems. In addition, the latter mapping allows one, for the first time, to determine the bound on the solution, i.e., the minimum number of faulty components, before solving the problem. This powerful insight potentially can lead to yet more powerful algorithms for the problem.

This article is organized as follows. In Section II, we review the main notions and concepts of model-based diagnosis by considering the GDE as applied to the hydrazine propulsion subsystem of the Cassini spacecraft. We also briefly discuss a novel and compact reconstruction of the GDE and its advantages. In Section III, we discuss the relationship between the calculation of the minimal diagnosis set and a well-known problem, the hitting-set problem. In Section IV, we discuss a simple representation of the hitting-set problem and calculation of the minimal diagnosis set problem, which allows simple mapping onto the Boolean satisfiability and integer programming problems, discussed in detail in Sections IV and V. In Section VI, we show that the mapping onto the integer programming problem can be used for a priori determination of the bound on the solution, i.e., on the number of faulty components. Finally, some concluding remarks and discussion of future works are presented in Section VII.

II. Model-Based Diagnosis: the General Diagnosis Engine

The general diagnosis engine (GDE) [3] is one of the most fundamental approaches to model-based diagnosis. The GDE combines a model of a device with observations of its actual behavior to detect discrepancies and diagnose root causes. For instance, consider Fig. 1, illustrating a model of a hydrazine propulsion subsystem for Cassini's attitude thrusters. Computationally, each connector corresponds to a set of variables, and each component corresponds to a set of rules for computing variable values of

[3] R. Rasmussen, personal communication, Jet Propulsion Laboratory, Pasadena, California, 2001.

incident connectors from variable values of other incident connectors. In this propulsion example, each pipe corresponds to a pair of variables for pressure and flow. Valve components correspond to rules that conditionally make these variables equivalent, depending on whether or not a valve was commanded open or closed, assuming that the valve is functioning.

The GDE performs a causal simulation by taking variable observations and using rules to compute the values of other variables in the network. Since computations have underlying assumptions, the GDE tags each value with the assumptions that contribute to its computation. A discrepancy arises when two incompatible values are assigned to the same variable. For instance, consider the two open valves between pressure sensors P_1 and P_2 in Fig. 1. Assuming that both valves work gives us identical pressures at each sensor. If the sensors are measuring very different pressures, then at least one sensor or one valve must be faulty. In general, whenever the GDE computes two incompatible values for the same variable, the union of the two supporting assumption sets is incompatible. In this case, assumptions that P_1 and the two valves work imply a pressure value that is incompatible with the pressure value supported by the assumption that P_2 works. Thus, the set of four assumptions leads to contradictory computations.

Typically, in the course of causal simulation, no discrepancies are found, but when failures occur, multiple incompatible assumption sets appear. For instance, continuing our causal simulation to the right from P_2 eventually leads to computing two values at P_3 with different assumption tags. Depending on the value measured by P_3, P_3 combines with one (or both) of these tags to form incompatible assumption set(s). This process continues to determine new incompatible sets until the causal simulation completes. The next step after causal simulation is to the find the minimal set of assumptions that intersects with all detected incompatible sets. This set contains the actual diagnoses of the root causes for contradictory measurements. However, the GDE also suffers from the two main limitations of other model-based diagnosis approaches—that is, the complexity of the software makes its application difficult, and there is an exponential computational cost for finding the minimal set.

In order to overcome the first limitation, we have developed a novel and compact reconstruction of the GDE. Traditionally, the GDE has been implemented using an inference engine to reason about a device model combined with an assumption-based truth maintenance system (ATMS) to keep track of assumptions. A surprising result that arose from our rational reconstruction of the GDE involves merging the ATMS with the inference engine. It turns out that the ATMS and the inference engine have many similarities and that combining the two dramatically simplifies the algorithm. The resultant system was completely implemented in under 150 lines of LISP code! This reconstruction also has some valuable properties for improving reasoning performance. Directly linking the reasoning about a device with reasoning about underlying assumptions facilitates the use of computation-reduction heuristics. For instance, tagging each assumption with a probability results in being able to focus the system on making inferences about high-probability situations that match the current observations.

The second limitation, however, is by far more challenging. For, while it is easy to state the minimal-set determination step, the actual computation solves the provably NP-complete prime implicants problem. In fact, one of the authors has recently shown, for the first time, that the diagnosis problem is NP-complete.[4] However, in the following, we show that, by mapping the diagnosis problem onto the Boolean satisfiability and integer programming problems, algorithms with much better performance and hence with a much wider range of applicability can be devised.

Our current effort on developing a more powerful and practical model-based diagnosis engine builds upon this unique and compact reconstruction of the GDE. In addition, the integration of these novel, efficient algorithms within this reconstruction of the GDE results in a new tool that can efficiently diagnose large systems.

---

[4] F. Vatan, "The Complexity of Diagnosis and Monotone Satisfiability," submitted to *Discrete Applied Mathematics*.

III. Hitting-Set Problem

Our interest in the hitting-set problem is primarily due to its connection with the problem of diagnosis. The hitting-set problem, also known as the transversal problem, is one of the key problems in the combinatorics of finite sets (see [4]) and the theory of diagnosis (see [1,2]). The problem is simply described as follows. A collection $\mathcal{S} = \{S_1, \cdots, S_m\}$ of nonempty subsets of a set M is given. A hitting set (or transversal) of $\mathcal{S}$ is a subset $H \subseteq M$ that meets every set in the collection $\mathcal{S}$; i.e., $S_j \cap H \neq \emptyset$, for every $j = 1, \cdots, m$. Of course, there are always trivial hitting sets; for example, the background set M is always a hitting set. Actually, we are interested in minimal hitting sets with minimal cardinality: a hitting set H is minimal if no proper subset of H is a hitting set.

Note that, for any system $\mathcal{S}$ of subsets of the set $M = \{m_1, m_2, \cdots, m_n\}$, finding one minimal hitting set is easy. We define the sequence $M_0, M_1, \cdots, M_{n+1}$ recursively as follows: let $M_0 = M$; suppose that the set M_j is defined; let $H = M_j \setminus \{m_j\}$, i.e., remove the member m_j from M_j; check whether H is a hitting set; if it is, then let $M_{j+1} = H$; otherwise, let $M_{j+1} = M_j$. Then it is easy to see that each set of the sequence $M_0, M_1, \cdots, M_{n+1}$ is a hitting set and the set M_{n+1} is in fact a minimal hitting set. The more challenging—and more interesting, from both the practical and the theoretical points of view—problem is finding hitting sets of small size. It turns out that this is a hard problem. First let us formalize the hitting-set problem.

Instance: A system $\mathcal{S} = \{S_1, \cdots, S_m\}$ of subsets of the set M and a constant $1/2 < c < 1$.

Question: Is there a hitting set $H \subseteq M$ such that $|H| \leq c\,|M|$?

We should mention that it is well-known that the above problem is NP-complete if the condition is replaced by $|H| \leq K$, where $K \leq |M|$ (see [5]). It is also known that, in this latter form, the problem remains NP-complete even if $|S_j| \leq 2$, for every $1 \leq j \leq m$. Utilizing our results on the complexity of the diagnosis problem, it is possible to show that this stronger form of the problem is NP-complete. In [10], the complexity of several other problems related to hitting sets is investigated.

As mentioned before, we are interested in the hitting-set problem because of its connection with the problem of diagnosis. In fact, as was discussed, each symptom identifies a set of conflicting components as initial candidates, and minimal diagnosis is then the smallest set of components that intersects all candidates sets. The main theorem in the theory of model-based diagnosis [1,2] also states that the minimal diagnoses of the system are exactly the minimal hitting sets of $\mathcal{C}$ (see Fig. 2).

Reiter's hitting-set algorithm [1] is one of the major algorithms for finding minimal hitting sets. The correction of this algorithm is presented in [6] and a modified and more efficient version in [7]. The original algorithm and its modifications are based on generating the lattice of the subsets of the background set M and then extracting a sublattice of it that provides the minimal hitting sets. If the goal is to find a minimal hitting set with minimal cardinality, then this algorithm is not efficient by any means because it requires saving of the whole sublattice, which leads (in the worst case) to the need for an exponential-sized

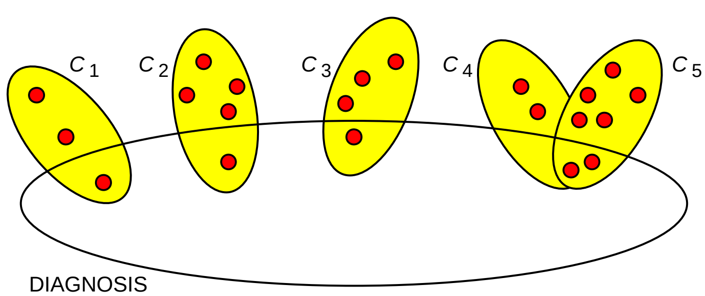

Fig. 2. Diagnosis as the hitting set of the conflicts.

memory to save the sublattice. We will show that it is possible to find a minimal hitting set with minimal cardinality with an algorithm that requires a linear-sized memory (while it still may need an exponential time to complete the computation).

Our approach for solving the hitting-set problem and thus calculation of the minimal diagnosis set is two-fold. On one hand, we map the problem onto the monotone Boolean satisfiability problem. This provides the opportunity of utilizing the superpolynomial algorithms for finding the prime implicants of monotone functions (see [11,8]) and, thus, the minimal diagnosis set. Also, this mapping makes it possible to better understand the complexity of the hitting-set problem, by comparing it with the well-studied Boolean function problems. On the other hand, we map the problem onto an integer programming optimization problem. This simple mapping gives us access to a vast repertoire of integer programming techniques that in some cases can effectively solve problems with several thousands variables. We would like to mention that mapping of the problem of finding prime implicants (not necessarily prime implicants of monotone formulas) onto the integer programming problem has already been introduced; see, e.g., [13,14]. The mappings of the hitting-set problem onto the monotone satisfiability and integer programming problems, which is introduced in this article, provides a new mapping of the problem of finding prime implicants of monotone formulas onto the integer programming problem.

IV. Mapping onto the Boolean Satisfiability Problem

In order to describe mapping of the hitting-set problem onto the Boolean satisfiability and 0/1 integer programming problems, consider a different representation of the problem by describing the attribution of the members (or, components) to subsets (or, initial candidate sets) as given by the following:

$$\begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_m \end{array} \begin{array}{c} \begin{array}{cccc} m_1 & m_2 & \cdots & m_n \end{array} \\ \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 0 \end{pmatrix} \end{array} \tag{1}$$

where $\mathcal{S} = \{S_1, S_2, \cdots, S_m\}$ and $M = \{m_1, m_2, \cdots, m_n\}$ denote the set of nonempty subsets and the set of members (elements), respectively. The (i, j)th entry is denoted as a_{ij}, and we have $a_{ij} = 1$ if m_j belongs to S_i; otherwise, $a_{ij} = 0$. To map the problem onto the Boolean satisfiability problem, we introduce the Boolean variables $x_1, x_2 \cdots, x_n$, where each variable x_j represents the member m_j. Then, to each subset $S_i = \{m_{i1}, m_{i2}, \cdots, m_{in_i}\}$, i.e., each row of Matrix (1), we correspond the disjunction

$$F_i = x_{i1} \vee x_{i2} \vee \cdots \vee x_{in_i} \tag{2}$$

i.e., for each "1" in the ith row of Matrix (1) the corresponding Boolean variable appears in the disjunction, Eq. (2). For example, if the ith row of Matrix (1) is $(0, 1, 1, 0, 0, 1, 0)$, then $F_i = x_2 \vee x_3 \vee x_6$. Then the formula

$$F_{\mathcal{S}} = F_1 \wedge F_2 \wedge \cdots \wedge F_m \tag{3}$$

represents the mapping of the hitting-set problem associated with the system $\mathcal{S}$ onto the Boolean satisfiability problem in the sense that every hitting set of the system $\mathcal{S}$, in a natural way, corresponds with a satisfying truth assignment for the formula $F_{\mathcal{S}}$, and vice versa. Let $(s_1, s_2, \cdots, s_n)$ be a Boolean vector that satisfies the formula $F_{\mathcal{S}}$, and let $S \subseteq M$ be the corresponding set. Then Eq. (2) guarantees that S intersects the set S_i, and Eq. (3) guarantees that S intersects all sets $S_1, S_2, \cdots, S_m$. Thus, S is a hitting set.

We should note that Eq. (3) is in fact monotone. In the case of monotone formulas, the standard form of the satisfiability problem should be slightly modified to avoid the trivial cases. Note that, in the case of the monotone formulas, the all-one vector $(1, 1, \cdots, 1)$ is always a satisfying truth assignment (equivalently, the background set M is always a hitting set). Here, the correct formulation of the problem is to find the assignments with bounded weight or, in the hitting-set setting, the problem is to find hitting sets with a bounded number of members. We have shown that the problem of finding truth assignments for monotone formulas with weight $\leq c\,n$, for $1/2 \leq c < 1$, is NP-complete. Also, the problem of finding minimal hitting sets of the system $\mathcal{S}$ reduces to the problem of finding prime implicants of the monotone function $F_{\mathcal{S}}$.

We should mention here a new result [11,8] that suggests a major breakthrough regarding finding hitting sets in the most general case of the problem. It shows that there is an algorithm that produces the list of prime implicants of a monotone Boolean function such that each prime implicant is produced in a time of $O\big(nt + n^{O(\log n)}\big)$, where t is the time needed to determine the value of the Boolean function at any point. Also, the list produced by this algorithm has no repetitions. The practical implication of this result for the hitting-set problem is that, for the systems that do not have a large number of minimal hitting sets (i.e., there are at most superpolynomially many minimal hitting sets), it is possible to solve the hitting-set problem in superpolynomial time, instead of the exponential time of a typical NP-complete problem.

V. Mapping onto the 0/1 Integer Programming Problem

In order to describe the mapping onto the 0/1 integer programming problem, define the $n \times m$ matrix $A = (a_{ij})_{1\leq i\leq m,\ 1\leq j\leq n}$ associated with the system $\mathcal{S}$, as defined in Matrix (1). Note that, by this definition, each row of A corresponds to a subset and each column to a member. The mapping onto the 0/1 integer programming problem is simply obtained by considering an operator application of A as follows. Identification of a minimal subset of members, representing a minimal hitting set, is equivalent to finding a minimal subset of columns of the matrix A whose summation results in a vector with elements equal to or greater than 1. This can be better described in terms of matrix-vector operation as follows. Let the vector A_i, for $i = 1, \cdots, m$, denote the ith row of the matrix A. Also, define a binary vector $\mathbf{x} = (x_1, x_2, \cdots, x_n) \in \mathbb{R}^n$, wherein $x_j = 1$ if the member m_j belongs to the minimal hitting set; otherwise, $x_j = 0$. Since at least one member should belong to every S_i, for every $i = 1, \cdots, m$, we then have

$$A_i \cdot \mathbf{x} \geq 1$$

Since, by the definition of the minimal subset, the above formula should be simultaneously satisfied for all $i = 1, \cdots, m$, we then have the following formulation of the problem as an 0/1 integer programming problem:

$$\begin{array}{ll} \text{minimize} & \mathrm{wt}(\mathbf{x}) \\ \text{subject to} & A\,\mathbf{x}^T \geq \mathbf{b}^T \end{array} \tag{4}$$

where $\mathbf{b} = (1, 1, \cdots, 1) \in \mathbb{R}^m$ is the all-one vector, and we denote the Hamming weight, i.e., the number of one-components of the binary vector $\mathbf{x}$, by $\mathrm{wt}(\mathbf{x})$. With this setting, identification of the minimal hitting set is then equivalent to solution for the binary vector $\mathbf{x}$ from Formula (4), which corresponds to the solution of the 0/1 integer programming problem.

Note that Formula (4) represents a rather special case of the 0/1 integer programming problem since the matrix A is a binary matrix, i.e., with 1 or 0 elements only. Interestingly, our above derivation also establishes a mapping of the monotone Boolean satisfiability problem onto this special case of the 0/1 integer programming problem. To see this, note that any monotone Boolean satisfiability problem, given by Eq. (3), can be equivalently represented by a matrix similar to Matrix (1), from which the mapping onto this special case of the 0/1 integer programming problem follows immediately.

Example 1. As an example, we consider the small problem solved in [6,7]. Here $M = \{a, b, c, d\}$ and $\mathcal{S} = \big\{\{a, b\}, \{b, c\}, \{a, d\}, \{b, d\}, \{b\}\big\}$. In this case, the incidence Matrix (1) is of the following form:

$$\begin{array}{c} \\ \{a,b\} \\ \{b,c\} \\ \{a,d\} \\ \{b,d\} \\ \{b\} \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

The corresponding 0/1 optimization problem is then formulated as

$$\begin{array}{ll} \text{minimize} & x_1 + x_2 + x_3 + x_4 \\ \text{subject to} & x_1 + x_2 \geq 1, x_2 + x_3 \geq 1, x_1 + x_4 \geq 1, x_2 + x_4 \geq 1, x_2 \geq 1 \end{array}$$

This problem has two solutions:

$$\big\{x_1 = x_2 = 1,\ x_3 = x_4 = 0\big\}$$

$$\big\{x_2 = x_4 = 1,\ x_1 = x_3 = 0\big\}$$

which correspond to the minimal hitting sets $\{a, b\}$ and $\{b, d\}$.

VI. Lower Bounds on Calculation of the Minimal Diagnosis Set

As stated before, the integer programming problem is known to be an intractable problem (see [5]), although there are several reasonably good algorithms that can solve the problem either exactly for a certain size or approximately for any size. However, our recent discoveries of the bounds on the size of the solution of Formula (4) opens a new direction for improving the efficiency of existing algorithms and/or devising new and more efficient algorithms. Here, we briefly describe these new results.

For two vectors $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ and $\mathbf{y} = (y_1, y_2, \cdots, y_n)$ in $\mathbb{R}^n$, we write $\mathbf{y} \geq \mathbf{x}$ if and only if $y_j \geq x_j$, for every $j = 1, \cdots, n$. Also, we consider the 1-norm and 2-norm of vectors defined as

$$\|\mathbf{x}\|_1 = \sum_{j=1}^{n} |x_j|$$

and

$$\|\mathbf{x}\|_2 = \sqrt{\sum_{j=1}^{n} {x_j}^2}$$

For the vector $\mathbf{b}$ in Formula (4), we then have $\|\mathbf{b}\|_1 = m$ and $\|\mathbf{b}\|_2 = \sqrt{m}$. Since the elements of both vectors $A\,\mathbf{x}^T$ and $\mathbf{b}$ in Formula (4) are positive, we can then derive the following two inequalities from Formula (4):

$$\left.\begin{array}{c} \|\mathbf{A}\|_1 \times \|\mathbf{x}\|_1 \geq m \Longrightarrow \|\mathbf{x}\|_1 \geq m/\|\mathbf{A}\|_1 \\ \\ \|\mathbf{A}\|_2 \times \|\mathbf{x}\|_2 \geq \sqrt{m} \Longrightarrow \|\mathbf{x}\|_2 \geq \sqrt{m}/\|\mathbf{A}\|_2 \end{array}\right\} \qquad (5)$$

Since $\mathbf{x}$ is a binary vector, then both norms in Inequality (5) give the bound on the size of the solution, that is, the number of nonzero elements of vector $\mathbf{x}$ which, indeed, corresponds to the minimal diagnosis set. Note that, depending on the structure of the problem, i.e., the 1- and 2-norms of the matrix A and m, a sharper bound can be derived from either member of Inequality (5). To our knowledge, this is the first time that such bounds on the solution of the problem have been derived without any need to explicitly solve the problem. Such a priori knowledge on the size of the solution will be used for developing much more efficient algorithms for the problem.[5]

VII. Conclusion

We proposed a two-fold approach to overcoming the two major limitations of the current model-based diagnosis techniques—that is, the complexity of the tools and the exponential complexity of calculation of the minimal diagnosis set. To overcome the first limitation, we have developed a novel and compact reconstruction of the GDE. To overcome the second and more challenging limitation, we have proposed a novel algorithmic approach for calculation of the minimal diagnosis set. Starting with the relationship between the calculation of the minimal diagnosis set and the celebrated hitting-set problem, we have proposed a new method for solving the hitting-set problem and, consequently, the diagnosis problem. This method is based on a powerful yet simple representation of the problem that enables its mapping onto two other well-known problems—that is, the Boolean satisfiability and 0/1 integer programming problems. The mapping onto the Boolean satisfiability problem enables the use of very efficient algorithms with a superpolynomial rather than an exponential complexity for the problem.

The mapping onto the 0/1 integer programming problem enables the use of a variety of algorithms that can efficiently solve the problem for up to several thousand components. These new algorithms are a significant improvement over the existing ones, enabling efficient diagnosis of large, complex systems. In addition, this mapping allows, for the first time, a priori determination of the bound on the solution, i.e., the minimum number of faulty components, before solving the problem. This is a powerful insight that potentially can lead to yet more powerful algorithms for the problem. It should be mentioned, however, that Formula (4) represents a rather special case of the 0/1 integer programming problem by being specific to the calculation of the minimal diagnosis set, since the matrix A is a binary matrix, i.e., with 1 or 0 elements only, and the vector $\mathbf{b}$ is the all-one vector. We are currently devising new techniques to exploit this special structure of this mapping to develop yet more efficient algorithms, optimized for calculation of the minimal diagnosis set.

Our current effort on developing a more powerful and practical model-based diagnosis engine builds upon the unique and compact reconstruction of the GDE. In addition, the integration of these novel,

[5] A. Fijany and F. Vatan, "New Bounds on Solution of Certain NP-Complete Problems," in preparation.

efficient algorithms within this reconstruction of the GDE enables the development of new tools that can efficiently diagnose large systems.

Acknowledgment

A. Fijany would like to acknowledge N. Askari for insightful discussions and encouragement.

References

[1] R. Reiter, "A Theory of Diagnosis from First Principles," *Artificial Intelligence*, vol. 32, pp. 57–95, 1987.

[2] J. de Kleer, A. K. Mackworth, and R. Reiter, "Characterizing Diagnoses and Systems," *Artificial Intelligence*, vol. 56, pp. 197–222, 1992.

[3] J. de Kleer and B. Williams, "Diagnosing Multiple Faults," *Readings in Model-Based Diagnosis*, San Mateo, California: Morgan Kaufmann Publishers, 1992.

[4] C. Berge, *Hypergraphs: Combinatorics of Finite Sets*, Amsterdam, The Netherlands: Elsevier-North Holland, 1989.

[5] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, New York: W. H. Freeman and Company, 1979.

[6] R. Greiner, B. A. Smith, and R. W. Wilkerson, "A Correction to the Algorithm in Reiter's Theory of Diagnosis," *Artificial Intelligence*, vol. 41, pp. 79–88, 1989.

[7] F. Wotawa, "A Variant of Reiter's Hitting-Set Algorithm," *Information Processing Letters*, vol. 79, pp. 45–51, 2001.

[8] V. Gurvich and L. Khachiyan, "On Generating the Irredundant Conjunctive and Disjunctive Normal Forms of Monotone Boolean Functions," *Discrete Appl. Math.*, vol. 96–97, pp. 363–373, 1999.

[9] T. Bylander, D. Allemang, M. C. Tanner, and J. R. Josephson, "The Computational Complexity of Abduction," *Artificial Intelligence*, vol. 49, pp. 25–60, 1991.

[10] T. Eiter and G. Gottlob, "Identifying the Minimal Transversals of a Hypergraph and Related Problems," *SIAM J. Comput.*, vol. 24, pp. 1278–1304, 1995.

[11] M. Fredman and L. Khachiyan, "On the Complexity of Dualization of Monotone Disjunctive Normal Forms," *J. of Algorithms*, vol. 21, pp. 618–628, 1996.

[12] G. Friedrich, G. Gottlob, and N. Nejdl, "Physical Impossibility Instead of Fault Models," in *Proceedings of Eighth National Conference on Artificial Intelligence AAAI–90*, vol. 1, Cambridge, Massachusetts: MIT Press, pp. 331–336, 1990.

[13] V. M. Manquinho, P. F. Flores, J. P. M. Silva, and A. L. Oliveira, "Prime Implicant Computation Using Satisfiability Algorithms," *Proceedings Ninth IEEE International Conference on Tools with Artificial Intelligence*, IEEE Computer Society, Los Alamitos, California, pp. 232–239, 1997.

[14] L. Palopoli, F. Pirri, and C. Pizzuti," Algorithms for Selective Enumeration of Prime Implicants," *Artificial Intelligence*, vol. 111, pp. 41–72, 1999.