Diagnosis of Simple Temporal Networks

Nico Roos [1] and **Cees Witteveen** [2]

Abstract. In many domains successful execution of plans requires careful monitoring and repair. Diagnosis of plan execution supports this process by identifying causes of plan failure.

Most plans have to satisfy temporal constraints. An important and common occurring problem during plan execution are violations of temporal plan constraints. This paper addresses diagnosis of such temporal constraint violations by modeling the temporal aspects of a plan as a Simple Temporal Network (STN). We investigate the computational properties of standard diagnostic concepts but we also argue that traditional notions of preferred diagnoses such as minimum diagnosis are not adequate. A new notion of a *maximum confirmation diagnosis* is introduced.

1 Introduction

A Simple Temporal Network (STN) [8] provides a way to describe (*i*) a *plan*, (*ii*) temporal aspects of plan steps, and (*iii*) temporal relations between plan steps. It also enables the description of *schedule constraints*, and of *observations* about the temporal execution of the plan, using the same formalism. The observations may violate the temporal constraints of the plan or its schedule, giving rise to a *Simple Temporal Diagnosis* (STD) problem. Diagnosis should identify the plan and scheduling constraints that have been violated during plan-execution.

A Simple Temporal Diagnosis problem is related to a Simple Temporal Problem (STP) [8]. An STP addressed the identification of an allowable schedule for an STN. The STD problem extends this by identifying where the actual execution schedule starts to deviate from the allowable schedule. Note that STD may also be used prior to plan execution if an STP does not have an allowable schedule.

2 Running example

To illustrate the ideas presented in the following sections, we will use a problem from the domain of Air Traffic Control as a running example.

Flight KL 123 has a delayed departure; a delayed takeoff at 16:30 instead of the scheduled takeoff at 15:55-16:00. The taxiing time of 15-20 minutes incurred no delays. In fact flight KL 123 had a delayed off-block time.

At the gate flight KL 123 incurred a delay because of the catering. Catering was scheduled to start the delivery of food between 15:15 and 15:30, which must arrive at the airplane 10 to 30 minutes before the off-block. The actual delivery time was 16:00.

Flight KL 123 was also delayed because the flight had to wait on transfer passengers from flight NW 456. At least 30 minutes are required for the transfer of passengers between flights. Flight NW 456 arrived 16:05 at the gate while it was scheduled to arrive at the gate between 14:55 and 15:00. The cause of its delay was a delayed departure of 15 minutes at JFK and an additional delay during the flight caused by unexpected head-winds.

Figure 1 show the schedule of the plan and the actual execution of the plan. The figure shows the time lines for the flights KL123 and NW 456, and the time line for the catering. The blocks drawn on the time line represent the plan steps. Note that the length of the blocks roughly indicate the duration of the plan steps. The uncertainty about the start or finish of a plan steps is indicated by the time intervals below each time line. Also note the white blocks that are placed above instead of on the time line. These blocks indicate (*i*) the 'waiting time' between the on-block time of flight NW 456 and the off-block time of flight KL 123 in which passengers are transferred between the two flights, and (*ii*) the 'waiting time' between the finish of the catering service and the off-block time of flight KL 123.

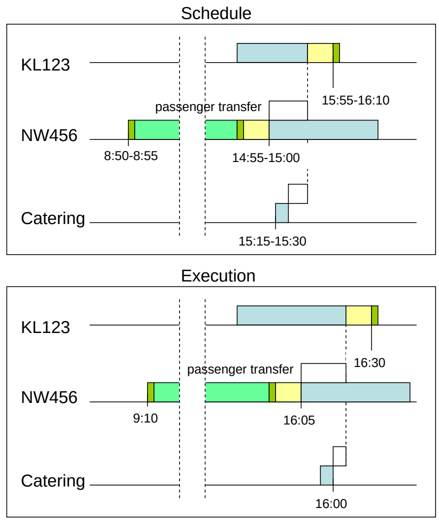

Figure 1. The schedule and the execution of two flights and the catering.

The goal of diagnosis is to determine to what extent the *plan constraints* describing plan step durations, time restriction on successive plan steps, and the scheduled, are satisfied using partial observations of the plan execution.

3 Preliminaries

Simple Temporal Networks A STN $(\mathcal{E}, \mathcal{C})$ describes a plan and its schedule by a set of events $\mathcal{E}$ and a set of constraints $\mathcal{C}$ over the events. Events denote such things as the start $start(s)$ of a plan step

[1] Maastricht ICT Competence Center, Universiteit Maastricht
P.O.Box 616, NL-6200 MD Maastricht, email: roos@micc.unimaas.nl
[2] Faculty EEMCS, Delft University of Technology
P.O.Box 5031, NL-2600 GA Delft, email: C.Witteveen@tudelft.nl

s and the finish $finish(s)$ of s. The constraints are used to specify the durations of plan steps, the precedence relations between plan steps, and the plan's schedule. It is also possible to specify requirements such as the requirement that a plan step that must start within δ minutes after the finish of its preceding plan step.

To describe a constraint, we associate a variable t_e with each event $e \in \mathcal{E}$. These variables take values in some dense time domain $Time$. We assume $Time$ to be a total order with element 0 and maximum element ∞. A constraint $c \in \mathcal{C}$ specifies the allowed temporal difference between two events: $lb \leq t_e - t_{e'} \leq ub$ where e and e' are events in $\mathcal{E}$, $lb, ub \in Time$ and $0 \leq lb \leq ub$.

Constraints define a strict precedence relation $\prec$ on the $\mathcal{E}$. We say that e' directly precedes e iff $lb \leq t_e - t_{e'} \leq ub \in \mathcal{C}$ and $ub > 0$. The transitive closure of the direct precedences defines the precedence relations; i.e., e' precedes e iff $e' \prec^+ e$.

Relating an STN to a traditional plan description $P = (S, \prec)$, the *duration* of a plan step s is described by $0 < lb \leq t_{finish(s)} - t_{start(s)} \leq ub$. A *precedence constraint* $s \prec s'$ is described by $lb \leq t_{start(s')} - t_{finish(s)} \leq ub$. Note that in the standard interpretation of a precedence constraint, $lb = 0$ and $ub = \infty$.

A *schedule* is a placement of events on the timeline. To describe a schedule we need a special event '0' marking start of the timeline; i.e., $t_0 = 0$. This enables us to schedule the period in which an event $e \in \mathcal{E}$ should take place: $lb \leq t_e - t_0 \leq ub$; i.e.: $lb \leq t_e \leq ub$.

Figure 2 and 3 shows the plan of our running example and the corresponding STN, respectively. Since there are no gaps between plan steps such as 'flight', 'landing', 'taxiing', and so on; i.e., precedence constraints of the form $0 \leq t_{start(s')} - t_{finish(s)} \leq 0$ hold between successive plan steps, and since these constraint cannot be violated, in Figure 3 we have chosen to represent the finish and start of successive plan steps of a flight by a single event.

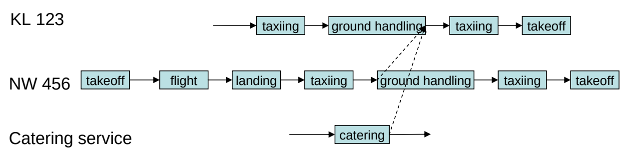

Figure 2. The plan.

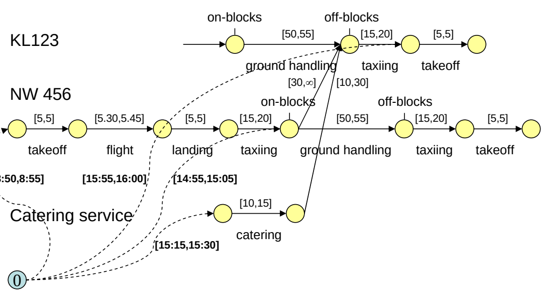

Figure 3. The Simple Temporal Network.

Semantics The constraints of an STN place restrictions on the way a plan may be executed; the *execution schedule*. An execution schedule for the set of events $\mathcal{E}$ of an STN $(\mathcal{E}, \mathcal{C})$ is a function $\sigma : \mathcal{E} \to Time$. We say that an execution schedule σ *satisfies* the constraints $\mathcal{C}$, denoted by $\sigma \models \mathcal{C}$, iff $lb \leq \sigma(e) - \sigma(e') \leq ub$ holds for every constraint $lb \leq t_e - t_{e'} \leq ub \in \mathcal{C}$. An execution schedule satisfying every constraints in $\mathcal{C}$ is called an *allowable schedule*. The identification of a allowable schedule for an STN is called a Simple Temporal Problem (STP) [8]. It is well-known that an STN has an allowable execution schedule iff its underlying labeled graph contains no negative cycles. [3]

We say that a constraint $c : a \leq t_e - t_{e'} \leq b$ is entailed by a set of constraints C, denoted by $C \models c$, iff every allowable schedule for C satisfies c.

Given a constraint $c : a \leq t_e - t_{e'} \leq b$ we say that $c' : a' \leq t_e - t_{e'} \leq b'$ is a tightening of c, denoted by $c' \models c$ iff $a \leq a' \leq b' \leq b$.

There is a sound and complete *derivation procedure* ($\vdash$) for determining the most tightened constraint $c : a \leq t_e - t_{e'} \leq b$ entailed by a set of constraints C: $C \vdash c$ iff $C \models c$.

Observations During the execution of a plan *observations* can be made. These observations may pertain to the time difference observed between two events e and e' as specified in the plan or may pertain to the time at which a certain event $e \in \mathcal{E}$ takes place.

We assume that the first type of observation is described by some constraint $a \leq t_e - t_{e'} \leq b$ indicating that we have observed that event e occurred at least a time steps, but within b time steps after e'.

The second type of observation is given by a constraint $a \leq t_e - t_0 \leq b$ indicating that e occurred after a time units but before b time units have been passed (after the occurrence of the time reference event '0'). The set of observations containing these constraints is denoted by Obs.

In the running example, we have the following observations. The delayed takeoff of flight KL 123 at 16:30, the catering starting at 16:00, and the delayed arrival at the gate of flight NW 456 at 16:05. These observations are described by the constraints 16:30 $\leq t_e - t_0 \leq$ 16:30, 16:00 $\leq t_{e'} - t_0 \leq$ 16:00, and 16:05 $\leq t_{e'} - t_0 \leq$ 16:05. respectively.

Compatibility An important notion is the compatibility between the STN specification $(\mathcal{E}, \mathcal{C})$ and the set of observations Obs.

We say that the set of observations is compatible with the plan specification if we can find an execution schedule σ that satisfies the original set of constraints $\mathcal{C}$ as well as the set Obs; i.e., the STN $(\mathcal{E}, \mathcal{C} \cup Obs)$ has an allowable schedule.

Qualifications If an STN $(\mathcal{E}, \mathcal{C})$ is not compatible with a set Obs of observations, some constraints in $\mathcal{C}$ must have been violated directly or indirectly by some of the observations. To restore the compatibility between plan and observations we need to indicate which constraints have been violated. Clearly, if a plan constraint c is violated, some part of it is executed in an abnormal way. To indicate such an abnormal execution we introduce a *qualification* Q of constraints. Given an STN $(\mathcal{E}, \mathcal{C})$, a qualification Q is a function $Q : \mathcal{C} \to H$ assigning a health mode to every constraint in $\mathcal{C}$. We distinguish the following health modes:

1. We use the mode $Q(c) = nor$ to denote the *normal* execution of a constraint $c \in \mathcal{C}$; i.e., c has not been violated,
2. we use $Q(c) = ab$ to denote the *abnormal* execution of a constraint c without exactly specifying how it is violated, and
3. we use a real number $\delta \in \mathbb{R}$ to denote the degree in which a constraint is violated: $Q(c) = \delta$.

Note that the last health mode describes how much shorter or longer the temporal difference between two events is with respect to what is specified by the constraint.

[3] A negative cycle refers to the fact that, first of all, a constraint $c : lb \leq t_e - t_{e'} \leq ub$ is equivalent to the following two inequalities: $t_{e'} - t_e \leq -lb$ and $t_e - t_{e'} \leq ub$. Next, such inequalities can be composed: if $t_{e'} - t_{e''} \leq ub'$ then $t_e - t_{e''} = t_e - t_{e'} + t_{e'} - t_{e''} \leq ub + ub'$. If we can derive an inequality $t_e - t_e < 0$, a clear inconsistency (a negative cycle) has been detected [8].

Qualifications will be used to restore the compatibility between observations and plan executions as follows:

1. For any constraint $c : lb \leq t_e - t_{e'} \leq ub$, if $Q(c) = ab$, we assume that the constraint is not respected anymore. So c will be replaced by its weakest implicate $-\infty \leq t_e - t_{e'} \leq \infty$, which in fact comes down to removing c from $\mathcal{C}$.
2. If $Q(c) = \delta \in \mathbb{R}$ then $c : lb \leq t_e - t_{e'} \leq ub$ will be replaced by the constraint $lb + \delta \leq t_e - t_{e'} \leq ub + \delta$. Since the duration of plan steps and waiting times between successive plan steps cannot be negative, we require that $Q(c) \geq -1 \cdot lb$.

We will use the *update* fuction $upd(\mathcal{C}, Q)$ to denote modification of the constraints $\mathcal{C}$ using qualification Q.

$$\begin{aligned} upd(\mathcal{C}, Q) = \; & \{c \in \mathcal{C} \mid Q(c) = nor\} \cup \\ & \{lb + \delta \leq t_e - t_{e'} \leq ub + \delta \mid \\ & \quad c : lb \leq t_e - t_{e'} \leq ub \in \mathcal{C}, Q(c) = \delta\} \end{aligned}$$

Note that the qualification of the health mode 'ab' to a constraint increases the uncertainty expressed by the constraint; i.e., the difference between the upper and lower bound of the constraint. The qualification of the health mode $\delta \in \mathbb{R}$ does not change the expressed uncertainty since $(ub + \delta) - (lb + \delta) = ub - lb$.

4 Diagnosis

Classical Model-Based Diagnosis (MBD) addresses the identification of failing components is some system. In MBD, two types of diagnosis are distinguished, abductive and consistency based diagnosis. The abductive diagnosis can be viewed as a special case of consistency based diagnosis where we have complete knowledge of both the way components may fail and how failing components behave.[4]

Since in abstraction, diagnosis of constraint violations in an STN is closest related to MBD, we will use the terminology used in classical MBD. Note however, that unlike MBD, we do not have components to be diagnosed. Instead we diagnose *temporal* constraints.

We distinguish two types of diagnosis: diagnosis without fault models where only the health modes nor and ab are used, and diagnosis with fault models.

4.1 Diagnosis without fault models

We consider consistency based diagnosis *without fault models*. That is, we try to make the STN compatible with the observations by identifying the constraints that could have been violated without considering how the constraints are violated. We therefore restrict ourselves to qualifications Q that map constraints to nor or ab. As we remarked before, constraints qualified as being *abnormal* will be removed from the set of constraints $\mathcal{C}$ defined by the STN $(\mathcal{E}, \mathcal{C})$.

Definition 1 *Let $(\mathcal{E}, \mathcal{C})$ be an STN and let Obs be the constraints describing the observations made. Moreover, let $Q : \mathcal{C} \rightarrow H$ be a qualification such that for every $c \in \mathcal{C}$, $Q(c) \in \{nor, ab\}$.*

The qualification Q is a consistency based diagnosis without fault models *iff the STN $(\mathcal{E}, \{c \in \mathcal{C} \mid Q(c) = nor\} \cup Obs)$ has an allowable schedule.*

[4] Diagnosis of Discrete Event Systems (DESs) is another form of model-based diagnosis that addresses the identification of failure events that change the states of components in dynamic systems.

In general, there may not be a unique diagnosis given the observations made. In fact the number of diagnoses can be quite large. For instance, in the absence of fault models, if Q is a diagnosis, then every Q' such that

$$\{c \in \mathcal{C} \mid Q(c) \neq nor\} \subseteq \{c \in \mathcal{C} \mid Q'(c) \neq nor\}$$

is also diagnosis.

Dependencies between constraint violations Among the set of diagnoses, some diagnoses are considered to be better than others. To select the most likely diagnosis, in MBD, preference orders are defined on the set of diagnoses. These preference orders are all based on the underlying assumption that fault probabilities are independent of each other. This assumptions does not hold for the constraints of an STN. In particular the schedule constraints are not independent of other schedule constraints, plan duration constraints and precedence constraints. For instance, a delay in boarding of passengers may imply a violation of the scheduled takeoff time.

To illustrate the problem of dependencies more clearly, consider the plan depicted in Figure 4. Suppose that we make the observations $11{:}00 \leq t_5 - t_0 \leq 11{:}05$ and $10{:}55 \leq t_6 - t_0 \leq 11{:}00$. Clearly, the schedule constraints $c_{0\text{-}5} : 10{:}40 \leq t_5 - t_0 \leq 10{:}50$ and $c_{0\text{-}6} : 10{:}45 \leq t_6 - t_0 \leq 10{:}50$ are violated and a *minimum* diagnosis qualifies these constraints as abnormal (ab) while qualifying all other constraints as normal (nor).

A diagnosis in which the schedule constraints $c_{0\text{-}5}$, $c_{0\text{-}6}$ and the plan constraint $c_{1\text{-}2} : 14 \leq t_2 - t_1 \leq 23$ are qualified as abnormal (ab) is *not* a minimum diagnosis. Since a violation of the plan constraint $c_{1\text{-}2}$; e.g., its execution is taking at least 15 minutes longer, *implies* the violations of the schedule constraints $c_{0\text{-}5}$ and $c_{0\text{-}6}$, we should only count $c_{1\text{-}2}$ when determining a minimum diagnosis. The violations of $c_{0\text{-}5}$ and $c_{0\text{-}6}$ are not independent of the violation of $c_{1\text{-}2}$.

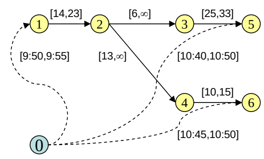

Figure 4. Dependencies between constraints.

The above example shows us that notions, such as minimum diagnoses, cannot be defined considering all the violated constraints. Instead, we should consider an "independent core" of a diagnosis Q.

To identify the independent core, we first define a causal dependency between a constraint c and a set of constraints D. The idea is that the upper and lower bound of c cannot be chosen independently of the constraints in D. Moreover, for the constraint $c : lb \leq t_e - t_{e'} \leq ub$ to *causally* depend on D, no event of a constraint c' in D may occur after the event e.

Definition 2 *A constraint $c : lb \leq t_e - t_{e'} \leq ub \in \mathcal{C}$ depends on a set of constraints D not containing c iff*

- *D is a minimal subset of $\mathcal{C}$ such that for some choices for lb and ub, $D \cup \{c\}$ has no allowable schedule,*
- *for no event e'' specified in a constraint in D, e precedes e''.*

The independent core of a diagnosis Q can now be determined by identifying the constraints in Q that (*i*) are qualified the health mode ab and (*ii*) do not depend on other constraints that are qualified the health mode ab in Q.

Definition 3 *Let Q be a diagnosis and let $C \subseteq \mathcal{C}$ be a set of constraints.*

C is an independent core *of Q iff C contains of all constraints $c \in \mathcal{C}$ such that $Q(c) = ab$, and for all sets of constraints $D \subseteq \mathcal{C}$ on which c depends, there is no $c' \in D$ such that $Q(c') = ab$.*

Minimum diagnoses In MBD, one usually prefers *minimum* diagnoses. The rational behind this preference is that the probability of n faults is usually much smaller than the probability of m faults for $n > m$ provided that fault probabilities are independent. In an STN the independence requirement does not hold. Therefore, minimum diagnoses must be defined with respect to the *independent core* of a diagnosis Q.

In the running example, we observe a delayed takeoff of flight KL 123, a delayed on-block of flight NW 456 and a delay in the finish of the catering service. One possible diagnosis Q qualifies as abnormal (ab) the scheduled takeoff time of flight KL 123, the flying time of flight NW 456 and its scheduled on-block time, and the scheduled starting time of the catering service. All other constraints are qualified as normal (nor). The independent causal core of Q are the constraints specifying the flying time of flight NW 456 and the scheduled starting time of the catering service. Since the number of constraints in the independent causal core is minimal, Q is a minimum diagnosis.

Theorem 1 *Finding a diagnosis with a* minimum independent core *for an STD problem is an NP-hard problem.*

We prove NP-hardness by reducing the well-known NP-complete Feedback Arc Set problem [9] to the problem of finding a diagnosis with a minimum independent core.

Consider an instance $I = (G(V, A), K)$ of the Feedback Arc Set problem. We construct an instance $f(I) = (P(\mathcal{E}, \mathcal{C}), Obs)$ of the temporal diagnosis problem by specifying the plan $P(\mathcal{E}, \mathcal{C})$ as follows:

- For every node $v \in V$ we create two events e^1_v and e^2_v in $\mathcal{E}$;
- For every arc $(v, w) \in A$ we add a temporal constraint $1 \le t_{e^2_w} - t_{e^1_v} \le \infty$ to the temporal network. Note that the source of an arc (v, w) in the graph $G(V, A)$ is always a e^1_v-event in the temporal network while the target is always a e^2_w-event; see Figure 5.

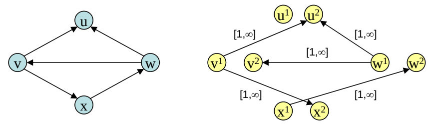

Figure 5. Reduction of a Feedback Arc Set problem to STD.

It is easy to see that this plan has an allowable execution schedule: for every event $t^i_v \in \mathcal{E}$, let $\sigma(t^i_v) = i$. This assignment satisfies all constraints. Moreover, since there is at most one path of constraints between each pair of events in the STN, the independent core consists of all constraints that are qualified as being abnormal in diagnosis.

The set of observations Obs of observations restores the structure of the graph G by containing for every node $v \in V$ the constraint $0 \le t_{e^1_v} - t_{e^2_v} \le \infty$. It is not hard to see that the observations are incompatible with the STN (i.e., the STN contains a negative cycle) iff the graph G contains a cycle. Moreover, a diagnosis in which K constraints are qualified as abnormal ab corresponds one to one with a directed feedback arc set of size K of the graph $G(V, A)$.

4.2 Diagnosis with fault models

An important difference with diagnosis in other domains is that in diagnosis of STNs fault models are always available. In an STN, a fault model of a temporal constraint describes the degree to which the constraint is violated. In the qualification Q we denote this by shift in the bound of the temporal constraints. So, if $Q(c) = \delta \in \mathbb{R}$, then $c : lb \le t_e - t_{e'} \le ub$ will be replaced by the constraint $lb + \delta \le t_e - t_{e'} \le ub + \delta$. Hence, diagnosis with fault models is defined as:

Definition 4 *Let $(\mathcal{E}, \mathcal{C})$ be an STN and let Obs be the constraints describing the observations made. Moreover, let $Q : \mathcal{C} \to H$ be a qualification.*

The qualification Q is a consistency based diagnosis with fault models *iff the STN $(\mathcal{E}, upd(\mathcal{C}, Q) \cup Obs)$ has an allowable schedule.*

Preferred diagnoses Definition 4 does not give us a unique diagnosis given the observations. Some diagnoses may be better than others. Generalizing the preference for minimum diagnoses in the absence of fault models, we could prefer *minimum-fault* diagnoses that minimize $\sum_{c \in \mathcal{C}} |Q(c)|$ where $Q(c) = nor$ and $Q(c) = ab$ are interpreted as $Q(c) = 0$ and $Q(c) = \omega$, respectively. Clearly, minimum diagnoses are a special case of minimum-fault diagnoses.

A minimum-fault diagnosis minimizes the number of execution schedules σ that satisfy an updated STN $(\mathcal{E}, upd(\mathcal{C}, Q))$ and the observations Obs. To give an illustration, consider a plan with two events e and e' and one constraint: $c : lb \le t_e - t_{e'} \le ub$. If we observe $a \le t_e - t_{e'} \le b$ with $a > ub$, then $Q(c) = a - lb$ is a minimum-fault diagnosis. Since there is only one execution schedule satisfying the updated plan and the observation, the probability that the diagnosis is correct is minimal. Therefore, a different notion of preference is desirable.

We should prefer diagnoses that have a high probability of being correct; i.e., maximize the number of execution schedules. The number of execution schedules is maximal if we can predict the observations made; i.e., *abductive diagnosis*.

To illustrate this point, consider the plan in figure 6 together with the observations: $10{:}40 \le t_5 - t_0 \le 11{:}15$ and $10{:}30 \le t_6 - t_0 \le \infty$. If all constraints are qualified normal (nor), then the constraints entail $10{:}35 \le t_5 - t_0 \le 11{:}03$ and $10{:}27 \le t_6 - t_0 \le \infty$, which do not explain our observations.

A diagnosis Q qualifying all plan constraint as normal (nor) except $c_{1-2} : 14 \le t_2 - t_1 \le 23$, which is qualified as: $Q(c) = 5$, does explain the observations. This diagnosis enables us to predict $10{:}40 \le t_5 - t_0 \le 11{:}08$ and $10{:}32 \le t_6 - t_0 \le \infty$. Since these predictions are a tightening of the observations, the diagnosis Q is an abductive diagnosis.

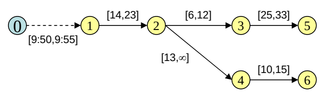

Figure 6. Abduction versus confirmation.

Maximum confirmation diagnosis In the above example the two observations are not very accurate. A more accurate observation such as $10{:}40 \le t_5 - t_0 \le 10{:}48$, cannot be explained by the normal execution of the plan: $Q(c) = nor$ for all constraints in $\mathcal{C}$. Nevertheless, the most tightened constraint $10{:}35 \le t_5 - t_0 \le 11{:}03$ entailed by the plan constraints, is *confirmed* by the observation. This also indicates the absence of violations of the constraints that are used to make the prediction for the pair of events '0' and e_5. Therefore, we propose a new notion of diagnosis, namely *maximum-confirmation diagnoses*. The idea of maximum-confirmation diagnosis is to identify the qualification Q for which the number of execution schedules is maximal. To measure the number of execution schedules, we introduce a confirmation percentage.

Definition 5 *Let Q be a qualification, let $o : lb \leq t_e - t_{e'} \leq ub$ be an observation and let $a \leq t_e - t_{e'} \leq b$ be the most tightened constraint implied by a qualified plan $(\mathcal{E}, upd(\mathcal{C}, Q))$.*

The confirmation percentage *of the observation o, denoted by $cp_Q(o)$, is defined as:*

$$cp_Q(o) = \begin{cases} \frac{\min(ub,b)-\max(lb,a)}{ub-lb} & \text{if } \min(ub,b) - \max(lb,a) \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

The sum of the confirmation percentages gives us a measure for comparing diagnoses.

Definition 6 *Let $(\mathcal{E}, \mathcal{C})$ be an STN, and let Obs be the constraints describing the observations made.*

A diagnosis Q of the STN and the observation Obs is a maximum-confirmation diagnosis *iff $\sum_{o \in Obs} cp_Q(o)$ is maximal.*

Note that a maximum-confirmation diagnosis need not be unique. From the set of maximum-confirmation diagnoses we can derive intervals of violation degrees for the constraints. In our running example, the maximum-confirmation diagnoses assign delays of 15 to 35 minutes to the catering process given the observed finish at 16:00.

An important question concerns the worst case time complexity of determining a maximum confirmation diagnosis.

Theorem 2 *A* maximum confirmation *diagnosis can be determined in polynomial time.*

To see why, note that each observation $o : lb \leq t_e - t_{e'} \leq ub \in Obs$ has one or more causal chains of events between the two events of the observation constraint e and e'. Starting from the earliest observation, we qualify the last plan constraint of each the causal chain of events between the two events e and e' as being violated. The qualification is chosen such that it maximizes the confirmation percentage of the observation o. Before continuing with the next constraint, we have to propagate the effect of the qualifications made. All steps can be carried out in polynomial time.

5 Related work

Several authors have addressed aspects of plan diagnosis.

- Diagnosis of an agent's planning assumptions: Birnbaum et al. [1].
- Diagnosis of a single task execution: Lesser et al. [2, 10].
- Social diagnosis of behavior selection in teams: Kalech and Kaminka [11, 13].
- Diagnosis of the abnormal effects of a plan execution: Roos et al. [19, 16, 7, 18].
- Diagnosis of coordination errors of agents executing a plan: Kalech and Kaminka [12] and Roos and Witteveen [17].
- Diagnosis of multi agent plan interactions: de Jonge et al. [3, 6].
- Diagnosis and repair of plan execution with agents share resources and provide services: Micalizio and Torasso [15, 14].
- Diagnosis of temporal constraint violations: de Jonge et al. [4, 5].

None of these approaches address diagnosis of Simple Temporal Networks. The approach to de Jonge et al. [4, 5] comes closest to our approach. However, they can only deal with abstract states such as *delayed* or *early* for plan steps.

6 Conclusion

Identifying causes of violations of a plan's temporal constraints is an important issue in plan execution. To enable such diagnosis, the temporal aspects of a plans are described by Simple Temporal Network (STN). Based on observations of the plan's execution, diagnosis has to identify the temporal constraint that are violated.

The notion of classical Model-Based Diagnosis (MBD) has been adapted to STNs. Two important issues had to be dealt with: (*i*) we cannot assume that temporal constraints are violated independently, and (*ii*) the notion of consistency-based and abductive diagnosis are not adequate for STNs. A new notion of a *maximum confirmation* diagnosis has been proposed.

In future work we will integrate whether the here presented model for diagnosis of STN can be combined with models for diagnosing other aspects of plan execution failures.

REFERENCES

[1] L. Birnbaum, G. Collins, M. Freed, and B. Krulwich. Model-based diagnosis of planning failures. In *AAAI 90*, pages 318–323, 1990.

[2] N. Carver and V. Lesser. Domain monotonicity and the performance of local solutions strategies for cdps-based distributed sensor interpretation and distributed diagnosis. *Autonomous Agents and Multi-Agent Systems*, 6(1):35–76, 2003.

[3] F. de Jonge and N. Roos. Plan-execution health repair in a multi-agent system. In *PlanSIG 2004*, 2004.

[4] F. de Jonge, N. Roos, and H. Aldewereld. Multiagent system technologies. In *Multiagent System Technologies*, 2007.

[5] F. de Jonge, N. Roos, and H. Aldewereld. Temporal diagnosis of multi-agent plan execution without an explicit representation of time. In *BNAIC-07*, 2007.

[6] F. de Jonge, N. Roos, and H. van den Herik. Keeping plan execution healthy. In *Multi-Agent Systems and Applications IV: CEEMAS 2005, LNCS 3690*, pages 377–387, 2005.

[7] F. de Jonge, N. Roos, and C. Witteveen. Diagnosis of multi-agent plan execution. In *Multiagent System Technologies: MATES 2006, LNCS 4196*, pages 86–97, 2006.

[8] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49:61–95, 1991.

[9] P. Festa, P. Pardalos, and M. Resende. Feedback set problems. In *Handbook of Combinatorial Optimization*, volume 4. Kluwer Academic Publishers, 1999.

[10] B. Horling, B. Benyo, and V. Lesser. Using Self-Diagnosis to Adapt Organizational Structures. In *Proceedings of the 5th International Conference on Autonomous Agents*, pages 529–536. ACM Press, 2001.

[11] M. Kalech and G. A. Kaminka. Diagnosing a team of agents: Scaling-up. In *AAMAS 2005*, pages 249–255, 2005.

[12] M. Kalech and G. A. Kaminka. Towards model-based diagnosis of coordination failures. In *AAAI 2005*, pages 102–107, 2005.

[13] M. Kalech and G. A. Kaminka. On the design of coordination diagnosis algorithms for theams of situated agents. *Artificial Intelligence*, 171:491–513, 2007.

[14] R. Micalizio and P. Torasso. On-line monitoring of plan execution: A distributed approach. *Knowledge-Based Systems*, 20:134–142.

[15] R. Micalizio and P. Torasso. Team cooperation for plan recovery in multi-agent systems. In *Multiagent System Technologies, LNCS 4687*, pages 170–181, 2007.

[16] N. Roos and C. Witteveen. Diagnosis of plans and agents. In *Multi-Agent Systems and Applications IV: CEEMAS 2005, LNCS 3690*, pages 357–366, 2005.

[17] N. Roos and C. Witteveen. Diagnosis of plan structure violations. In *Multiagent System Technologies*, 2007.

[18] N. Roos and C. Witteveen. Models and methods for plan diagnosis. *Journal of Autonomous Agents and Multi-Agent Systems*, DOI: 10.1007/s10458-007-9017-6, 2008.

[19] C. Witteveen, N. Roos, R. van der Krogt, and M. de Weerdt. Diagnosis of single and multi-agent plans. In *AAMAS 2005*, pages 805–812, 2005.