Solving Temporal Problems using SMT: Weak Controllability

A. Cimatti and **A. Micheli** and **M. Roveri**
Fondazione Bruno Kessler — Irst
{cimatti,amicheli,roveri}@fbk.eu

Abstract

Temporal problems with uncertainty are a well established formalism to model time constraints of a system interacting with an uncertain environment. Several works have addressed the definition and the solving of controllability problems, and three degrees of controllability have been proposed: weak, strong, and dynamic.

In this work we focus on weak controllability: we address both the decision and the strategy extraction problems. Extracting a strategy means finding a function from assignments to uncontrollable time points to assignments to controllable time points that fulfills all the temporal constraints.

We address the two problems in the satisfiability modulo theory framework. We provide a clean and complete formalization of the problems, and we propose novel techniques to extract strategies. We also provide experimental evidence of the scalability and efficiency of the proposed techniques.

Introduction

A temporal problem (TP) is a collection of temporal constraints over a given set of time points. A typical example is a set of activities with the respective durations subject to constraints. When durations are controllable, TP's range from simple temporal problems (STP), to Temporal constraint satisfaction problem (TCSP) (Dechter, Meiri, and Pearl 1991), to disjunctive temporal problems (DTP) (Tsamardinos and Pollack 2003), depending on the structure of the constraints. In these cases, a solution is an assignment to all the time points (to the starting and ending instants of the activities) that satisfies all the constraints.

When activities have uncertain (and uncontrollable) duration, we speak of TP with uncertainty (TPU), and previous problems are generalized to STPU, TCSPU, and DTPU (Vidal and Fargier 1999; Peintner, Venable, and Yorke-Smith 2007). A TPU admits several forms of solution. In the case of strong controllability, a solution is a precise, unconditioned assignment to each activity start, that will satisfy the constraints regardless of the uncontrollable duration of the activities. To draw an analogy, strong controllability is the TP counterpart for conformant planning, where the course of action is decided without observing the uncontrollable behavior of the environment.

Weak controllability, on the other hand, concerns the existence of a strategy that associates values to the starting points of each activity, as a function of the uncontrollable durations, that are assumed to be known in advance. Weak controllability is analogous to a form of conditional planning with uncertainty under full observability. At a first sight the "clairvoyance" needed to apply a weak strategy makes the weak controllability problem impractical; nevertheless, it has a clear theoretical importance in the field of temporal reasoning and it is also useful in situations where a parametric TP has to be solved. Given actual parameter values it is relatively easy to solve such a problem as it consists in solving consistency. However, if the same problem occurs many times with different parameter instantiations and the computation power of the device that needs to solve the TP is limited, it can be more effective to pre-compute solutions for any possible parameter instantiation. This problem can be modeled as a TPU where uncontrollable time points represent the parameter values. Finding a weak strategy for the TPU means finding a solution for any parameter allocation in advance.

In this paper, we tackle the problem of weak controllability, making two key contributions. First, we propose a general decision procedure for the problem of weak controllability for TPU's. Our approach is a cast in the framework of Satisfiability Modulo Theory (SMT) (Barrett et al. 2009), a formal framework that allows for the analysis of problems in decidable fragments of first order logic. The decision procedure is based on a reduction to an SMT problem for the theory of Quantified Linear Real Arithmetic (LRA). The encoding can be thought as working by refutation: we state the existence of an assignment to uncontrollable time points that cannot be countered by any controllable assignment. This means that the SMT problem is satisfiable if and only if the TP is not weakly controllable. The problem can be directly provided to an efficient SMT solver, thus this approach accounts for the first implemented decision procedure for weak controllability of DTPU.

Unfortunately, the approach is not constructive: in fact, when the problem is weakly controllable, the SMT solver will simply conclude the unsatisfiability of the SMT problem, providing no information on the strategy. The second contribution is then to investigate various constructive approaches to strategy extraction for the STPU problem class.

We first consider the case of linear strategies, that expresses the (controllable) starts of the activities as a linear function of the (uncontrollable) action durations. We show that strategy extraction can be reduced to an SMT problem for the theory of quantified nonlinear polynomials. The encoding is constructive in that the assignments to the existential variables are the coefficients for the linear function being sought. This approach is however impractical because of technical limitations of current SMT solvers, therefore we further simplify the approach by exploiting a generalization of the result in (Vidal and Fargier 1999).

We also show that a linear strategy is not always guaranteed to exist. Thus, we analyze the case of a piecewise-linear strategy, i.e. a conditional strategy that associates a linear function to each element of a finite partition of the uncontrollable space. We show that a piecewise-linear strategy is always guaranteed to exist for any STPU. From the constructive proof, we determine a procedure that induces the partition based on the enumeration of simplexes (hyper-tetrahedra). We further improve the method using a lazy approach, that tries to reduce the cardinality of the partition, by enlarging the scope of applicability of the linear sub-strategy whenever possible.

All the proposed algorithms have been implemented on top of state of the art SMT solvers, and have been evaluated on a comprehensive set of benchmarks.

Related work. The notion of weak controllability is proposed in the seminal paper by Fargier and Vidal (Vidal and Fargier 1999) for the STPU problem class and has been extended for TCSPU and DTPU in (Peintner, Venable, and Yorke-Smith 2007). In (Venable et al. 2010) the authors approach the problem of deciding weak controllability of DTPU using an explicit algorithm that enumerates the STPU components of a DTPU. In this work we focus on the same problem but we exploit symbolic techniques to avoid this explicit enumeration. In general, no algorithm implementations is available for deciding TPU weak controllability nor for extracting a strategy.

There exists a third form of controllability for TPU's, namely dynamic controllability, that is also analogous to conditional planning with uncertainty under full observability. The key difference is that in the case of weak controllability clairvoyance in the strategy is not allowed, i.e. the strategy cannot see in advance all the action durations, but only the ones that have already happened. Similarly to weak controllability, also dynamic controllability is a largely open problem (Venable et al. 2010). In the STPU problem class, a series of works (Morris, Muscettola, and Vidal 2001; Morris and Muscettola 2005) describe a technique to efficiently check dynamic controllability.

Structure of the paper. First, we provide some background and we define the problem. Second, we show the decision procedure. Third, we tackle the case of linear strategy extraction. Fourth, we discuss how to extract piecewise-linear strategies. Fifth, we present an experimental evaluation of the approach. Finally, we draw some conclusions and discuss future work.

Background

Satisfiability modulo theory

Given a first-order formula ψ in a background theory T the satisfiability modulo theory (SMT) problem consists in deciding whether there exists a *model* (i.e. an assignment to the free variables in ψ) that satisfies ψ. For example, consider the formula $(x \leq y) \wedge (x + 3 = z) \vee (z \geq y)$ in the theory of real numbers ($x, y, z \in \mathbb{R}$). The formula is satisfiable and a valid model is $\{x := 5,\ y := 6,\ z := 8\}$.

An SMT solver (Barrett et al. 2009) is a decision procedure which solves the satisfiability problem for a formula expressed in a decidable subset of first-order logic.

SMT solvers can support different *Theories*. A widely used theory is *Linear Real Arithmetic* (LRA). A formula in LRA is an arbitrary Boolean combination, or universal ($\forall$) and existential ($\exists$) quantification, of atoms in the form $\sum_i a_i x_i \bowtie c$ where $\bowtie \in \{>, <, \geq, \leq, \neq, =\}$, every x_i is a real variable and every a_i and c are real constants. We denote with QF_LRA the quantifier-free fragment. The logic in which we allow arbitrary polynomial atoms is called *Non-linear Real Arithmetic* (NRA), and its quantifier-free fragment is denoted with QF_NRA. To the best of our knowledge, currently there are no available SMT solvers for full NRA, while some SMT solvers like for instance *Microsoft Z3* (de Moura and Bjørner 2008) support QF_NRA.

Temporal problems with uncertainty

A temporal problem (TP) is a formalism that is used to represent temporal constraints over time-valued variables representing time points. This formalism is expressive enough to express Allen's interval algebra (Allen 1983) and also quantitative constraints over intervals and time points. Two families of TP's have been presented in literature over the years: TP without Uncertainty, in which all the time points are freely assignable (Dechter, Meiri, and Pearl 1991; Tsamardinos and Pollack 2003); and TP with Uncertainty (TPU) in which the represented situation is a game between an agent that tries to fulfill the constraints and an adversarial environment (Vidal and Fargier 1999). In this work we focus on TPU's.

Definition 1. *A TPU is a tuple* (X_c, X_u, C_c, C_f)*, where* $X_c \doteq \{b_1, ..., b_n\}$ *is the set of* controllable time points*,* $X_u \doteq \{e_1, ..., e_m\}$ *is the set of* uncontrollable time points*,* $C_c \doteq \{cc_1, ..., cc_m\}$ *is the set of* contingent constraints*, and* $C_f \doteq \{cf_1, ..., cf_h\}$ *is the set of* free constraints*.*

$$cc_i \doteq (e_i - b_{j_i}) \in [l_i, u_i] \quad cf_i \doteq \bigvee_{j=1}^{D_i} (x_{ij} - y_{ij}) \in [l_{ij}, u_{ij}]$$

such that: $j_i \in [1 \dots n]$, $l_i, u_j, l_{ij}, u_{ij} \in \mathbb{R}$, $l_i \leq u_j$, $l_{ij} \leq u_{ij}$, D_i *is the number of disjuncts for the* i*-th free constraint and* $x_{ij}, y_{ij} \in X_c \cup X_u$

Intuitively, time points belonging to X_c are time decisions that can be controlled by the agent, while time points in X_u are under the control of the environment. A similar subdivision is imposed on the constraints: free constraints C_f are constraints that the agent is required to fulfill, while contingent constraints C_c are the assumptions that the environment will fulfill. As in (Vidal and Fargier 1999) we consider only contingent constraints that start with a controllable time point. Thus, each uncontrollable time point is linked by exactly one contingent constraint to a controllable time point. However, this assumption does not affect the generality of the formalism, as for each contingent constraint $(e_i - e_j) \in [l, u]$ we can add an artificial controllable time point a, and add $(a - e_j) \in [0, 0]$ to the free constraints and $(e_i - a)$ to the contingent constraints.

Three nested TPU classes have been defined depending on the type of disjunctive structure allowed in the problem constraints. Definition 1 coincides to the literature definition of the *Disjunctive Temporal Problem with Uncertainty* (DTPU). If we admit only disjunctions on a single couple of variables the resulting problem is a *Temporal Constraint Satisfaction Problem with Uncertainty* (TCSPU) and if we disallow disjunctions we obtain a *Simple Temporal Problem with Uncertainty* (STPU).

For a TPU three different problems can be addressed: strong controllability, dynamic controllability and weak controllability (Vidal and Fargier 1999). In all these problems, the agent is required to fulfill all the free constraints under any possible assignment of the uncontrollable time points that fulfill the contingent constraints. The difference between them resides in the amount of observability that the agent is allowed to have over the environment decisions.

Weak controllability. Weak controllability consists in finding a function that maps a total assignment to uncontrollable time points to a total assignment to controllable time points. The function output must fulfill all the free constraints under any input assignment that fulfills the contingent constraints. We assume to have full observability over the uncontrollable evolution of the system. Intuitively, a problem is weakly controllable if for every valid assignment of the uncontrollable time points, there exists a winning *strategy* for the allocation of controllable time points, assuming to know in advance the values of all the uncontrollable time points.

In order to formally define weak controllability we perform some transformations. We first rewrite each uncontrollable time point e_i in terms of its time difference with its starting time point b_{j_i} by means of an uncontrollable offset y_i. For every contingent constraint cc_i, let $y_i \in \mathbb{R}$ be an offset for the uncontrollable duration such that: $y_i \geq 0$, $y_i \leq u_i - l_i$ and $e_i = b_{j_i} + u_i - y_i$. Intuitively, y_i represents the offset with respect to the maximal duration of the time difference between b_{j_i} and e_i, and can be used to rewrite all the constraints involving e_i in terms of b_{j_i} and y_i only. To simplify the notation, we introduce two vectors: $\vec{x}$ is the vector of controllable time points $(b_1, ..., b_n)$, and $\vec{y}$ is the vector of uncontrollable offsets $(y_1, ..., y_m)$. Thanks to the redefinition of each e_i in terms of y_i, the rewriting of the contingent constraints depends only on $\vec{y}$.

We call $\Gamma(\vec{y})$ the formula representing the conjunction of all the contingent constraints, and $\Psi(\vec{x}, \vec{y})$ the conjunction of all the free constraints rewritten in terms of $\vec{x}$ and $\vec{y}$.

$$\Gamma(\vec{y}) \doteq \bigwedge_{k=1}^{m} (y_k \geq 0) \wedge (y_k \leq (u_k - l_k)) \qquad \Psi(\vec{x}, \vec{y}) \doteq \bigwedge_{c \in C_f} c(\vec{x}, \vec{y})$$

In this setting, the weak controllability decision problem consists in deciding whether for every value of $\vec{y}$ such that $\Gamma(\vec{y})$ there exists an assignment to $\vec{x}$ such that $\Psi(\vec{x}, \vec{y})$ evaluates to $\top$.

Definition 2. *A TPU is* weakly controllable *if and only if* $\forall \vec{y}. \exists \vec{x}. (\Gamma(\vec{y}) \rightarrow \Psi(\vec{x}, \vec{y}))$ *is valid.*

In many cases, we are not only interested in checking weak controllability, but we also want to find the winning strategy for the agent. A weak strategy is a total function f that maps a total assignment to the uncontrollable offsets $\vec{y}$ to a total assignment to the controllable time points $\vec{x}$, if $\vec{y}$ satisfies $\Gamma(\vec{y})$. Otherwise, the strategy is inapplicable and returns $\bot$.

Definition 3. *A function* $f : \mathbb{R}^{|X_u|} \rightarrow \mathbb{R}^{|X_c|} \cup \bot$ *is a weak strategy for a TPU if*

$$f(\vec{y}) \doteq \begin{cases} \bot & \textit{if } \neg\Gamma(\vec{y}) \\ \vec{x} \mid \Psi(\vec{x}, \vec{y}) & \textit{Otherwise.} \end{cases}$$

This definition does not impose any constraint (e.g. linearity, continuity) on f other than the fact of being a function.

Encoding weak controllability into SMT

Looking at the weak controllability formal characterization in Definition 2 from an SMT perspective, it is clear that we are solving the validity problem of an LRA formula. Any SMT solver with full support of LRA is able to deal with that formula directly and it can correctly solve the problem. However, due to the high computational cost of directly handling quantifiers, an optimized encoding is in order.

We first rewrite the formula encoding weak controllability in Definition 2 by transforming the external universal quantifier into the negation of an existential one, and we consider the negation of the resulting formula. We call the resulting formula *inverted SMT encoding*.

$$\exists \vec{y}. \neg \exists \vec{x}. (\Gamma(\vec{y}) \rightarrow \Psi(\vec{x}, \vec{y}))$$

If this formula is unsatisfiable, then the problem is weakly controllable, while if it is satisfiable, then the problem is not weakly controllable. Intuitively the encoding is a search for an assignment to uncontrollable time points that is able to violate the free constraints under any possible strategy (it is a winning strategy for the environment). This encoding still requires a solver with full support of LRA, but is able to exploit the searching power of the SMT framework (the external quantifier is existential) and in case of non-weak controllability it allows for the extraction of debug information by providing a model of the formula.

A further improvement can be achieved by limiting as much as possible the scope of the quantified variables. To this extent, we push the existential quantifier over the implication, and thus the quantification is limited to the problem's free constraints only (ref. as *gamma extraction encoding*):

$$\exists \vec{y}. (\Gamma(\vec{y}) \wedge \neg \exists \vec{x}. \Psi(\vec{x}, \vec{y})).$$

For the special case of STPU, in (Vidal and Fargier 1999) the concept of *weak controllability on bounds* is presented: in order to check whether a STPU is weakly controllable it suffices to check whether it is controllable in all the extreme assignments of the uncontrollable values. This idea can also be exploited in the SMT framework. However, the resulting encoding would be exponentially big: the extreme points of the uncontrollable region are the $2^{|X_u|}$ vertexes of an hyper-rectangle in $|X_u|$ dimensions. Nevertheless, we will extend and use this result for the strategy extraction problem.

Extraction of linear strategies

A *linear strategy* is such that the value of every controllable time point is obtained as a linear combination of $\vec{y}$. In the following we discuss several approaches for extracting a linear strategy from a weakly controllable problem.

Encoding into NRA

Let $N \doteq |X_c|$ and $M \doteq |X_u|$, a linear strategy can be represented with a matrix A of real coefficients of size $N \times (M+1)$. The idea is to express every controllable variable as a linear function of the uncontrollable offsets.

$$x_i \doteq f_i(\vec{y}) \doteq A \cdot \left(\frac{\vec{y}}{1} \right) = A_{i,1} \cdot y_1 + \ldots + A_{i,M} \cdot y_M + A_{i,M+1}$$

Therefore, the matrix A must have one column for every offset and an additional column for the constant additive term. Equation 1 is an encoding into NRA for extracting a linear strategy for any TPU problem in a single check.

$$\exists A_{1,1}, \ldots, A_{n,m}. \forall \vec{y}. \Gamma(\vec{y}) \rightarrow \Psi(A \cdot \left(\frac{\vec{y}}{1} \right), \vec{y}) \qquad (1)$$

The idea is to let the solver search for the A_{ij} coefficients of the linear combination of $\vec{y}$ that represent the set of hyperplanes that are strategies for each $x \in \vec{x}$. If the solver reports unsatisfiable, it means that no linear strategy exists for the given problem. Unfortunately, this approach needs a solver supporting the quantified NRA theory, and to the best of our knowledge, no SMT solver currently supports this theory.

In the following, we restrict to STPU problems. For such problems we can exploit the convexity of the constraints to design cheaper algorithms for strategy extraction.

Reduction to quantifier-free LRA

If the problem is convex, given any two points in the solution space, any point in the line connecting these two points is also a solution. Following this idea we can generalize the result of weak controllability on bounds in (Vidal and Fargier 1999) to the search of linear strategies.

Theorem 1. *If a STPU admits a linear strategy $f(\vec{y})$ that fulfills the problem constraints in all the bounds of the uncontrollable region, than $f(\vec{y})$ is a valid linear strategy for the entire problem.*

The idea is to create a single SMT problem that encodes the problem with a symbolic strategy in all the extremes of the uncontrollable region. The encoding is obtained from the NRA formula in Equation 1 by substituting the universal quantifier with the instantiation of the problem free constraints ($\Psi(P\vec{y}, \vec{y})$) in all the extreme values of $\vec{y}$, that are the vertexes of the polyhedron represented by $\Gamma(\vec{y})$. Algorithm 1 shows the pseudo-code for extracting a linear strategy with such encoding.

Algorithm 1 Linear strategy extraction (LSE)

procedure LINEARSTRATEGY($\Gamma(\vec{y})$, $\Psi(\vec{x}, \vec{y})$)
  $P \leftarrow$ VARIABLEMATRIX($|X_c|$,$|X_u|+1$)
  $\phi(P) \leftarrow \top$
  for all $\vec{c} \in$ EXTREMEASSIGNMENTS($\Gamma(\vec{y})$) **do**
    $\phi(P) \leftarrow \phi(P) \wedge \Psi(P \cdot \vec{c}, \vec{c})$
  end for
  if SMT($\phi(P)$) **then**
    return GETMODEL()
  end if
  return None
end procedure

In the pseudocode, the function VARIABLEMATRIX creates a matrix of real variables, while the function EXTREMEASSIGNMENTS generates all the vertexes of $\Gamma(\vec{y})$. The function SMT checks the satisfiability of the given SMT formula using a solver, while GETMODEL returns the produced model in case of SAT answer. Note that, this approach leads to an exponential blowup in the size of the SMT problem: this is a strong limitation on the scalability of the approach, nevertheless this is the first practically-working approach for linear strategy extraction.

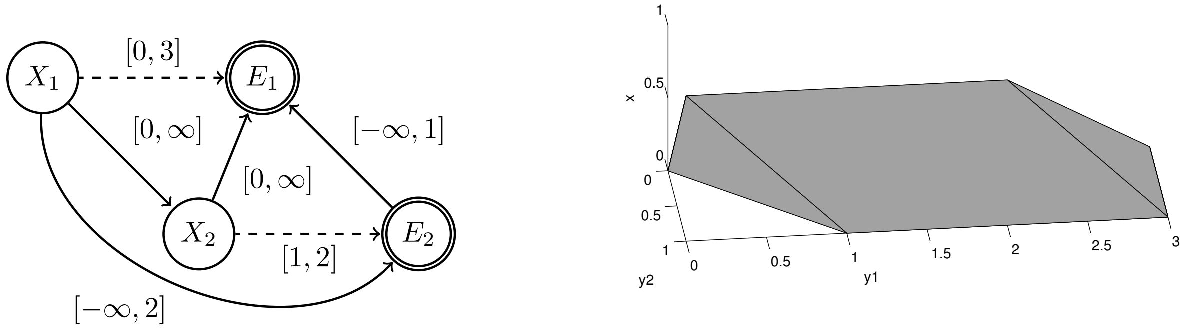
From strong to weak controllability: incremental weakening

In order to limit the exponential blowup of the previous encoding to the worst case only, we developed another approach called "*incremental weakening*" that tries to limit the number of coefficients to search for and to reduce the amount of observability needed to solve the problem. The underlying idea is to start from solving a strong controllability problem, if a solution is found, the strong assignment is a valid weak linear strategy for the problem, otherwise an uncontrollable offset y_p is heuristically picked and marked as observable. The algorithm then tries to build a linear strategy that relies on y_p only. This is done by exploding in the extremes of y_p the strong controllability problem defined on the remaining uncontrollable offsets only. If the algorithm fails in finding a linear strategy, another variable is picked and the approach is iterated, until all the uncontrollable offsets are marked as observable and the encoding coincides with the previous approach. The pseudo-code of this method is reported in Algorithm 2.

Algorithm 2 Incremental weakening LSE

procedure IWLINEARSTRATEGY($\Gamma(\vec{y})$, $\Psi(\vec{x}, \vec{y})$)
  while $\vec{p} \leftarrow$ GETHEURISTICPIVOTS($\Gamma(\vec{y})$) **do**
    $\vec{n} \leftarrow \{y \in \vec{y} \mid y \notin \vec{p}\}$
    $\eta(\vec{x}, \vec{p}) \leftarrow$ SC_ENC($\Gamma(\vec{n})$, $\Psi(\vec{x}, \vec{n})$)
    $Res \leftarrow$ LINEARSTRATEGY($\Gamma(\vec{p})$, $\eta(\vec{x}, \vec{p})$)
    if $Res \neq$ None **then**
      return Res
    end if
  end while
  return None
end procedure

In the pseudocode, the function GETHEURISTICPIVOTS



(a) (b)

Figure 1: (a) A STPU that does not admit a linear strategy. Controllable time points are represented as circled nodes, uncontrollable nodes are represented as double circled nodes, free constraints are represented as solid arrows and contingent constraints as dashed arrows. (b) The region of feasibility of the STPU in the space of X_2, Y_1 and Y_2.

shall return an heuristically computed subset of $\vec{y}$ and eventually must return the entire $\vec{y}$ that terminates the algorithm; the function SC_ENC produces the encoding of a strong controllability problem in SMT, while the function LINEARSTRATEGY is the function described in Algorithm 1. This algorithm is highly dependent on the heuristic used for $\vec{p}$ extraction: the number of cycles of the algorithm directly depends on the heuristic.

In principle, one could substitute the use of strong controllability in this algorithm with dynamic controllability, as they both imply weak controllability (Vidal and Fargier 1999) and are tractable for STPU. Nevertheless, a yes/no answer is not enough in this context: we need an equivalent encoding of the controllability problem itself. Encoding strong controllability of STPU is relatively simple using the worst-case argument proposed by Vidal in (Vidal and Fargier 1999), while we are not aware of any encoding of dynamic controllability in the SMT framework.

Linearity is not enough

A linear strategy is very useful in practice: it is compact to represent and easy to evaluate. In general, unfortunately, a weakly controllable STPU is not guaranteed to have such a strategy. Let us consider the STPU depicted in Figure 1(a). This STPU is weakly controllable, but it does not admit a linear strategy. In Figure 1(b) we plotted the solution space of the STPU problem in the space (Y_1, Y_2, X_2) regions (Without loss of generality we projected the solution space in $X_1 = 0$). The plot clearly shows that there exists no linear strategy for X_2: considering the extreme $(0, 0)$ in the space (Y_1, Y_2), a linear solution must contain the point $(0, 0, 0)$; considering $(0, 1)$ we must include $(0, 1, 1)$, considering $(3, 0)$ we must include $(3, 0, 0)$ and for $(3, 1)$ the linear solution must include the point $(3, 1, 0)$. However, no linear solution can exists, as no 3D-plane can contain all of them at the same time.

Extraction of piecewise-linear strategies

Given that there is no guarantee that a linear strategy always exists, we investigate the existence of a *piecewise-linear strategy*. A piecewise-linear strategy is defined by cases over a splitting of the uncontrollable offsets valid region, in which every region admits a linear strategy. Thus, an executor will have to check in which region the actual value of $\vec{y}$ is contained in, and successively apply the corresponding strategy.

Definition 4. *A* piecewise-linear strategy *is a function*

$$f(\vec{y}) \doteq \begin{cases} S_1(\vec{y}) & \textit{if } \eta_1(\vec{y}) \\ ... & \\ S_k(\vec{y}) & \textit{if } \eta_k(\vec{y}) \\ \bot & \textit{Otherwise} \end{cases}$$

where S_i are linear strategies and $\eta_i(\vec{y})$ are sub-regions of $\Gamma(\vec{y})$ such that $(\bigvee_{i=1}^{k} \eta_i(\vec{y})) \leftrightarrow \Gamma(\vec{y})$.

It can be proved that, a piecewise-linear strategy always exists for any weakly controllable STPU.

Theorem 2. *For any given STPU P, if P is weakly controllable, then P admits a piecewise-linear strategy.*

Proof. (Sketch) Consider the solution space for the given problem P, since P is a STPU it is a convex polyhedron. Since P is weakly controllable, the projection of the polyhedron in the uncontrollable offsets dimensions, entirely covers the space of the valid $\vec{y}$ which is an hyper-rectangle. If we focus on the skin of the solution space, it is composed of exactly two piecewise-linear continuous strategies. □

In the following, we present two algorithms for extracting a piecewise-linear strategy for a weakly controllable STPU.

Simplexes decomposition

A simple and direct approach to extract a piecewise-linear strategy consists in partitioning the region of the uncontrollables in a set of simplexes (hyper-tetrahedra); we use these polyhedra because they are the minimal polytopes. For every simplex, a linear strategy is guaranteed to exist, as it is always possible to force a d-dimensional hyper-plane to pass over all the vertexes of a d-simplex.

A straightforward method to obtain the simplexes is to use the extreme vertexes of the uncontrollable offsets region. The number of derived simplexes is factorial w.r.t. the number of uncontrollable offsets. For each simplex it is possible to find a linear strategy separately by enforcing an hyper-plane to satisfy the problem constraints in all the simplex vertexes. Algorithm 3 shows the pseudo-code for extracting a piecewise linear strategy enumerating all the simplexes. The complexity of this algorithm is very high because of the enumeration of all the $(|X_u|!)$ simplexes.

Algorithm 3 Piecewise-linear strategy extraction

procedure GETSTRATEGY($\Gamma(\vec{y})$, $\Psi(\vec{x}, \vec{y})$)
  $P \leftarrow \emptyset$
  for all $s \in$ EXTREMALSIMPLEXES($\Gamma(\vec{y})$) **do**
    $L \leftarrow$ GETSIMPLEXLINEARSTRATEGY(s, $\Gamma(\vec{y})$, $\Psi(\vec{x}, \vec{y})$)
    $P \leftarrow P \cup \{$("IF $\vec{y} \in s$ THEN L ")$\}$
  end for
  return P
end procedure

In the pseudocode, the function EXTREMALSIMPLEXES iterates over all the simplexes needed to cover the $\Gamma(\vec{y})$ polyhedron, while GETSIMPLEXLINEARSTRATEGY returns a linear strategy suitable for the given simplex. We represent a piecewise strategy as a set P of linear sub-strategies, each conditioned to some region (The statement "IF $\vec{y} \in s$ THEN L" represent the conditioning of the linear strategy L in the region s).

Lazy expansion

To overcome the complexity limitation of the previous approach we developed a second technique, called *lazy expansion*, that first selects a simplex in the uncontrollable region and finds a linear strategy in that simplex. Second, we symbolically compute the region of the uncontrollable offsets that is satisfied by the computed strategy. Third, we associate the computed strategy to the resulting region. Finally, we search a new simplex in the remaining part of the uncontrollable space, until the uncontrollable offsets space is completely covered. The main advantage of this algorithm with respect to the previous one is that it is not forced to enumerate all the possible simplexes because the computed strategy once found is exploited in all the possible points of the space where it is applicable. Algorithm 4 shows the pseudo-code for extracting a piecewise linear strategy exploiting lazy expansion. The difficulty in this approach resides in efficiently finding a simplex from a symbolically-defined region: the choice of the simplex is pivotal for the strategy extraction. A naïve approach consists in extracting vertexes of the simplex using simple queries to an SMT solver.

Algorithm 4 Lazy piecewise-linear strategy extraction

procedure GETSTRATEGY($\Gamma(\vec{y})$, $\Psi(\vec{x}, \vec{y})$)
  $P \leftarrow \emptyset$
  $\eta(\vec{y}) \leftarrow \Gamma(\vec{y})$
  while SMT($\eta(\vec{y})$) **do**
    $s \leftarrow$ SIMPLEX($\eta(\vec{y})$)
    $S \leftarrow$ GETSIMPLEXLINEARSTRATEGY(s, $\Gamma(\vec{y})$, $\Psi(\vec{x}, \vec{y})$)
    $P \leftarrow P \cup \{$("IF $\eta(\vec{y}) \wedge \Psi(S(\vec{y}), \vec{y})$ THEN S ")$\}$
    $\eta(\vec{y}) \leftarrow \eta(\vec{y}) \wedge \neg\Psi(S(\vec{y}), \vec{y})$
  end while
  return P
end procedure

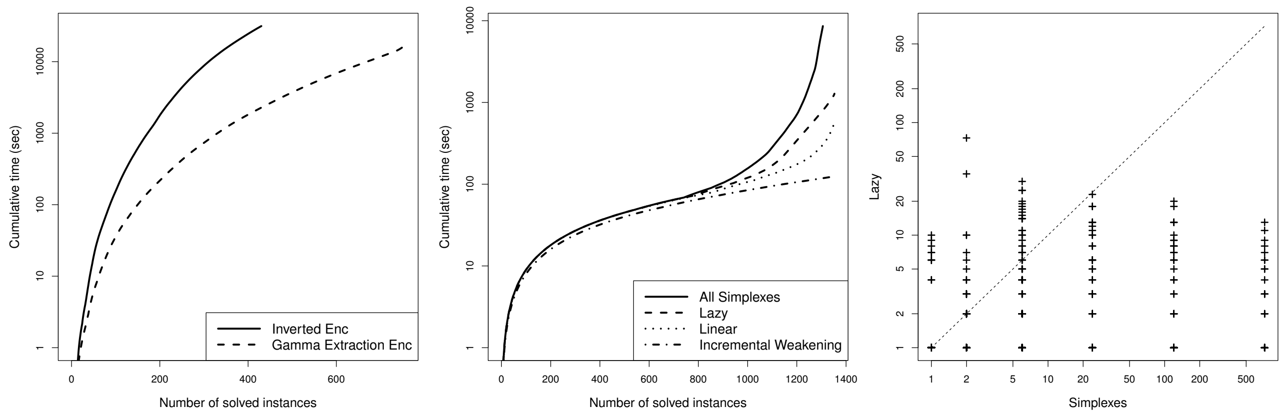
Experimental evaluation

In order to empirically test the effectiveness of the proposed approaches, we implemented a tool written in Python that reads a TPU problem, and applies to it the portfolio of encodings we discussed. We used the *Z3* (de Moura and Bjørner 2008) SMT solver for the weak controllability decision problem, while we rely on the Python API provided by the *MathSAT5* (Cimatti et al. 2011) SMT solver for all the strategy-extraction techniques. All the tests have been performed on a Linux workstation equipped with a quad-core Q6600 CPU and 4GB of RAM memory. We considered a memory limit of 2GB, a time-out of 300 seconds and we used sequential, single-core computation only.

We tested the decision problem encoding over a set of 1524 randomly generated DTPU, TCSPU and STPU instances, with a number of time points ranging from 6 to 19956. For the random generation of problem instances we extended the generator presented in (Armando, Castellini, and Giunchiglia 1999) to produce TPU's. For the evaluation of the strategy-extraction techniques, we used 1354 weakly controllable STPU benchmarks.

The results of checking the decision problem over the set of TPU's are plotted in Figure 2(a). This plot shows the cumulative time (in logarithmic scale) needed to build and solve the corresponding encoding on the y-axis and the number of solved problems on the x-axis. The plot highlights the fact that Z3 performs much better when the *gamma extraction encoding* of the problem is considered: in fact, this approach is able to solve, in less time, an higher number of instances with respect to the *inverted* encoding.

The results for the evaluation of the strategy-extraction techniques for the 1354 benchmarks are reported in Figure 2(b). The plot considers only those benchmarks for which there exists a linear strategy, and compares the four different approaches. It is evident from the plot that for linear strategies, the incremental weakening approach outperforms all the others. The method enumerating all the simplexes quickly explodes due to the factorial complexity of their enumeration. Although, the techniques for piecewise-linear strategy extraction are penalized as they are strictly more general than the others, the plot shows that lazy approach is much faster than the simplex enumeration. In Figure 2(c) we plotted the number of "pieces" of the strategies for the lazy and simplexes methods. The plot clearly shows that, although for small numbers of uncontrollable variables the lazy approach generates additional not needed "pieces", when the size increases the number of "pieces" identified by the lazy method is much smaller than for the simplexes one. We also experimented cases with 16 uncontrollable variables in which the lazy approach can terminate in 4 iterations. In general, the lazy approach has a huge gain in performances



(a) (b) (c)

Figure 2: (a) Results for WC decision problem using Z3 SMT solver. (b) Results for linear strategy extraction problem. (c) Size of the strategy for piecewise-linear algorithms expressed as the number of splitted regions.

and in strategy size, especially for sparse problems.

Conclusions and future work

In this paper we presented a comprehensive approach to the problem of weak controllability for temporal problems under uncertainty. The approach is a cast of the problem in the SMT framework, and provides both a decision procedure, and various constructive forms of strategy extraction.

The "incremental weakening" approach for linear strategy extraction and the "lazy" approach for piecewise-linear strategy extraction, are strongly influenced by heuristics that we surely want to optimize. In particular, we are investigating the possibility to use topological information for the generation of effective subsets of $\vec{y}$ in "incremental weakening", and the use of extremal simplexes in the "lazy" approach.

In the future, we plan to study the applicability of SMT proof-extraction techniques to the strategy construction problem: the capability of modern solvers to extract a proofs of unsatisfiability could be useful to build the strategy. In addition, we plan to tackle the problem of dynamic controllability, where strategies are required to rely only on the observation of events that have (instead of being "clairvoyant", i.e. assuming perfect forecast). In particular, we plan to combine the approaches to strong controllability and the one developed in this paper to explore the continuum between strong and weak controllability.

References

Allen, J. F. 1983. Maintaining knowledge about temporal intervals. *Commun. ACM* 26(11):832–843.

Armando, A.; Castellini, C.; and Giunchiglia, E. 1999. Sat-based procedures for temporal reasoning. In *ECP*, 97–108.

Barrett, C. W.; Sebastiani, R.; Seshia, S. A.; and Tinelli, C. 2009. Satisfiability modulo theories. In *Handbook of Satisfiability*. IOS Press. 825–885.

Cimatti, A.; Griggio, A.; Sebastiani, R.; and Schaafsma, B. 2011. The MathSAT 5 SMT solver. `http://mathsat.fbk.eu`.

de Moura, L. M., and Bjørner, N. 2008. Z3: An Efficient SMT Solver. In *TACAS*, 337–340.

Dechter, R.; Meiri, I.; and Pearl, J. 1991. Temporal constraint networks. *Artif. Intell.* 49(1-3):61–95.

Morris, P. H., and Muscettola, N. 2005. Temporal dynamic controllability revisited. In *AAAI*, 1193–1198.

Morris, P. H.; Muscettola, N.; and Vidal, T. 2001. Dynamic control of plans with temporal uncertainty. In *IJCAI*, 494–502.

Peintner, B.; Venable, K. B.; and Yorke-Smith, N. 2007. Strong controllability of disjunctive temporal problems with uncertainty. In *CP*, 856–863.

Tsamardinos, I., and Pollack, M. E. 2003. Efficient solution techniques for disjunctive temporal reasoning problems. *Artificial Intelligence* 151(12):43 – 89.

Venable, K. B.; Volpato, M.; Peintner, B.; and Yorke-Smith, N. 2010. Weak and dynamic controllability of temporal problems with disjunctions and uncertainty. In *Proceedings of ICAPS10 Workshop on Constraint Satisfaction Techniques for Planning and Scheduling Problems (COPLAS10)*.

Vidal, T., and Fargier, H. 1999. Handling contingency in temporal constraint networks: from consistency to controllabilities. *J. Exp. Theor. Artif. Intell.* 11(1):23–45.