Model-Based Diagnosis of Multi-Agent Systems: A Survey

Meir Kalech and Avraham Natan

Ben - Gurion University of the Negev, Israel
kalech@bgu.ac.il, natanavr@post.bgu.ac.il

Abstract

As systems involving multiple agents are increasingly deployed, there is a growing need to diagnose failures in such systems. Model-Based Diagnosis (MBD) is a well known AI technique to diagnose faults in systems. In this approach, a model of the diagnosed system is given, and the real system is observed. A failure is announced when the real system's output contradicts the model's expected output. The model then is used to deduce the defective components that explain the unexpected observation. MBD has been increasingly being deployed in distributed and multi-agent systems. In this survey, we summarize twenty years of research in the field of model-based diagnosis algorithms for MAS diagnosis. We depict three attributes that should be considered when examining MAS diagnosis: (1) The objective of the diagnosis. Either diagnosing faults in the MAS plans or diagnosing coordination faults. (2) Centralized vs. distributed. The diagnosis method could be applied either by a centralized agent or by the agents in a distributed manner. (3) Temporal vs. non-temporal. Temporal diagnosis is used to diagnose the MAS's temporal behaviors, whereas non-temporal diagnosis is used to diagnose the conduct based on a single observation. We survey diverse studies in MBD of MAS based on these attributes, and provide novel research challenges in this field for the AI community.

Introduction

Multi-Agent Systems (MAS) can be found in wide variety of applications, such as automatic warehousing, autonomous vehicles, logistics and public transport (such as train systems), video games, etc. (Hausman, Schwarz, and Graves 1976; Kato et al. 2015; Runhua et al. 2013; Marín-Lora et al. 2020). Such systems may vary in many characteristics, such as the amount of cooperation between the agents, the relations between them, or how dynamic those systems are. In many of these schemes, agents are assigned to perform tasks in a coordinated manner, in what is called a Multi-Agent Plan (MAP). For example, Amazon warehouses use robots for automated relocation of items in the warehouse, freeing manpower for less trivial tasks (Edwards 2020).

As agents are deployed more and more in complex, dynamic environments, ways to react to failures in multi-agent systems have become more and more important. In some systems agents have to agree on their goals, plans and at least some of their values. In other systems agents should coordinate their interactions. However, agents may disagree or avoid consistent interactions, because of sensory ambiguity, communication defects, mechanical faults, etc. If an inconsistency arises, it needs to be diagnosed and the agents that triggered the coordination breakdown should be identified. This is the task of **diagnosis**.

Model-Based Diagnosis (*MBD*) (Reiter 1987; de Kleer and Williams 1987) is an Artificial Intelligence diagnosis approach. MBD relies on a **model** of the diagnosed system, which is utilized to simulate the behavior of the system given the operational context (typically, the system inputs). The resulting simulated behavior (typically, outputs) are compared to the **observed** behavior to detect discrepancies indicating failures. The model can then be used to pinpoint possible failing components within the system. In multi-agent system domains, the MAS model includes a model of the agents, their plans and the interactions between them, and the observation of the MAS is the observed behaviours of the agents. **The first contribution of this survey** is by presenting 20 years of research in the field of MBD of MAS.

Diagnosing MAS raises several challenges: (1) Typically, diagnosis algorithms do not scale well. Multi-agent systems are usually large and complex and pose a great challenge to diagnosis algorithms. (2) Dynamic structures, like MAS, pose a major challenge to diagnostic algorithms, since they involve dynamic change in the behaviours, multiple observations and interactions that rapidly change. (3) MBD has been difficult to apply to diagnose coordination failures, since many such failures take place at the boundaries between the agents and their environment, including other agents. (4) MAS presents additional challenges as a result of a noisy environment that provokes uncertainty and partial observation.

Diagnosing MAS algorithms discuss different MAS diagnosis attributes and propose solutions accordingly. (1)The first attribute takes into account the goal of the MAS diagnosis. Typically, two objectives have been investigated: the first is connected to the occurrence of defective plan-steps in the global or derived local plan. The second goal looks at coordination faults, which occur when agents disagree on key components of their joint task. (2) The second characteristic of the diagnosis is whether it is applied in a centralized or distributed manner. A single diagnoser models and

observes the entire MAS and infers the diagnosis in a centralized method. In a distributed approach, each agent has a partial model and observes the system in part; then, all of the agents communicate information to arrive at a global diagnosis. (3) The third diagnosis attribute asks the question whether to consider temporal aspects of the MAS or not. Considering temporal aspects in the inference process means either using various observations over time or considering possible agent behaviors over time. **The second contribution of the survey** is by organizing the different MBD methods of MAS according to these three attributes.

In this survey, we first explain the main challenges that the above MAS diagnosis aspects pose (Section). In Section we present a literature review of MAS diagnosis approaches. We divide this review between algorithms that consider multi-agent plan diagnosis (Section) and MAS coordination diagnosis (Section). Note that we consider only works that deal with diagnosing faults in multi-agent systems, rather than studies that use MAS to diagnose faults in other systems, as lastly proposed (El Koujok et al. 2020; Srivastava, Bhat, and Singh 2020). Then in Section , we present research opportunities in MAS diagnosis, **this is the third contribution of this survey**. There are six open research directions: diagnosing MAS with uncertain observation, troubleshooting MAS failures, recover from failures, integrating machine-learning approaches, diagnosing MAS with privacy constraints and diagnosing intermittent faults in MAS. Section concludes.

Attributes of MAS Diagnosis

Diagnosing MAS algorithms consider different aspects of the MAS diagnosis problem and propose solutions appropriately. Next we explain these different aspects.

Planing and Coordination Faults

Parker and Lynne (Parker 2012) consider several faults a multi-agent system might encounter: individual agent malfunctions, local perspectives that are globally incoherent, inter-agent interference, software errors or incompleteness, and communications failures. In this study we focus on two fundamental types of faults, which are at the core of the "multi" aspect of multi-agent systems: planning related faults and coordination related faults. The other types of faults described in (Parker 2012), such as communication failures or incorrect local perceptions, will eventually evolve to, or be the result of either planning related faults or coordination related faults (Khalastchi and Kalech 2019).

Planning Related Faults: Planning related faults introduce the occurrence of faulty plan-steps either in the global or the derived local plan. A faulty plan-step is an instruction which may lead to mission failure, degraded performance, or waste of resources such as energy. For instance, in a foraging domain, a faulty step of the global plan resulted in the following task allocation: two agents forage in the same area, following a predefined plan. Although we can assume that their plans are coordinated, it may happen that the agents will interfere each other by allocating the same place simultaneously. This fault may be caused by a faulty behaviour of each one of the agents, deviated from its plan. A diagnosis mechanism is challenged to identify the faulty plan-step and time as the root cause. It is challenging since there could be many time steps along the plan as well as many agents that may cause the fault. Also, a faulty agent may cause another healthy agent a delay in his plan, which in turn may cause a third agent an additional delay, and so on. Diagnosing the root cause agent in such a chain reaction is challenging.

A known research field that considers the synchronization challenge is Multi-Agent Path Finding problem (MAPF) (Pallottino et al. 2007; Erdem et al. 2013; Surynek et al. 2016; Švancara et al. 2019; Barták et al. 2019; Barták, Švancara, and Krasičenko 2020; Li, Ruml, and Koenig 2021). MAPF describes the problem of moving agents to destinations while avoiding collisions. For example, Amazon warehouses use robots for automated relocation of items in the warehouse, freeing manpower for less trivial tasks. In another example (Švancara et al. 2019; Ma and Li 2021) an autonomous intersection management (AIM) is presented. In such systems, the agents are assigned to execute tasks. This assignment is called Multi-Agent Plan (MAP).

In such environments agents often share reusable resources such as doorways, charge points, moving space etc. for a given amount of time. In case of a fault, an agent may result in holding a resource for longer amount of time than planned, and creating a chain reaction of agents failing to follow their original plans, which could result in halting of a production process or worse. This raises the necessity to isolate the faulty agents and the time at which they failed.

Coordination Related Faults: Coordination (e.g., on a joint plan or goal) is a key to establishment and maintenance of teamwork (Tambe 1997; Pynadath and Tambe 2003; Sycara and Sukthankar 2006; Geihs 2020). One type of failure of particular interest in multi-agent systems is a *coordination fault*, where agents come to disagree on salient aspects of their joint task. There is thus a particular need to be able to detect and diagnose the causes for coordination faults that may occur, to facilitate recovery and reestablishment of collaboration, e.g., by negotiations (Beer et al. 1999; Calvaresi et al. 2018). This type of diagnosis is called *social diagnosis*, since it focuses on finding causes for failures to maintain social relationships (Franchi and Poggi 2012), i.e., coordination failures.

Coordination failures take place at the boundaries between the agents and their environment, including other agents. For instance, in a team, an agent may send a message that another agent, due to a broken radio, did not receive. As a result, the two agents come to disagree on an action to be taken. Lacking an omniscient diagnoser that knows of the sending of the message, the receiver has no way to detect and diagnose its fault, since the context—the message that can be fed into a model of the radio of both agents—is unobservable to the diagnoser.

It is possible to diagnose coordination failures, given the actions of agents, and the coordination constraints that should ideally hold between them. In the example above, knowing that the two agents should be in agreement as to their actions, and observing that their actions are not in

agreement, is sufficient to (1) show that a coordination failure has occurred; and (2) to propose several possible diagnoses for it (e.g., the first agent did not send a message, the second agent did not receive it, etc.). However, the diagnosis task is challenging since agents could blame each other, which increasing the number of possible diagnoses.

Centralized vs. Distributed Diagnosis

Most approaches in model–based diagnosis depict centralized diagnosis, where a single observer makes the diagnosis (de Kleer and Williams 1987; Metodi et al. 2014). On the other hand, there are a few approaches to distributed diagnosis, where multiple observers distribute the diagnosis task among them (Su et al. 2002; Lo, Lynch, and Liu 2016). In this case, each diagnoser observes part of the system and infers local diagnosis. Then, they coordinate to share information and reach a global diagnosis.

In MAS diagnosis, most approaches present centralized diagnosis, but there are some distributed approaches. A distributed approach for MAS is motivated even more than in regular systems, since MAS is a distributed system by definition. Moreover, a single diagnosing agent is a single point of failure. Distributing the diagnosing process among the agents may overcome such difficulties.

Temporal vs. Non-Temporal Faults

Many approaches in model-based diagnosis do not consider temporal aspects of a system. In such cases, a system can be modelled and diagnosed without the need to consider or model temporal relations between the agents. On the other hand, there are approaches that consider temporal aspects of the actions and the relations between components (Brusoni et al. 1998; Bunte, Stein, and Niggemann 2019). In those cases, when modelling the system, there is importance to the time and order at which actions, including faults, take place.

When diagnosing MAS, temporal aspects may or may not be taken into account. The nature of the diagnostic system has a considerable bearing on this. Non-temporal systems are depicted by one type of MAS. In such systems, regular execution refers to the agents' agreement on a specific behavior. An example can be seen in a work where agent teams coordinate their plans (Kalech and Kaminka 2007). In that work, coordination manifests in the agreement of agents on the execution modes they are in. Another type of MAS depicts systems in which the temporal relations have great importance in the system modelling. Such systems are highly sensitive to the order of events. For instance, an agent that experiences delay in taking a box from point A to point B, might impact the execution of another agent, that is supposed to take the same box from point B to point C. In such approaches considering temporal aspects is major part of the modelling of the system, and relations between actions of different agents should be taken into consideration. An example can be seen in (Micalizio and Torasso 2014). The authors present a system where agents use resources in an office environment (rooms, desks, doors, parking points etc.). In this example, the agents work cooperatively, and when a fault occurs, the modelled system needs to have perspective of temporal order between the different events, to track down the root cause of the failure in the execution of the plan.

Single vs. Multiple Observation Times

MAS diagnosis algorithms are based on observation(s) of the system. They compare between the expected behaviour of the agents and the observed behaviour to infer which agents behaved abnormally. The greater the number of observation times, the more likely the diagnosis can be isolated. In the extreme case, where all agents are observed throughout all time steps, it is possible to conclude the exact agents that caused the MAS to fail. Some of the previous work assume multiple observation time points during the agents' execution, and deduce a diagnosis that is consistent with all observations. Other studies assume a single observation time, and provide diagnosis algorithms based on that observation.

A Survey of MAS Diagnosis

Previous research has looked on the diagnosis of multi-agent system failures. We divide this survey into two topics, based on the main attribute of the diagnosis MAS, described above: diagnosing MAS plans (Section), and diagnosing coordination faults (Section). But first we introduce the fundamentals of MBD.

Model-Based Diagnosis: Background

An MBD problem arises when the normal behavior of a system is violated due to faulty components as indicated by certain observations. An MBD problem is specified by the tuple $\langle \mathit{SD}, \mathit{COMPS}, \mathit{OBS} \rangle$ where: *SD* is a system description, *COMPS* is a set of components, and *OBS* is an observation. *SD* takes into account that some components might be abnormal (faulty). This is specified by the unary predicate $h(\cdot)$ on components such that $h(c)$ is true when component c is healthy, while $\neg h(c)$ is true when c is faulty. A diagnosis problem arises when the assumption that all components are healthy is inconsistent with the system model and observed system behavior. This is expressed formally as follows: $\mathit{SD} \wedge \bigwedge_{c \in \mathit{COMPS}} h(c) \wedge \mathit{OBS} \vdash \bot$.

Diagnosis algorithms try to find diagnoses, which are possible ways to explain the above inconsistency by assuming that some components are faulty. A component set $\triangle$ is a diagnosis if: $\mathit{SD} \wedge \bigwedge_{c \in \triangle} \neg h(c) \wedge \bigwedge_{c \notin \triangle} h(c) \wedge \mathit{OBS} \nvdash \bot$.

Diagnosis of Plan Related Faults

In this section, we cover works related to the diagnosis problem in multi-agent systems plan; i.e. the plan performed by the agents is considered as part of the system description (*SD*), the agents are components (*COMPS*) and the observation (*OBS*) includes the actions of the agents.

Roos and Witteveen (Roos and Witteveen 2009) investigate this problem. They introduce a formal model where partial observations of plan states are compared with predicted states based on a model of the expected plan execution. Deviation between observed and predicted states can be explained by faulty plan(s). The diagnosis is a subset of

abnormal plan steps that can explain the incompatibility between the predicted and the observed state. They show how these diagnoses can be found efficiently if the plan is distributed over a number of agents.

Extending Roos and Witteveen's notion, de Jonge et al. (De Jonge, Roos, and Witteveen 2009) introduce the use of model-based diagnosis in two general types of plan diagnosis: primary plan diagnosis identifies the incorrect or failed execution of actions, and secondary plan diagnosis identifies the root cause of these faulty actions. The primary diagnosis is linked to the secondary diagnosis and thus the root cause (e.g., agent, equipment) of failed plan steps can be diagnosed.

Micalizio and Torasso (Micalizio and Torasso 2007) also address the problem of MAS plan diagnosis, aiming to find the faulty agents. They model the normal and abnormal execution of actions and propose a mechanism for identifying how agent faults affect the functionality of the MAS. By modelling of the agents, they assume that agents monitor their behavior through series of observations as part of the diagnosis.

In another work, Micalizio (Micalizio 2009) proposes a distributed approach to autonomous plan repair. Each agent executes a local plan that is derived from the global multi-agent plan. The agents autonomously monitor, diagnose, and repair their local plan. Since the system is only partially observed, the state of an agent is not certain but rather estimated. Thus, the diagnosis is typically ambiguous, and thus the repair or re-planning step must handle uncertainty. They show that the proposed methodology is adequate to promptly react to an action failure and that the computational cost of the approach is affordable since the agent diagnosis highly constrains the search for a recovery plan.

Another approach to plan related faults (Micalizio and Torta 2012) focuses on temporal aspects of a diagnosis and tries to explain delays in the execution of the multi agent plans by assigning an execution modes to actions (e.g. nominal, faulty1, faulty2,. . .) that are essentially time lengths that an action took, in order to explain consistently the observations received throughout the plan.

In a later work, Micalizio and Torasso (Micalizio and Torasso 2014) presented a novel methodology, named Cooperative Weak-Committed Monitoring (CWCM), where the diagnosis of the multi-agent plan, executed in a dynamic and partially observable environment, is addressed in a fully distributed and asynchronous way. As opposed to previous approaches the action failures are not assumed as independent of each other. CWCM exploits non-deterministic action models to carry out two main tasks: detecting action failures and reconstructing possible beliefs an agent has had about the environment. Thus, each agent has the ability for self-diagnosis in terms of explaining action failures as exogenous events. A diagnostic engine is utilized for distinguishing primary and secondary action failures. They show that CWCM is effective in identifying and explaining action failures even when the observability of the system is significantly reduced.

In the recent series of studies by Torta et al. (Torta and Micalizio 2018; Torta, Micalizio, and Sormano 2019a,b), the authors focus on temporal aspects of a diagnosis similarly to (Micalizio and Torta 2012). Those works focus on explaining failures that happen during the execution of Temporal MAP (TMAP), in a way that explains how root failures propagate into later stages of the execution. The model of the system contains normal as well as abnormal execution modes of the actions, and the diagnosis algorithm identifies actions that were executed in faulty mode as well as actions that executed abnormally as a result of failure propagation.

A distributed diagnosis approach is proposed by Qin et al. (Qin et al. 2018). A distributed fault detection (DFD) unit and a distributed fault isolation (DFI) unit are included in each agent. A model of each agent's interactions with its neighbors is supplied, which is used to discover faults and isolate them using a residual generator. The agents are able to isolate the global fault by interacting this information.

Diagnosis of Coordination Faults

In this section, we cover works related to the diagnosis of coordination faults, i.e., faults which prevent or disturb the ability of agents in the system to coordinate their actions. Coordination faults are not usually relevant to adversary agents, but typically could appear in systems with cooperative agents. The components (*COMPS*) and the observation (*OBS*), in this case are the same as in diagnosis of MAS plans, but the system description (*SD*) includes a description of the coordination between agents.

Early studies in this subject depict centralized architectures. For instance, Micalizio et al. (Micalizio, Torasso, and Torta 2004) have utilized causal models of failures and diagnoses to centrally detect and respond to single-agent failures and to multi-agent coordination failures. Unfortunately, a centralized architecture can be computationally expensive in terms of communications and run-time, and there is a single point of failure as the diagnosing agent might fail. Distributed approaches address this limitation.

Roos et al. (Roos, Ten Teije, and Witteveen 2003) present model-based diagnosis methods for spatially distributed agents, where each agent is responsible for diagnosing a different subsystem of the MAS. Every agent makes a local diagnosis to its own sub-system and then all agents compute a global diagnosis. However, while building the global diagnosis set there is an assumption that there are no conflicts between the knowledge of the different agents, i.e., that no coordination faults occur.

The following works of Kalech and Kaminka explicitly tackle the problem of diagnosing coordination faults. Continuing with their centralized approach (Kalech and Kaminka 2005), Kalech et al. introduce a distributed model-based coordination-failure diagnosis approach (Kalech et al. 2006). In their work, the coordination between the agents is modeled as a constraint graph. For the diagnosis, they utilize different distributed CSP (Constraint Satisfaction Problem) algorithms. They conclude that there is a trade-off between the effectiveness of the algorithms, in terms of communication and computation, and the correctness of the diagnosis that the algorithms produce. In those works, the diagnosis process tries to explain inconsistencies in the constraint graph isolating agents that do not behave as expected.

	Fault Type	Centralized/ Distributed	# Observations	Temporal?
(Roos and Witteveen 2009)	Plan related	Both	Many	Temporal
(De Jonge, Roos, and Witteveen 2009)	Plan related	Centralized	Many	Temporal
(Micalizio and Torasso 2007)	Plan related	Distributed	Many	Temporal
(Micalizio 2009)	Plan related	Distributed	Many	Temporal
(Micalizio and Torta 2012)	Plan related	Unspecified	Many	Temporal
(Micalizio and Torasso 2014)	Plan related	Distributed	Many	Temporal
(Torta and Micalizio 2018)	Plan related	Unspecified	Many	Temporal
(Torta, Micalizio, and Sormano 2019a)	Plan related	Unspecified	Many	Temporal
(Torta, Micalizio, and Sormano 2019b)	Plan related	Centralized	Many	Temporal
(Qin et al. 2018)	Plan related	Distributed	Many	Temporal
(Micalizio, Torasso, and Torta 2004)	Coordination related	Centralized	Many	Temporal
(Daigle, Koutsoukos, and Biswas 2006)	Coordination related	Distributed	Many	Temporal
(Kalech and Kaminka 2005)	Coordination related	Centralized	One	Not temporal
(Kalech et al. 2006)	Coordination related	Distributed	One	Not temporal
(Kalech and Kaminka 2007)	Coordination related	Both	One	Not temporal
(Kalech and Kaminka 2011)	Coordination related	Both	One	Not temporal
(Kalech 2012)	Coordination related	Centralized	Many	Not temporal
(Passos, Abreu, and Rossetti 2015)	Coordination related	Centralized	Many	Temporal
(Elimelech et al. 2017)	Plan/Coordination related	Centralized	One	Temporal
(Natan and Kalech 2020)	Plan/Coordination related	Distributed	One	Temporal

Table 1: This table summarizes the related work considering the discussed aspects.

The following year, Kalech and Kaminka (Kalech and Kaminka 2007) introduced a novel design space of coordination-diagnosis algorithms. Their underlying assumption is that different faults might lead agents to disagree. They use the term "social diagnosis" to describe the process that diagnoses the reason why agents disagree. This process is divided into the task of selecting the diagnosing agent and the task of computing the diagnosis. Different methods were implemented for each task, and thousands of diagnosis cases were tested. They concluded that (a) centralizing the diagnosis calculation task is critical in reducing communications, and (b) techniques where agents do not explicitly reason about the beliefs of their peers are preferable in terms of computational runtime.

In (Kalech and Kaminka 2011), Kalech and Kaminka have extended their work to scale well with a high number of agents. The social diagnosis scalability can be achieved in two ways: (a) agents should use communication early in the hypotheses generation process to stave off unneeded reasoning, which ultimately leads to unneeded communication, and (b) by diagnosing only a limited number of representative agents (instead of all the agents).

In a following work (Kalech 2012), Kalech has proposed a matrix-based representation for the coordination between the agents and a set of operators to handle the exploration of the coordination along time. These operators use to model the MAS coordination and to diagnose the faulty agents when some fail.

A different approach, that integrates diagnosis of MAP and coordination faults, is presented by Elimelech et al. (Elimelech et al. 2017). In their work, the authors propose a model based approach to diagnose resource usage failures in multi-agent systems. On the one hand, each agents has a predefined plan, on the other hand, the plans dictate the agents' resource usage, which defines coordination constraints between the agents. They model the diagnosis problem, called TMARA-Diag, as a Model-Based Diagnosis problem by defining a set of constraints over the usage of the resources. They use a SAT solver to assign health values to different agents in order to explain the observation of the system. In later work, Natan and Kalch (Natan and Kalech 2020) extended this framework to distributed diagnosis, where the agents collaborate to compute the diagnosis without sharing their plans. They present synchronous and asynchronous distributed algorithms to diagnose the faulty agents.

Passos et al. (Passos, Abreu, and Rossetti 2015) present an original way to reduce the complexity of the diagnosis process. Instead of modeling the behaviours of the agents, they propose a model-free approach, usually used in automate debugging, Spectrum-based Fault Localisation (Abreu, Zoeteweij, and van Gemund 2011; Elmishali, Stern, and Kalech 2018). In this approach, the model is built on the fly, by tracking the MAS tasks and marking for each agent whether it participated an observed task or not. The success of the tasks is analysed too. This information is used then to generate possible diagnoses. This approach scales well but assumes full and many observations along the time. Also, they assume that the agents' and MAS' normal behavior are deduced from earlier executions in which no faults occurred.

Summary

Table 1 summarizes the different MAS diagnosis work detailed above. We grouped these works according to the MAS diagnosis objectives, either to diagnose plan-related faults (Section) or coordination-related faults (Section). In each objective, we further divide the works according to the following criteria:

- **Input.** The input given to the diagnoser. This can be a

global MAP that is shared between the agents, the agents' behaviors, a group of interconnected sub components of the system, Bond graph between varying subsystems.

- **Distributed or centralized.** Whether the computation of the diagnosis is done in a distributed or in a centralized manner.
- **Number of observations.** The number of observations given to the diagnoser.
- **Temporal or non-temporal.** Whether the diagnosis process takes into consideration temporal constraints or not.

Research Opportunities for MAS Diagnosis

Next, we describe open research opportunities to the AI community in the field of MAS diagnosis.

Troubleshooting. Troubleshooting is the task of failure root cause analysis and repair. Troubleshooting starts by a diagnosis process which outputs a set of candidate diagnoses and their likelihood. Then a discrimination process collects additional observations to localize the root cause of the fault. Finally, a repair process should fix the faulty component. Although the diagnosis process for multi-agent has been researched in the literature, troubleshooting remains an open research question. Given a set of diagnoses and their likelihood, choose what information (such as observations) to obtain in order to enable finding the correct diagnosis and to assist the agents to reach their goals.

Privacy. Privacy is a growing concern while designing a multi-agent system (Such, Espinosa, and García-Fornes 2014; Katewa, Pasqualetti, and Gupta 2018). Distributing the diagnosis process might help with preserving agent privacy to some degree. Agents in multi-agent systems could be cooperative (Kalech et al. 2006), non-cooperative (Elimelech et al. 2017) or even adversary (Rehák, Pěchouček, and Tožička 2005). While cooperative agents might not be so concerned about privacy, by definition non-cooperative and adversarial agents are usually presumed to be interested in keeping as much of their information for themselves. When diagnosing a system fault, such goal may be achieved to some degree by using a distributed approach in which agents share only part of their internal data or some sort of processed data that does not disclose sensitive information.

Uncertain observation. Typically, model-based diagnosis algorithms assume the existence of a model of the system and observations. In real world the observations are not always certain due to inaccurate sensors, environmental noises and communication failures (Cazes and Kalech 2020, 2021). In such cases we should diagnose the system with uncertain observation. In multi-agent domains, uncertain observations are natural and thus a third challenge is to develop diagnosis algorithms to cope with uncertain observations.

Online replan. As part of the troubleshooting process, once the malfunctioned agent is isolated, a repair action should be preformed. In a multi-agent system this task is not always feasible since the agents could be physically distributed so the malfunctioned agent could be far away from the control center. In this case, we would like to enable the other agents to continue according to their plans and reach their goals. However, due to some malfunctioned agents, the other agents may be stuck. Thus, the fourth challenge is to develop online replan algorithms for the multi-agent with a minimum deviation from the original plans.

A machine-learning approach. All the methods mentioned in this survey focus on model-based diagnosis approach. The reason is that, to the best of our knowledge, no other diagnosis approaches have been proposed for MAS. Some papers propose machine-learning algorithms that use multi-agent approach to diagnose other systems (Koujok, Ragab, and Amazouz 2019), but no paper proposed machine-learning algorithms to diagnose multi-agent system. As the growing of machine-learning field in academia and industry, and specifically for diagnostic tasks (Huang, Zhang, and Li 2018; Hoang and Kang 2019), we believe that this approach could be applied to diagnose MAS too.

Diagnosing intermittent faults. Previously developed MAS diagnosis algorithms were based on the assumption that faulty agents behave abnormally over time. However, there are systems in which faulty agents do not persistently perform faulty actions. Furthermore, in some systems, agents may perform faulty actions, but the system does not fail. These are known as Intermittent Faults (Kalech, Stern, and Lazebnik 2021). Observing the MAS in a single execution for a short period of time with such intermittent faults may result in an incorrect diagnosis. As a result, the MAS should be observed over time in a variety of executions. This presents the challenge of identifying the faulty agents that can explain the failed MAS executions. To the best of our knowledge, diagnosing intermittent faults in MAS has not been addressed.

Conclusions

In this survey, we presented studies on model-based diagnosis techniques for multi-agent systems. We divided the algorithms considering three attributes: (1) the objective of the MAS diagnosis - either algorithms that consider diagnosis of multi-agent plan or algorithms that consider MAS coordination faults. (2) The diagnosis method - either algorithms that use a centralized agent who makes the diagnosis or algorithms that run in a distributed manner. (3) Temporal vs. non- temporal - either algorithms that consider temporal aspects of MAS failures or only a single time fault. In addition, we presented challenges in MAS diagnosis that could be new opportunities for future work for the AI community.

Acknowledgments

This research was funded by ISF grant No. 1716/17, and by the ministry of science grant No. 3-6078.

References

Abreu, R.; Zoeteweij, P.; and van Gemund, A. J. C. 2011. Simultaneous debugging of software faults. *Journal of Systems and Software*, 84(4): 573–586.

Barták, R.; Švancara, J.; and Krasičenko, I. 2020. MAPF Scenario: Software for Evaluating MAPF Plans on Real

Robots. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 13602–13603.

Barták, R.; Švancara, J.; Škopková, V.; Nohejl, D.; and Krasičenko, I. 2019. Multi-agent path finding on real robots. *AI Communications*, 32: 175–189.

Beer, M.; d'Inverno, M.; Luck, M.; Jennings, N.; Preist, C.; and Schroeder, M. 1999. Negotiation in multi-agent systems. *The Knowledge Engineering Review*, 14(3): 285–289.

Brusoni, V.; Console, L.; Terenziani, P.; and Dupré, D. T. 1998. A spectrum of definitions for temporal model-based diagnosis. *Artificial Intelligence*, 102(1): 39–79.

Bunte, A.; Stein, B.; and Niggemann, O. 2019. Model-based diagnosis for cyber-physical production systems based on machine learning and residual-based diagnosis models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 2727–2735.

Calvaresi, D.; Appoggetti, K.; Lustrissimini, L.; Marinoni, M.; Sernani, P.; Dragoni, A. F.; and Schumacher, M. 2018. Multi-Agent Systems' Negotiation Protocols for Cyber-Physical Systems: Results from a Systematic Literature Review. In *ICAART (1)*, 224–235.

Cazes, D.; and Kalech, M. 2020. Model-Based Diagnosis with Uncertain Observations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 2766–2773.

Cazes, D.; and Kalech, M. 2021. Model-based diagnosis with uncertain observations. *Int. J. Intell. Syst.*, 36(7): 3259–3292.

Daigle, M.; Koutsoukos, X.; and Biswas, G. 2006. Distributed diagnosis of coupled mobile robots. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 3787–3794. IEEE.

De Jonge, F.; Roos, N.; and Witteveen, C. 2009. Primary and secondary diagnosis of multi-agent plan execution. *Autonomous Agents and Multi-Agent Systems*, 18(2): 267–294.

de Kleer, J.; and Williams, B. C. 1987. Diagnosing Multiple faults. *Artificial Intelligence*, 32(1): 97–130.

Edwards, D. 2020. Amazon now has 200,000 robots working in its warehouses. *Robotics and Automation*, 21.

El Koujok, M.; Ragab, A.; Ghezzaz, H.; and Amazouz, M. 2020. A Multi-Agent-Based Methodology for Known and Novel Faults Diagnosis in Industrial Processes. *IEEE Transactions on Industrial Informatics*.

Elimelech, O.; Stern, R.; Kalech, M.; and Bar-Zeev, Y. 2017. Diagnosing resource usage failures in multi-agent systems. *Expert Systems with Applications*, 77: 44–56.

Elmishali, A.; Stern, R.; and Kalech, M. 2018. An Artificial Intelligence paradigm for troubleshooting software bugs. *Engineering Applications of Artificial Intelligence*, 69: 147–156.

Erdem, E.; Kisa, D. G.; Oztok, U.; and Schüller, P. 2013. A general formal framework for pathfinding problems with multiple agents. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*.

Franchi, E.; and Poggi, A. 2012. Multi-agent systems and social networks. In *Handbook of Research on Business Social Networking: Organizational, Managerial, and Technological Dimensions*, 84–97. IGI Global.

Geihs, K. 2020. Engineering challenges ahead for robot teamwork in dynamic environments. *Applied Sciences*, 10(4): 1368.

Hausman, W. H.; Schwarz, L. B.; and Graves, S. C. 1976. Optimal storage assignment in automatic warehousing systems. *Management science*, 22(6): 629–638.

Hoang, D.-T.; and Kang, H.-J. 2019. A survey on deep learning based bearing fault diagnosis. *Neurocomputing*, 335: 327–335.

Huang, Q.; Zhang, F.; and Li, X. 2018. Machine learning in ultrasound computer-aided diagnostic systems: a survey. *BioMed research international*, 2018.

Kalech, M. 2012. Diagnosis of coordination failures: a matrix-based approach. *Autonomous Agents and Multi-Agent Systems*, 24(1): 69–103.

Kalech, M.; and Kaminka, G. A. 2005. Towards model-based diagnosis of coordination failures. In *AAAI*, volume 5, 102–107.

Kalech, M.; and Kaminka, G. A. 2007. On the design of coordination diagnosis algorithms for teams of situated agents. *Artificial Intelligence*, 171(8-9): 491–513.

Kalech, M.; and Kaminka, G. A. 2011. Coordination diagnostic algorithms for teams of situated agents: Scaling up. *Computational Intelligence*, 27(3): 393–421.

Kalech, M.; Kaminka, G. A.; Meisels, A.; and Elmaliach, Y. 2006. Diagnosis of multi-robot coordination failures using distributed CSP algorithms. In *AAAI*, 970–975.

Kalech, M.; Stern, R.; and Lazebnik, E. 2021. Minimal Cardinality Diagnosis in Problems with Multiple Observations. *Diagnostics*, 11(5).

Katewa, V.; Pasqualetti, F.; and Gupta, V. 2018. On privacy vs. cooperation in multi-agent systems. *International Journal of Control*, 91(7): 1693–1707.

Kato, S.; Takeuchi, E.; Ishiguro, Y.; Ninomiya, Y.; Takeda, K.; and Hamada, T. 2015. An open approach to autonomous vehicles. *IEEE Micro*, 35(6): 60–68.

Khalastchi, E.; and Kalech, M. 2019. Fault detection and diagnosis in multi-robot systems: a survey. *Sensors*, 19(18): 4019.

Koujok, M. E.; Ragab, A.; and Amazouz, M. 2019. A Multi-Agent Approach Based on Machine-Learning for Fault Diagnosis. *IFAC-PapersOnLine*, 52(10): 103–108. 13th IFAC Workshop on Intelligent Manufacturing Systems IMS 2019.

Li, J.; Ruml, W.; and Koenig, S. 2021. EECBS: A Bounded-Suboptimal Search for Multi-Agent Path Finding. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

Lo, C.; Lynch, J. P.; and Liu, M. 2016. Distributed model-based nonlinear sensor fault diagnosis in wireless sensor networks. *Mechanical Systems and Signal Processing*, 66: 470–484.

Ma, M.; and Li, Z. 2021. A time-independent trajectory optimization approach for connected and autonomous vehicles under reservation-based intersection control. *Transportation Research Interdisciplinary Perspectives*, 9: 100312.

Marín-Lora, C.; Chover, M.; Sotoca, J. M.; and García, L. A. 2020. A game engine to make games as multi-agent systems. *Advances in Engineering Software*, 140: 102732.

Metodi, A.; Stern, R.; Kalech, M.; and Codish, M. 2014. A novel sat-based approach to model based diagnosis. *Journal of Artificial Intelligence Research*, 51: 377–411.

Micalizio, R. 2009. A distributed control loop for autonomous recovery in a multi-agent plan. In *Twenty-First International Joint Conference on Artificial Intelligence*.

Micalizio, R.; and Torasso, P. 2007. Plan diagnosis and agent diagnosis in multi-agent systems. In *Congress of the Italian Association for Artificial Intelligence*, 434–446. Springer.

Micalizio, R.; and Torasso, P. 2014. Cooperative monitoring to diagnose multiagent plans. *Journal of Artificial Intelligence Research*, 51: 1–70.

Micalizio, R.; Torasso, P.; and Torta, G. 2004. On-line monitoring and diagnosis of multi-agent systems: a model based approach. In *Proceedings of the 16th European Conference on Artificial Intelligence*, 848–852. IOS Press.

Micalizio, R.; and Torta, G. 2012. Diagnosing delays in multi-agent plans execution. In *European Conference on Artificial Intelligence*, 594–599. IOS Press.

Natan, A.; and Kalech, M. 2020. Distributed Diagnosis of Multi-Agent Plans. In *31st International Workshop on Principle of Diagnosis (DX-20)*.

Pallottino, L.; Scordio, V. G.; Bicchi, A.; and Frazzoli, E. 2007. Decentralized cooperative policy for conflict resolution in multivehicle systems. *IEEE Transactions on Robotics*, 23(6): 1170–1183.

Parker, L. E. 2012. Reliability and fault tolerance in collective robot systems. *Handbook on Collective Robotics: Fundamentals and Challenges*.

Passos, L. S.; Abreu, R.; and Rossetti, R. J. 2015. Spectrum-based fault localisation for multi-agent systems. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Pynadath, D. V.; and Tambe, M. 2003. An automated teamwork infrastructure for heterogeneous software agents and humans. *Autonomous Agents and Multi-Agent Systems*, 7(1): 71–100.

Qin, L.; He, X.; Zhou, D.; Cui, J.; Li, F.; and Wang, G. 2018. A new local-model-based distributed fault diagnosis scheme for multi-agent systems with actuator faults. *IFAC-PapersOnLine*, 51(24): 292–299.

Rehák, M.; Pěchouček, M.; and Tožička, J. 2005. Adversarial behavior in multi-agent systems. In *International Central and Eastern European Conference on Multi-Agent Systems*, 470–479. Springer.

Reiter, R. 1987. A Theory of Diagnosis from First Principles. *Artificial Intelligence*, 32(1): 57–96.

Roos, N.; Ten Teije, A.; and Witteveen, C. 2003. A protocol for multi-agent diagnosis with spatially distributed knowledge. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 655–661. ACM.

Roos, N.; and Witteveen, C. 2009. Models and methods for plan diagnosis. *Autonomous Agents and Multi-Agent Systems*, 19(1): 30–52.

Runhua, Q.; Hua, C.; Ruiling, Z.; and Yuanxing, L. 2013. Design scheme of public transport comprehensive dispatching MIS based on MAS. *Procedia-Social and Behavioral Sciences*, 96: 1063–1068.

Srivastava, I.; Bhat, S.; and Singh, A. R. 2020. Fault diagnosis, service restoration, and data loss mitigation through multi-agent system in a smart power distribution grid. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, 1–26.

Su, R.; Wonham, W.; Kurien, J.; and Koutsoukos, X. 2002. Distributed diagnosis for qualitative systems. In *Sixth International Workshop on Discrete Event Systems*, 169–174. IEEE.

Such, J. M.; Espinosa, A.; and García-Fornes, A. 2014. A survey of privacy in multi-agent systems. *Knowledge Engineering Review*, 29(3): 314–344.

Surynek, P.; Felner, A.; Stern, R.; and Boyarski, E. 2016. Efficient SAT approach to multi-agent path finding under the sum of costs objective. In *Proceedings of the Twenty-second European Conference on Artificial Intelligence*, 810–818.

Švancara, J.; Vlk, M.; Stern, R.; Atzmon, D.; and Barták, R. 2019. Online multi-agent pathfinding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 7732–7739.

Sycara, K.; and Sukthankar, G. 2006. Literature review of teamwork models. *Robotics Institute, Carnegie Mellon University*, 31: 31.

Tambe, M. 1997. Towards flexible teamwork. *Journal of artificial intelligence research*, 7: 83–124.

Torta, G.; and Micalizio, R. 2018. SMT-Based Diagnosis of Multi-Agent Temporal Plans. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2097–2099. International Foundation for Autonomous Agents and Multiagent Systems.

Torta, G.; Micalizio, R.; and Sormano, S. 2019a. Explaining Failures Propagations in the Execution of Multi-Agent Temporal Plans. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2232–2234. International Foundation for Autonomous Agents and Multiagent Systems.

Torta, G.; Micalizio, R.; and Sormano, S. 2019b. Temporal Multiagent Plan Execution: Explaining What Happened. In *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, 167–185. Springer.