Instantaneous Reaction-Time in Dynamic-Consistency Checking of Conditional Simple Temporal Networks

Massimo Cairo, Carlo Comin, Romeo Rizzi

Instantaneous Reaction-Time in Dynamic-Consistency Checking of Conditional Simple Temporal Networks

Massimo Cairo
Department of Mathematics
University of Trento, Trento, Italy
(massimo.cairo@unitn.it)

Carlo Comin
Department of Mathematics
University of Trento, Trento, Italy;
LIGM (UMR 8049)
Université Paris-Est, Marne-la-Vallée, France
(carlo.comin@unitn.it)

Romeo Rizzi
Department of Computer Science
University of Verona, Verona, Italy
(romeo.rizzi@univr.it)

***Abstract*—Conditional Simple Temporal Network (CSTN) is a constraint-based graph-formalism for conditional temporal planning. Three notions of consistency arise for CSTNs and CSTPs: weak, strong, and *dynamic*. Dynamic-Consistency (DC) is the most interesting notion, but it is also the most challenging. In order to address the DC-Checking problem, in [5] we introduced ε-DC (a refined, more realistic, notion of DC), and provided an algorithmic solution to it. Next, given that DC implies ε-DC for some sufficiently small $\varepsilon > 0$, and that for every $\varepsilon > 0$ it holds that ε-DC implies DC, we offered a sharp lower bounding analysis on the critical value of the *reaction-time* $\hat{\varepsilon}$ under which the two notions coincide. This delivered the first (pseudo) singly-exponential time algorithm for the DC-Checking of CSTNs. However, the ε-DC notion is interesting per se, and the ε-DC-Checking algorithm in [5] rests on the assumption that the reaction-time satisfies $\varepsilon > 0$; leaving unsolved the question of what happens when $\varepsilon = 0$. In this work, we introduce and study π-DC, a sound notion of DC with an *instantaneous* reaction-time (i.e., one in which the planner can react to any observation at the same instant of time in which the observation is made). Firstly, we demonstrate by a counter-example that π-DC is not equivalent to 0-DC, and that 0-DC is actually inadequate for modeling DC with an instantaneous reaction-time. This shows that the main results obtained in our previous work do not apply directly, as they were formulated, to the case of $\varepsilon = 0$. Motivated by this observation, as a second contribution, our previous tools are extended in order to handle π-DC, and the notion of *ps-tree* is introduced, also pointing out a relationship between π-DC and HyTN-Consistency. Thirdly, a simple *reduction* from π-DC to DC is identified. This allows us to design and to analyze the first sound-and-complete π-DC-Checking procedure. Remarkably, the time complexity of the proposed algorithm remains (pseudo) singly-exponential in the number of propositional letters.**

***Index Terms*—Conditional Simple Temporal Networks, Dynamic-Consistency, Instantaneous Reaction-Time, Hyper Temporal Networks, Singly-Exponential Time.**

I. Introduction and Motivation

In *temporal planning* and *temporal scheduling*, *Simple Temporal Networks (STNs)* [6] are directed weighted graphs, where nodes are events to be scheduled in time and arcs represent temporal distance constraints between pairs of events.

This work is focused on the *Conditional Simple Temporal Problem (CSTP)* [12] and its graph-based counterpart *Conditional Simple Temporal Network (CSTN)* [10], a constraint-based model for conditional temporal planning. The CSTN formalism extends STNs in that: (1) some of the nodes are called *observation events* and to each of them is associated a propositional letter, to be disclosed only at execution time; (2) *labels* (i.e. conjunctions over the literals) are attached to all nodes *and* constraints, to indicate the scenarios in which each of them is required. The planning agent (planner) must schedule all the required nodes, while respecting all the required temporal constraints among them. This extended framework allows for the off-line construction of conditional plans that are guaranteed to satisfy complex temporal constraints. Importantly, this can be achieved even while allowing the decisions about the precise timing of actions to be postponed until execution time, in a least-commitment manner, thereby adding flexibility and making it possible to adapt the plan dynamically, during execution, in response to the observations that are made [12].

Then, three notions of consistency arise for CSTNs: weak, strong, and *dynamic*. Dynamic-Consistency (DC) is the most interesting one, as it requires the existence of conditional plans where decisions about the precise timing of actions are postponed until execution time, but anyhow guaranteeing that all of the relevant constraints will be ultimately satisfied. Still, it is the most challenging one and it was conjectured to be hard to assess by Tsamardinos, Vidal and Pollack [12].

In our previous work [5], it was unveiled that HyTNs and MPGs are a natural underlying combinatorial model for the DC-Checking of CSTNs. Indeed, STNs have been recently generalized into *Hyper Temporal Networks (HyTNs)* [3], [4], by considering weighted directed hypergraphs, where each hyperarc models a disjunctive temporal constraint called *hyper-constraint*. In [3], [4], the computational equivalence between checking the consistency of HyTNs and determining the win-

This work was partially supported by *Department of Computer Science, University of Verona, Verona, Italy*, under PhD grant "Computational Mathematics and Biology"; and, for the 2nd author, on a co-tutélle agreement with *LIGM, Université Paris-Est in Marne-la-Vallée, Paris, France*.

ning regions in Mean Payoff Games (MPGs) was also pointed out. The approach was shown to be viable and robust thanks to some extensive experimental evaluations [4]. MPGs [1], [7], [13] are a family of two-player infinite games played on finite graphs, with direct and important applications in model-checking and formal verification [8], and they are known for having theoretical interest in computational complexity for their special place among the few (natural) problems lying in $\mathsf{NP} \cap \mathsf{coNP}$.

All this combined, in [5] it was provided the first (pseudo) singly-exponential time algorithm for the DC-Checking problem, also producing a dynamic execution strategy whenever the input CSTN is DC. For this, it was introduced ε-DC (a refined, more realistic, notion of DC), and provided the first algorithmic solution to it. Next, given that DC implies ε-DC for some sufficiently small $\varepsilon > 0$, and that for every $\varepsilon > 0$ it holds that ε-DC implies DC, it was offered a sharp lower bounding analysis on the critical value of the *reaction-time* $\hat{\varepsilon}$ under which the two notions coincide. This delivered the first (pseudo) singly-exponential time algorithm for the DC-Checking of CSTN. However, the ε-DC notion is interesting per se, and the ε-DC-Checking algorithm in [5] rests on the assumption that the reaction-time satisfies $\varepsilon > 0$; leaving unsolved the question of what happens when $\varepsilon = 0$.

Contribution: In this work we introduce and study π-DC, a sound notion of DC with an *instantaneous* reaction-time (i.e., one in which the planner can react to any observation at the same instant of time in which the observation is made). Firstly, we provide a counter-example showing that π-DC is not just the $\varepsilon = 0$ special case of ε-DC. This implies that the algorithmic results obtained in [5] do not apply directly to the study of those situation where the planner is allowed to react instantaneously. Motivated by this observation, as a second contribution, we extend the previous formulation to capture a sound notion of DC with an instantaneous reaction-time, i.e., π-DC. Basically, it turns out that π-DC needs to consider an additional internal ordering among all the observation nodes that occur at the same time instant. Next, the notion of *ps-tree* is introduced to reflect the ordered structure of π-DC, also pointing out a relationship between π-DC and HyTN-Consistency. Thirdly, a simple *reduction* from π-DC to DC is identified. This allows us to design and to analyze the first sound-and-complete π-DC-Checking procedure. The time complexity of the proposed algorithm remains (pseudo) singly-exponential in the number $|P|$ of propositional letters.

II. Background

This section provides some background and preliminary notations. To begin, if $G = (V, A)$ is a directed weighted graph, every arc $a \in A$ is a triplet (u, v, w_a) where $u = t(a) \in V$ is the *tail* of a, $v = h(a) \in V$ is the *head* of a, and $w_a = w(u, v) \in \mathbf{Z}$ is the (integer) *weight* of a.

Let us now recall Simple Temporal Networks (STNs) [6].

Definition 1 (STNs)**.** *An STN [6] is a weighted directed graph whose nodes are events that must be placed on the real time line and whose arcs, called* standard arcs*, express binary constraints on the allocations of their end-points in time.*

An STN $G = (V, A)$ is called consistent if it admits a feasible schedule*, i.e., a schedule $\phi : V \mapsto \mathbf{R}$ such that $\phi(v) \leq \phi(u) + w(u, v)$ for all arcs $(u, v, w(u, v)) \in A$.*

A. Conditional Simple Temporal Networks

Let us recall the CSTN model from [5], [10], [12].

Let P be a set of propositional letters (boolean variables), a *label* is any (possibly empty) conjunction of letters, or negations of letters, drawn from P. The *empty label* is denoted by λ. The set of all these labels is denoted by P^*. Two labels, ℓ_1 and ℓ_2, are called *consistent*, denoted by $Con(\ell_1, \ell_2)$, when $\ell_1 \wedge \ell_2$ is satisfiable. A label ℓ_1 *subsumes* a label ℓ_2, denoted by $Sub(\ell_1, \ell_2)$, whenever $\ell_1 \Rightarrow \ell_2$ holds. We are now in the position to recall the formal definition of CSTNs.

Definition 2 (CSTNs [10], [12])**.** *A* Conditional Simple Temporal Network (CSTN) *is a tuple $\langle V, A, L, \mathcal{O}, \mathcal{O}V, P\rangle$ where:*

- *V is a finite set of* events*; $P = \{p_1, \ldots, p_{|P|}\}$ is a finite set of* propositional letters*;*
- *A is a set of* labelled temporal constraints (LTCs) *each having the form $\langle v - u \leq w(u, v), \ell\rangle$, where $u, v \in V$, $w(u, v) \in \mathbf{Z}$, and $\ell \in P^*$ is satisfiable;*
- *$L : V \to P^*$ assigns a label to each event in V;*
- *$\mathcal{O}V \subseteq V$ is a finite set of* observation events*; $\mathcal{O} : P \to \mathcal{O}V$ is a bijection that associates a unique observation event $\mathcal{O}(p) = \mathcal{O}_p$ to each proposition $p \in P$;*
- *The following* reasonability assumptions *must hold:*
  *(*WD1*) for any LTC $\langle v - u \leq w, \ell\rangle \in A$ the label ℓ is satisfiable and subsumes both $L(u)$ and $L(v)$; intuitively, whenever a constraint $\langle v - u \leq w\rangle$ is required, then its endpoints u and v must be scheduled (sooner or later);*
  *(*WD2*) for each $p \in P$ and each $u \in V$ such that either p or $\neg p$ appears in $L(u)$, we require:* $\mathrm{Sub}(L(u), L(\mathcal{O}_p))$*, and $\langle \mathcal{O}_p - u \leq 0, L(u)\rangle \in A$; intuitively, whenever a label $L(u)$, for some $u \in V$, contains some $p \in P$, and u is eventually scheduled, then $\mathcal{O}_p = \mathcal{O}(p)$ must be scheduled before or at the same time of u.*
  *(*WD3*) for each labelled constraint $\langle v - u \leq w, \ell\rangle$ and $p \in P$, for which either p or $\neg p$ appears in ℓ, it holds that* $\mathrm{Sub}(\ell, L(\mathcal{O}_p))$*; intuitively, assuming that a required constraint contains some $p \in P$, then $\mathcal{O}_p = \mathcal{O}(p)$ must be scheduled (sooner or later).*

In all of the following definitions we shall implicitly refer to some CSTN $\Gamma = \langle V, A, L, \mathcal{O}, \mathcal{O}V, P\rangle$.

Definition 3 (Scenario)**.** *A* scenario *over a subset $U \subseteq P$ of boolean variables is a truth assignment $s : U \to \{0, 1\}$, i.e., s is a function that assigns a truth value to each proposition $p \in U$. When $U \subsetneq P$ and $s : U \to \{0, 1\}$, then s is said to be a* partial *scenario; otherwise, when $U = P$, then s is said to be a* (complete) *scenario. The set comprising all of the complete scenarios over P is denoted by Σ_P. If $s \in \Sigma_P$ is a scenario and $\ell \in P^*$ is a label, then $s(\ell) \in \{0, 1\}$ denotes the truth value of ℓ induced by s in the natural way.*

Notice that any scenario $s \in \Sigma_P$ can be described by means of the label $\ell_s \triangleq l_1 \wedge \cdots \wedge l_{|P|}$ such that, for every $1 \le i \le |P|$, the literal $l_i \in \{p_i, \neg p_i\}$ satisfies $s(l_i) = 1$.

Example 1. *Consider the set of boolean variables $P = \{p, q\}$. The scenario $s : P \to \{0,1\}$ defined as $s(p) = 1$ and $s(q) = 0$ can be compactly described by the label $\ell_s = p \wedge \neg q$.*

Definition 4 (Schedule). *A* schedule *for a subset of events $U \subseteq V$ is a function $\phi : U \to \mathbf{R}$ that assigns a real number to each event in U. The set of all schedules over U is denoted Φ_U.*

Definition 5 (Scenario-Restriction). *Let $s \in \Sigma_P$ be a scenario. The* restriction *of V, $\mathcal{O}V$, and A w.r.t. s are defined as follows:*

- $V_s^+ \triangleq \{v \in V \mid s(L(v)) = 1\}$; $\mathcal{O}V_s^+ \triangleq \mathcal{O}V \cap V_s^+$;
- $A_s^+ \triangleq \{\langle u, v, w\rangle \mid \exists \ell \, \langle v - u \le w, \ell\rangle \in A, s(\ell) = 1\}$.

The restriction of Γ w.r.t. $s \in \Sigma_P$ is the STN $\Gamma_s^+ \triangleq \langle V_s^+, A_s^+\rangle$. Finally, it is worth to introduce the notation $V_{s_1,s_2}^+ \triangleq V_{s_1}^+ \cap V_{s_2}^+$.

Definition 6 (Execution-Strategy). *An* Execution-Strategy (ES) *for Γ is a mapping $\sigma : \Sigma_P \to \Phi_{V_s^+}$ such that, for any scenario $s \in \Sigma_P$, the domain of the schedule $\sigma(s)$ is V_s^+. The set of ESs of Γ is denoted by $\mathcal{S}_\Gamma$. The* execution time *of an event $v \in V_s^+$ in the schedule $\sigma(s) \in \Phi_{V_s^+}$ is denoted by $[\sigma(s)]_v$.*

Definition 7 (History). *Let $\sigma \in \mathcal{S}_\Gamma$ be any ES, let $s \in \Sigma_P$ be any scenario and let $\tau \in \mathbf{R}$. The* history $\mathrm{Hst}(\tau, s, \sigma)$ *of τ in the scenario s under strategy σ is defined as:* $\mathrm{Hst}(\tau, s, \sigma) \triangleq \big\{\big(p, s(p)\big) \in P \times \{0,1\} \mid \mathcal{O}_p \in V_s^+,\ [\sigma(s)]_{\mathcal{O}_p} < \tau\big\}$.

The history can be compactly encoded as the conjunction of the literals corresponding to the observations comprising it, that is, by means of a label.

Definition 8 (Viable Execution-Strategy). *We say that $\sigma \in \mathcal{S}_\Gamma$ is a* viable *ES if, for each scenario $s \in \Sigma_P$, the schedule $\sigma(s) \in \Phi_{V_s^+}$ is feasible for the STN Γ_s^+.*

Definition 9 (Dynamic-Consistency). *An ES $\sigma \in \mathcal{S}_\Gamma$ is called* dynamic *if, for any $s_1, s_2 \in \Sigma_P$ and any $v \in V_{s_1,s_2}^+$, the following implication holds on $\tau \triangleq [\sigma(s_1)]_v$:*

$$\mathrm{Con}(\mathrm{Hst}(\tau, s_1, \sigma), s_2) \Rightarrow [\sigma(s_2)]_v = \tau.$$

We say that Γ is dynamically-consistent (DC) *if it admits $\sigma \in \mathcal{S}_\Gamma$ which is both viable and dynamic. The problem of checking whether a given CSTN is DC is named* DC-*Checking.*

We provide next the definition of *difference set* $\Delta(s_1; s_2)$.

Definition 10 (Difference-Set). *Let $s_1, s_2 \in \Sigma_P$ be any two scenarios. The set of observation events in $\mathcal{O}V_{s_1}^+$ at which s_1 and s_2 differ is denoted by $\Delta(s_1; s_2)$. Formally,*

$$\Delta(s_1; s_2) \triangleq \big\{\mathcal{O}_p \in \mathcal{O}V_{s_1}^+ \mid s_1(p) \neq s_2(p)\big\}.$$

The various definitions of history and dynamic consistency that are used by different authors [5], [11], [12] are equivalent.

B. Hyper Temporal Networks

This subsection surveys the *Hyper Temporal Network* (HyTN) model, which is a strict generalization of STNs. The reader is referred to [3], [4] for an in-depth treatise on HyTNs.

Definition 11 (Hypergraph). *A* hypergraph *$\mathcal{H}$ is a pair $(V, \mathcal{A})$, where V is the set of nodes, and $\mathcal{A}$ is the set of* hyperarcs. *Each hyperarc $A = (t_A, H_A, w_A) \in \mathcal{A}$ has a distinguished node t_A called the* tail *of A, and a nonempty set $H_A \subseteq V \setminus \{t_A\}$ containing the* heads *of A; to each head $v \in H_A$ is associated a* weight *$w_A(v) \in \mathbf{Z}$.*

Provided that $|A| \triangleq |H_A \cup \{t_A\}|$, the *size* of a hypergraph $\mathcal{H} = (V, \mathcal{A})$ is defined as $m_{\mathcal{A}} \triangleq \sum_{A \in \mathcal{A}} |A|$; it is used as a measure for the encoding length of $\mathcal{H}$. If $|A| = 2$, then $A = (u, v, w)$ can be regarded as a *standard arc*. In this way, hypergraphs generalize graphs.

A HyTN is a weighted hypergraph $\mathcal{H} = (V, \mathcal{A})$ where a node represents an *event* to be scheduled, and a hyperarc represents a set of temporal distance *constraints* between the *tail* and the *heads*.

In the HyTN framework the consistency problem is the following decision problem.

Definition 12 (HyTN-Consistency). *Given some HyTN $\mathcal{H} = (V, \mathcal{A})$, decide whether there is a schedule $\phi : V \to \mathbf{R}$ such that:*

$$\phi(t_A) \ge \min_{v \in H_A} \{\phi(v) - w_A(v)\},\ \forall\, A \in \mathcal{A}$$

any such a schedule $\phi : V \to \mathbf{R}$ is called feasible.

A HyTN is called consistent whenever it admits at least one feasible schedule. The problem of checking whether a given HyTN is consistent is named HyTN-Consistency.

Theorem 1. *[3] There exists an $O((|V|+|\mathcal{A}|)m_{\mathcal{A}}W)$ pseudo-polynomial time algorithm for checking HyTN-Consistency; moreover, when the input HyTN $\mathcal{H} = (V, \mathcal{A})$ is consistent, the algorithm returns as output a feasible schedule $\phi : V \to \mathbf{R}$ of $\mathcal{H}$; Here, $W \triangleq \max_{A \in \mathcal{A}, v \in H_A} |w_A(v)|$.*

C. ε-Dynamic-Consistency

In CSTNs, decisions about the precise timing of actions are postponed until execution time, when informations meanwhile gathered at the observation nodes can be taken into account. However, the planner is allowed to factor in an outcome, and differentiate its strategy according to it, only *strictly* after the outcome has been observed (whence the strict inequality in Definition 7). Notice that this definition does not take into account the reaction-time, which, in most applications, is non-negligible. In order to deliver algorithms that can also deal with the *reaction-time* $\varepsilon > 0$ of the planner, we introduced in [5] a refined notion of DC.

Definition 13 (ε-Dynamic-Consistency). *Given any CSTN $\langle V, A, L, \mathcal{O}, \mathcal{O}V, P\rangle$ and any real number $\varepsilon \in (0, +\infty)$, an ES $\sigma \in \mathcal{S}_\Gamma$ is* ε-dynamic *if it satisfies all of the H_ε-constraints, namely, for any two scenarios $s_1, s_2 \in \Sigma_P$ and any event $u \in V_{s_1,s_2}^+$, the ES σ satisfies the following constraint, which is denoted by $H_\varepsilon(s_1; s_2; u)$:*

$$[\sigma(s_1)]_u \ge \min\Big(\{[\sigma(s_2)]_u\} \cup \{[\sigma(s_1)]_v + \varepsilon \mid v \in \Delta(s_1; s_2)\}\Big)$$

We say that a CSTN Γ is ε-dynamically-consistent (ε-DC) *if it admits $\sigma \in \mathcal{S}_\Gamma$ which is both viable and ε-dynamic.*

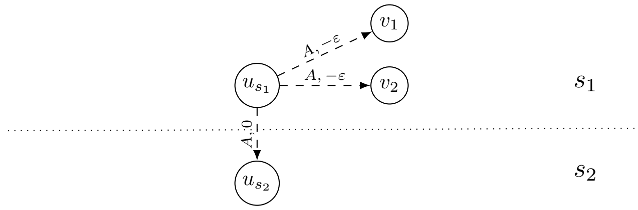

Fig. 1: An $H_\varepsilon(s_1; s_2; u)$ constraint, modeled as a hyperarc.

As shown in [5], ε-DC can be modeled in terms of HyTN-Consistency. Fig. 1 depicts an illustration of an $H_\varepsilon(s_1; s_2; u)$ constraint, modeled as an hyperarc.

Also, in [5] we proved that DC coincides with $\hat{\varepsilon}$-DC, provided that $\hat{\varepsilon} \triangleq |\Sigma_P|^{-1}|V|^{-1}$.

Theorem 2. *Let $\hat{\varepsilon} \triangleq |\Sigma_P|^{-1}|V|^{-1}$. Then, Γ is DC if and only if Γ is $\hat{\varepsilon}$-DC. Moreover, if Γ is ε-DC for some $\varepsilon > 0$, then Γ is ε'-DC for every $\varepsilon' \in (0, \varepsilon]$.*

Then, the main result offered in [5] is a (pseudo) singly-exponential time DC-checking procedure (based on HyTNs).

Theorem 3. *There exists an $O\big(|\Sigma_P|^3|A|^2|V| + |\Sigma_P|^4|A||V|^2|P| + |\Sigma_P|^5|V|^3|P|\big)W$ time algorithm for checking DC on any input CSTN $\Gamma = \langle V, A, L, \mathcal{O}, \mathcal{O}V, P\rangle$. In particular, given any dynamically-consistent CSTN Γ, the algorithm returns a viable and dynamic ES for Γ.*

Here, $W \triangleq \max_{a\in A} |w_a|$.

III. DC with Instantaneous Reaction-Time

Theorem 2 points out the equivalence between ε-DC and DC, that arises for a sufficiently small $\varepsilon > 0$. However, Definition 13 makes sense even if $\varepsilon = 0$, so a natural question is what happens to the above mentioned relationship between DC and ε-DC when $\varepsilon = 0$. In this section we first show that 0-DC doesn't imply DC, and, moreover, that 0-DC is in itself too weak to capture an adequate notion of DC with an instantaneous reaction-time. In light of this we will introduce a stronger notion, which is named *ordered-Dynamic-Consistency (π-DC)*; this will turn out to be a suitable notion of DC with an instantaneous reaction-time.

Example 2 (CSTN $\Gamma_\Box$). *Consider the following CSTN $\Gamma_\Box = (V_\Box, A_\Box, L_\Box, \mathcal{O}_\Box, \mathcal{O}V_\Box, P_\Box)$; see Fig. 2 for an illustration.*

– $V_\Box = \{\bot, \top, A, B, C\}$;

– $A_\Box = \{(\top - \bot \leq 1, \lambda), (\bot - \top \leq -1, \lambda), (\top - A \leq 0, b \wedge \neg c), (\top - B \leq 0, a \wedge c), (\top - C \leq 0, \neg a \wedge \neg b), (\bot - A \leq 0, \lambda), (A - \bot \leq 0, \neg b), (A - \bot \leq 0, c), (\bot - B \leq 0, \lambda), (B - \bot \leq 0, \neg a), (A - \bot \leq 0, \neg c), (\bot - C \leq 0, \lambda), (C - \bot \leq 0, a), (C - \bot \leq 0, b)\}$;

– $L_\Box(A) = L_\Box(B) = L_\Box(C) = L_\Box(\bot) = L_\Box(\top) = \lambda$;

– $\mathcal{O}_\Box(a) = A$, $\mathcal{O}_\Box(b) = B$, $\mathcal{O}_\Box(c) = C$;

– $\mathcal{O}V_\Box = \{A, B, C\}$;

– $P_\Box = \{a, b, c\}$.

Proposition 1. *The CSTN $\Gamma_\Box$ (Example 2, Fig. 2) is 0-DC.*

Proof. Consider the execution strategy $\sigma_\Box : \Sigma_{P_\Box} \to \Psi_{V_\Box}$:

– $[\sigma_\Box(s)]_A \triangleq s(a \wedge b \wedge \neg c) + s(\neg a \wedge b \wedge \neg c)$;

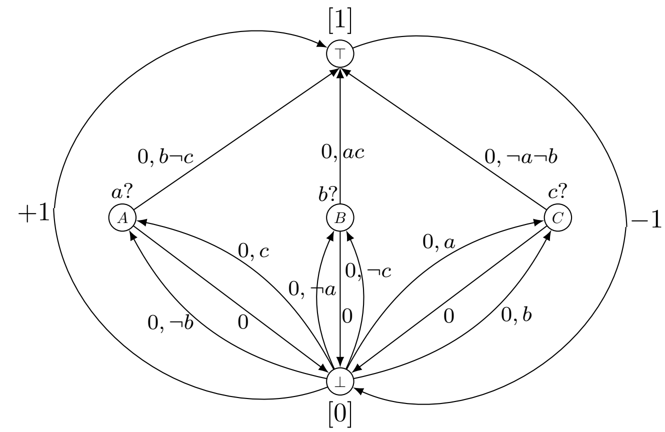

Fig. 2: A CSTN $\Gamma_\Box$ which is 0-DC but not DC.

– $[\sigma_\Box(s)]_B \triangleq s(a \wedge b \wedge c) + s(a \wedge \neg b \wedge c)$;

– $[\sigma_\Box(s)]_C \triangleq s(\neg a \wedge \neg b \wedge \neg c) + s(\neg a \wedge \neg b \wedge c)$;

– $[\sigma_\Box(s)]_\bot \triangleq 0$ and $[\sigma_\Box(s)]_\top \triangleq 1$, for every $s \in \Sigma_P$.

An illustration of $\sigma_\Box$ is offered in Fig 3. Three cubical graphs are depicted in which every node is labelled as v_s for some $(v, s) \in V_\Box \times \Sigma_{P_\Box}$: an edge connects $v_{1 s_1}$ and $v_{2 s_2}$ if and only if: (i) $v_1 = v_2$ and (ii) the Hamming distance between s_1 and s_2 is unitary; each scenario $s \in \Sigma_{P_\Box}$ is represented as $s = \alpha\beta\gamma$ for $\alpha, \beta, \gamma \in \{0, 1\}$, where $s(a) = \alpha$, $s(b) = \beta$, $s(c) = \gamma$; moreover, each node $v_s = (v, s) \in V_\Box \times \Sigma_{P_\Box}$ is filled in black if $[\sigma_\Box(s)]_v = 0$, and in white if $[\sigma_\Box(s)]_v = 1$. So all three 3-cubes own both black and white nodes, but each of them, in its own dimension, decomposes into two identically colored 2-cubes. Fig. 4 offers another visualization of $\sigma_\Box$ in

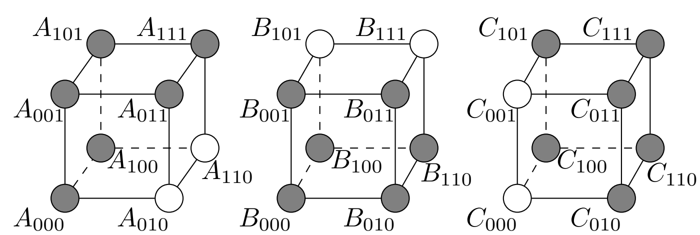

Fig. 3: The ES $\sigma_\Box$ for the CSTN $\Gamma_\Box$.

which every component of the depicted graph corresponds to a restriction STN $\Gamma^+_{\Box s}$ for some $s \in \Sigma_{P_\Box}$, where $s_i, s_j \in \Sigma_{P_\Box}$ are grouped together whenever $\Gamma^+_{\Box s_i} = \Gamma^+_{\Box s_j}$. It is easy to see from Fig. 4 that $\sigma_\Box$ is viable for $\Gamma_\Box$. In order to check that $\sigma_\Box$ is 0-dynamic, look again at Fig. 4, and notice that for every $s_i, s_j \in \Sigma_\Box$, where $s_i \neq s_j$, there exists an event node $X \in \{A, B, C\}$ such that $[\sigma_\Box(s_i)]_X = 0 = [\sigma_\Box(s_j)]_X$ and $s_i(X) \neq s_j(X)$. With this in mind it is easy to check that all of the H_0 constraints are thus satisfied by $\sigma_\Box$. Therefore, the CSTN $\Gamma_\Box$ is 0-DC. $\Box$

Proposition 2. *The CSTN $\Gamma_\Box$ is not DC.*

Proof. Let σ be a viable ES for $\Gamma_\Box$. Then, σ must be the ES $\sigma_\Box$ depicted in Fig. 4, there is no other choice here. Let $\hat{s} \in \Sigma_{P_\Box}$.

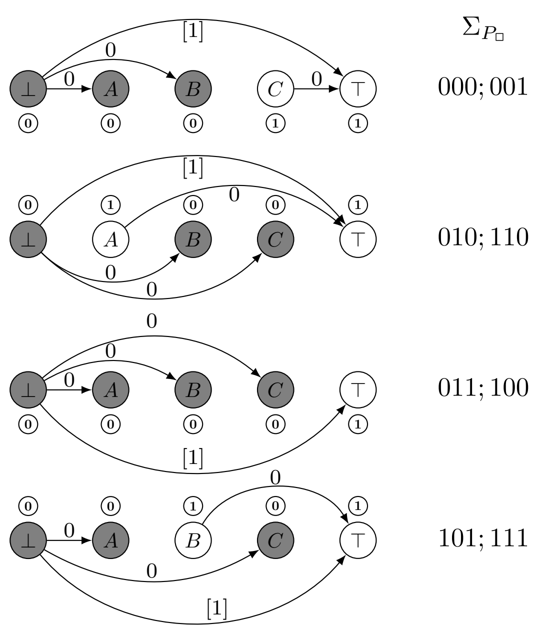

Fig. 4: The restrictions $\Gamma^+_{\Box\,s}$ for $s \in \Sigma_{P_\Box}$, where the execution times $[\sigma_\Box(s)]_v \in \{\mathbf{0}, \mathbf{1}\}$ are depicted in bold face.

Then, it is easy to check from Fig. 4 that: (i) $[\sigma_\Box(\hat{s})]_\bot = 0$, $[\sigma_\Box(\hat{s})]_\top = 1$, and it holds $[\sigma_\Box(\hat{s})]_X \in \{0, 1\}$ for every $X \in \{A, B, C\}$; (ii) there exists at least two observation events $X \in \{A, B, C\}$ such that $[\sigma(\hat{s})]_X = 0$; still, (iii) there is no $X \in \{A, B, C\}$ such that $[\sigma(s)]_X = 0$ for every $s \in \Sigma_{P_\Box}$, i.e., no observation event is executed first at all possible scenarios. Therefore, the ES $\sigma_\Box$ is not dynamic. $\Box$

We now introduce a stronger notion of dynamic consistency; it is named *ordered-Dynamic-Consistency (π-DC)*, and it takes explicitly into account an additional ordering between the observation events scheduled at the same execution time.

Definition 14 (π-Execution-Strategy)**.** *An* ordered-Execution-Strategy (π-ES) *for Γ is a mapping:*

$$\sigma : s \mapsto ([\sigma(s)]^t, [\sigma(s)]^\pi),$$

where $s \in \Sigma_P$, $[\sigma(s)]^t \in \Phi_V$, and finally, $[\sigma(s)]^\pi : \mathcal{OV}^+_s \rightleftharpoons \{1, \ldots, |\mathcal{OV}^+_s|\}$ is bijective. The set of π-ES of Γ is denoted by $\mathcal{S}_\Gamma$. For any $s \in \Sigma_P$, the execution time *of an event $v \in V^+_s$ in the schedule $[\sigma(s)]^t \in \Phi_{V^+_s}$ is denoted by $[\sigma(s)]^t_v \in \mathbf{R}$; the* position *of an observation $\mathcal{O}_p \in \mathcal{OV}^+_s$ in $\sigma(s)$ is $[\sigma(s)]^\pi_{\mathcal{O}_p}$. We require positions to be* coherent *w.r.t. execution times, i.e., $\forall(\mathcal{O}_p, \mathcal{O}_q \in \mathcal{OV}^+_s)$ if $[\sigma(s)]^t_{\mathcal{O}_p} < [\sigma(s)]^t_{\mathcal{O}_q}$ then $[\sigma(s)]^\pi_{\mathcal{O}_p} < [\sigma(s)]^\pi_{\mathcal{O}_q}$. In addition, it is worth to adopt the notation $[\sigma(s)]^\pi_v \triangleq |\mathcal{OV}| + 1$ whenever $v \in V^+_s \setminus \mathcal{OV}$.*

Definition 15 (π-History)**.** *Let $\sigma \in \mathcal{S}_\Gamma$, $s \in \Sigma_P$, and let $\tau \in \mathbf{R}$ and $\psi \in \{1, \ldots, |V|\}$. The* ordered-history *$\pi$-Hst$(\tau, \psi, s, \sigma)$ of τ and ψ in the scenario s, under the π-ES σ, is defined as: $\pi\text{-Hst}(\tau, \psi, s, \sigma) \triangleq \{(p, s(p)) \in P \times \{0, 1\} \mid \mathcal{O}_p \in \mathcal{OV}^+_s, [\sigma(s)]^t_{\mathcal{O}_p} \le \tau, [\sigma(s)]^\pi_{\mathcal{O}_p} < \psi\}$.*

We are finally in the position to define π-DC.

Definition 16 (π-Dynamic-Consistency)**.** *Any $\sigma \in \mathcal{S}_\Gamma$ is called* π-dynamic *when, for any two scenarios $s_1, s_2 \in \Sigma_P$ and any event $v \in V^+_{s_1,s_2}$, if $\tau \triangleq [\sigma(s_1)]^t_v$ and $\psi \triangleq [\sigma(s_1)]^\pi_v$, then:*

$$\text{Con}(\pi\text{-Hst}(\tau, \psi, s_1, \sigma), s_2) \Rightarrow [\sigma(s_2)]^t_v = \tau, [\sigma(s_2)]^\pi_v = \psi.$$

We say that Γ is π-dynamically-consistent (π-DC) *if it admits $\sigma \in \mathcal{S}_\Gamma$ which is both viable and π-dynamic. The problem of checking whether a given CSTN is π-DC is named* π-DC-*Checking.*

Remark 1. *It is easy to see that, due to the strict inequality "$[\sigma(s)]^\pi_{\mathcal{O}_p} < \psi$" in the definition of π-Hst$(\cdot)$ (Definition 15), in a π-dynamic π-ES, there must be exactly one $\mathcal{O}_{p'} \in \mathcal{OV}$, for some $p' \in P$, which is executed at first (w.r.t. both execution time and position) under all possible scenarios $s \in \Sigma_P$.*

Proposition 3. *The CSTN $\Gamma_\Box$ is not π-DC.*

Proof. The proof goes almost in the same way as that of Proposition 2. In particular, no observation event is executed first (i.e., at time $t = 0$ and position $\psi = 1$) in all possible scenarios. Since there is no first-in-time observation event, then, the ES σ is not π-dynamic. $\Box$

We provide next a CSTN which is π-DC but not DC.

Example 3. *Define $\Gamma_\pi = (V_\pi, A_\pi, \mathcal{O}_\pi, \mathcal{OV}_\pi, P_\pi)$ as follows. $V_\pi = \{\mathcal{O}_p, X, \top\}$, $A_\pi = \{(\top - \mathcal{O}_p \le 1, \lambda), (\mathcal{O}_p - \top \le -1, \lambda), (X - \mathcal{O}_p \le 0, p), (\top - X \le 0, \neg p)\}$, $\mathcal{O}_\pi(p) = \mathcal{O}_p$, $\mathcal{OV}_\pi = \{\mathcal{O}_p\}$, $P_\pi = \{p\}$. Fig. 5 depicts the CSTN Γ_π.*

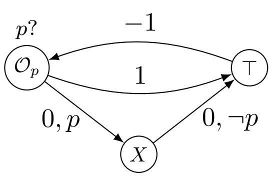

Fig. 5: The CSTN Γ_π.

Proposition 4. *The CSTN Γ_π is π-DC, but it is not DC.*

Proof. Let $s_1, s_2 \in \Sigma_{P_\pi}$ be two scenarios such that $s_1(p) = 1$ and $s_2(p) = 0$. Consider the π-ES σ defined as follows: $[\sigma(s_1)]^t_{\mathcal{O}_p} = [\sigma(s_1)]^t_X = 0$, $[\sigma(s_1)]^t_\top = 1$; and $[\sigma(s_2)]^t_{\mathcal{O}_p} = 0$, $[\sigma(s_2)]^t_X = [\sigma(s_2)]^t_\top = 1$; finally, $[\sigma(s)]^\pi_{\mathcal{O}_p} = 1$, $[\sigma(s)]^\pi_X = [\sigma(s)]^\pi_\top = 2$, for all $s \in \{s_1, s_2\}$. Then, σ is viable and π-dynamic for Γ_π. To see that Γ_π is not DC, pick any $\varepsilon > 0$. Notice that any viable ES must schedule X either at $t = 0$ or $t = 1$, depending on the outcome of $\mathcal{O}_p$, which in turn happens at $t = 0$; however, in any ε-dynamic strategy, the planner can't react to the outcome of $\mathcal{O}_p$ before time $t = \varepsilon > 0$. This implies that Γ_π is not ε-DC. Since ε was chosen arbitrarily ($\varepsilon > 0$), then Γ_π can't be DC by Theorem 2. $\Box$

So Γ_π is ε-DC for $\varepsilon = 0$ but for *no* $\varepsilon > 0$. In summary, the following chain of implications holds on the various DCs:

$$\boxed{[\varepsilon\text{-DC}, \forall \varepsilon \in (0, \hat{\varepsilon}]] \Leftrightarrow \text{DC} \overset{\not\Leftarrow}{\Rightarrow} \pi\text{-DC} \overset{\not\Leftarrow}{\Rightarrow} [\varepsilon\text{-DC, for } \varepsilon = 0]}$$

where $\hat{\varepsilon} \triangleq |\Sigma_P|^{-1} \cdot |V|^{-1}$ as in Theorem 2.

A. The ps-tree: a "skeleton" structure for π-dynamic π-ESs

In this subsection we introduce a labelled tree data structure, named the *ps-tree*, which turns out to capture the "skeleton" ordered structure of π-dynamic π-ESs.

Definition 17 (PS-Tree). *Let P be any set of boolean variables. A permutation-scenario tree (ps-tree) π_T over P is an outward (non-empty) rooted binary tree such that:*

- *Each node u of π_T is labelled with a letter $p_u \in P$;*
- *All the nodes that lie along a path leading from the root to a leaf are labelled with distinct letters from P.*
- *Each arc (u, v) of π_T is labelled by some $b_{(u,v)} \in \{0, 1\}$;*
- *The two arcs (u, v_l) and (u, v_r) exiting a same node u have opposite labels, i.e., $b_{(u,v_l)} \neq b_{(u,v_r)}$.*

Fig. 6 depicts an example of a ps-tree.

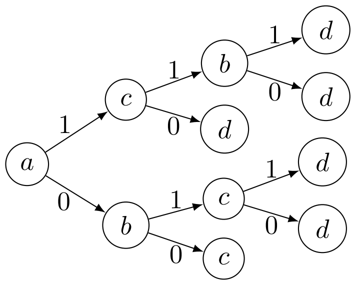

Fig. 6: An example of a ps-tree over $P = \{a, b, c, d\}$.

Definition 18 (π_s, s_i, Coherent-PS-Tree). *Let π_T be a ps-tree over P, let r be the root and s be any leaf. Let $(r, v_2, \ldots, s)$ be the sequence of the nodes encountered along the path going from r down to s in π_T. Then:*

- *The sequence of labels $\pi_s = (p_r, p_{v_2}, \ldots, p_s)$ is a permutation of the subset of letters $\{p_r, p_{v_2}, \ldots, p_s\} \subseteq P$.*
- *Each sequence of bits $(b_{(r,v_2)}, \ldots, b_{(v_i,v_{i+1})})$, for each $i \in \{1, 2, \ldots, k_s - 1\}$ (where $v_1 \triangleq r$ and $v_{k_s} \triangleq s$), can be seen as a partial scenario s_i over P; i.e., define $s_i(v_j) \triangleq b_{(v_j,v_{j+1})}$, for every $j \in \{1, \ldots, i\}$.*
- *π_T is* coherent (c-ps-tree) *with Γ if, for every leaf s of π_T, $\{\mathcal{O}_{p_r}, \mathcal{O}_{p_{v_2}}, \ldots, \mathcal{O}_{p_s}\} = \mathcal{O}V^+_{s'}$ holds for every complete scenario $s' \in \Sigma_P$ such that $\mathrm{Sub}(s', s_{k_s-1})$.*

It is not difficult to see that a π-dynamic π-ES induces one and only one c-ps-tree π_T. So, the existence of a suitable c-ps-tree is a necessary condition for a π-ES to be π-dynamic. One may ask whether a π-dynamic π-ES can be reconstructed from its c-ps-tree; the following subsection answers affirmatively.

B. Verifying a c-ps-tree: on π-DC and HyTN-Consistency.

This subsection builds on the notion of c-ps-tree to work out the details of the relationship between π-DC and HyTN-Consistency. Once this picture is in place, it will be easy to reduce to HyTN-Consistency the problem of deciding whether a given CSTN admits a valid π-dynamic π-ES with a given c-ps-tree. This easy result already provides a first combinatorial algorithm for π-DC, though of doubly exponential complexity in $|P|$; a bound to be improved in later subsections, but that can help sizing the sheer dimensionality and depth of the problem.

Firstly, the notion of *Expansion* of CSTNs is recalled [5].

Definition 19 (Expansion $\langle V^{\mathrm{Ex}}_\Gamma, \Lambda^{\mathrm{Ex}}_\Gamma \rangle$). *Consider a CSTN $\Gamma = (V, A, L, \mathcal{O}, \mathcal{O}V, P)$. Consider the family of all (distinct) STNs (V_s, A_s), one for each scenario $s \in \Sigma_P$, defined as follows:*

$$V_s \triangleq \{v_s \mid v \in V^+_s\} \text{ and } A_s \triangleq \{(u_s, v_s, w) \mid (u, v, w) \in A^+_s\}.$$

The expansion *$\langle V^{Ex}_\Gamma, \Lambda^{Ex}_\Gamma \rangle$ of the CSTN Γ is defined as follows:*

$$\langle V^{Ex}_\Gamma, \Lambda^{Ex}_\Gamma \rangle \triangleq \Big(\bigcup_{s \in \Sigma_P} V_s, \bigcup_{s \in \Sigma_P} A_s \Big).$$

Notice, $(V^{\mathrm{Ex}}_\Gamma, \Lambda^{\mathrm{Ex}}_\Gamma)$ is an STN with at most $|V^{\mathrm{Ex}}_\Gamma| \leq |\Sigma_P| \cdot |V|$ nodes and at most $|\Lambda^{\mathrm{Ex}}_\Gamma| \leq |\Sigma_P| \cdot |A|$ standard arcs.

We now show that the expansion of a CSTN can be enriched with some standard arcs and some hyperarcs in order to model the π-DC property, by means of an HyTN denoted $\mathcal{H}^{\pi_T}_0(\Gamma)$.

Definition 20 (HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$). *Let $\Gamma = (V, A, L, \mathcal{O}, \mathcal{O}V, P)$ be a given CSTN. Let π_T be a given c-ps-tree over P.*

Then, the HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$ is defined as follows:

- *For every scenarios $s_1, s_2 \in \Sigma_P$ and $u \in V^+_{s_1,s_2} \setminus \mathcal{O}V$, define a hyperarc $\alpha = \alpha_0(s_1; s_2; u)$ as follows (with the intention to model $H_0(s_1; s_2; u)$, see Definition 13):*

$$\alpha = \alpha_0(s_1; s_2; u) \triangleq \langle t_\alpha, H_\alpha, w_\alpha \rangle,$$

  where:

  - *$t_\alpha \triangleq u_{s_1}$ is the tail of the hyperarc α;*
  - *$H_\alpha \triangleq \{u_{s_2}\} \cup \Delta(s_1; s_2)$ is the set of the heads;*
  - *$w_\alpha(u_{s_2}) \triangleq 0$; $\forall (v \in \Delta(s_1; s_2))\ w_\alpha(v) \triangleq 0$.*

  Now, consider the expansion of the CSTN Γ $\langle V^{Ex}_\Gamma, \Lambda^{Ex}_\Gamma \rangle = \big(\bigcup_{s \in \Sigma_P} V_s, \bigcup_{s \in \Sigma_P} A_s \big)$ (as in Definition 19). Then:

- *For each internal node x of π_T, A'_x is a set of (additional) standard arcs defined as follows. Let $\pi_x = (r, \ldots, x')$ be the sequence of all and only the nodes along the path going from the root r to the parent x' of x in π_T (where we can assume $r' = r$). Let $P_x \subseteq P$ be the corresponding literals, p_x excluded, i.e., $P_x \triangleq \{p_z \in P \mid z \text{ appears in } \pi_x \text{ and } p_z \text{ is the label of } z \text{ in } \pi_T\} \setminus \{p_x\}$. Let s_x be the partial scenario defined as follows:*

$$s_x : P_x \to \{0, 1\} : \begin{cases} \lambda\,, & \text{if } x = r; \\ p_z \mapsto b_{(z,z')}, & \text{if } x \neq r. \end{cases}$$

  where z' is the unique child of z in π_T lying on π_x. Let x_0 (x_1) be the unique child of x in π_T such that $b_{x,x_0} = 0$ ($b_{x,x_1} = 1$). For every complete $s'_x \in \Sigma_P$ such that $\mathrm{Sub}(s'_x, s_x)$, we define:

$$B'_{s'_x} \triangleq \begin{cases} \{\langle (\mathcal{O}_{p_{x_0}})_{s'_x}, (\mathcal{O}_{p_x})_{s'_x}, 0 \rangle\}, & \text{if } s'_x(x) = 0; \\ \{\langle (\mathcal{O}_{p_{x_1}})_{s'_x}, (\mathcal{O}_{p_x})_{s'_x}, 0 \rangle\}, & \text{if } s'_x(x) = 1. \end{cases}$$

  Also, for every complete $s'_x, s''_x \in \Sigma_P$ such that $\mathrm{Sub}(s'_x, s_x)$ and $\mathrm{Sub}(s''_x, s_x)$, where $s'_x \neq s''_x$, we define: $C'_{s'_x,s''_x} \triangleq \{\langle (\mathcal{O}_{p_x})_{s'_x}, (\mathcal{O}_{p_x})_{s''_x}, 0 \rangle\}$.

Finally,

$$A'_x \triangleq \bigcup_{s'_x \in \Sigma_P \,:\, \mathsf{Sub}(s'_x, s_x)} B'_{s'_x} \cup \bigcup_{\substack{s'_x, s''_x \in \Sigma_P \,:\, s'_x \neq s''_x, \\ \mathsf{Sub}(s'_x, s_x), \mathsf{Sub}(s''_x, s_x)}} C'_{s'_x, s''_x}.$$

- *Then, $\mathcal{H}^{\pi}_0(\Gamma)$ is defined as $\mathcal{H}^{\pi}_0(\Gamma) \triangleq \langle V^{Ex}_{\Gamma}, \mathcal{A}_{\mathcal{H}^{\pi}_0(\Gamma)} \rangle$, where,*

$$\mathcal{A}_{\mathcal{H}^{\pi}_0(\Gamma)} \triangleq \Lambda^{Ex}_{\Gamma} \cup \bigcup_{\substack{s_1, s_2 \in \Sigma_P \\ u \in V^+_{s_1, s_2}}} \alpha_{\varepsilon}(s_1; s_2; u) \cup \bigcup_{\substack{x \,:\, \textit{internal} \\ \textit{node of } \pi_T}} A'_x.$$

Notice that the following holds: each $\alpha_{\varepsilon}(s_1; s_2; u)$ has size $|\alpha_{\varepsilon}(s_1; s_2; u)| = \Delta(s_1; s_2) + 1 \leq |P| + 1$.

The following theorem establishes the connection between the π-DC of CSTNs and the consistency of HyTNs.

Theorem 4. *Given any CSTN $\Gamma = \langle V, A, L, \mathcal{O}, \mathcal{OV}, P \rangle$, it holds that the CSTN Γ is π-DC if and only if there exists a c-ps-tree π_T such that the HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$ is consistent. Moreover, $\mathcal{H}^{\pi_T}_0(\Gamma)$ has at most $|V_{\mathcal{H}^{\pi_T}_0(\Gamma)}| \leq |\Sigma_P|\,|V|$ nodes, $|\mathcal{A}_{\mathcal{H}^{\pi_T}_0(\Gamma)}| = O(|\Sigma_P|\,|A| + |\Sigma_P|^2|V|)$ hyperarcs, and it has size at most $m_{\mathcal{A}_{\mathcal{H}^{\pi_T}_0(\Gamma)}} = O(|\Sigma_P|\,|A| + |\Sigma_P|^2|V|\,|P|)$.*

Proof. (1) Firstly, we prove that the CSTN Γ is π-DC if and only if there exists a c-ps-tree π_T such that the HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$ is consistent.

($\Rightarrow$) Let $\sigma \in \mathcal{S}_{\Gamma}$ be a given viable and π-dynamic execution strategy for the CSTN Γ. Since σ is π-dynamic, then for any two $s_1, s_2 \in \Sigma_P$ and any $v \in V^+_{s_1, s_2}$ the following holds on the execution time $\tau \triangleq [\sigma(s_1)]^t_v$ and position $\psi \triangleq [\sigma(s_1)]^{\pi}_v$:

$$\textit{Con}(\pi\textit{-Hst}(\tau, \psi, s_1, \sigma), s_2) \Rightarrow [\sigma(s_2)]^t_v = \tau, [\sigma(s_2)]^{\pi}_v = \psi.$$

It is easy to see that this induces one and only one c-ps-tree π_T: indeed, due to Remark 1, there must be exactly one $\mathcal{O}_{p'} \in \mathcal{OV}$, for some $p' \in P$, which is executed at first (w.r.t. to both execution time and position) under all possible scenarios; then, depending on the boolean result of p', a second observation p'' can be differentiated, and it can occur at the same or at a subsequent time instant, but still at a subsequent position; again, by Remark 1, there is exactly one $\mathcal{O}_{p''} \in \mathcal{OV}$ which comes first under all possible scenarios that agree on p'; and so on and so forth, thus forming a tree structure over P, rooted at p', which is captured exactly by our notion of c-ps-tree. Then, let $\phi_{\sigma} : V^{\text{Ex}}_{\Gamma} \to \mathbf{R}$ be the schedule of $\mathcal{H}^{\pi_T}_0(\Gamma)$ defined as: $\phi_{\sigma}(v_s) \triangleq [\sigma(s)]^t_v$ for every $v_s \in V^{\text{Ex}}_{\Gamma}$, where $s \in \Sigma_P$ and $v \in V^+_s$. It is not difficult to check from the definitions, at this point, that all of the standard arc and hyperarc constraints of $\mathcal{H}^{\pi_T}_0(\Gamma)$ are satisfied by ϕ_{σ}, that is to say that ϕ_{σ} must be feasible for $\mathcal{H}^{\pi_T}_0(\Gamma)$. Hence, $\mathcal{H}^{\pi_T}_0(\Gamma)$ is consistent.

($\Leftarrow$) Assume that there exists a c-ps-tree π_T such that the HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$ is consistent, and let $\phi : V^{\text{Ex}}_{\Gamma} \to \mathbf{R}$ be a feasible schedule for $\mathcal{H}^{\pi_T}_0(\Gamma)$. Then, let $\sigma_{\phi, \pi_T}(s) \in \mathcal{S}_{\Gamma}$ be the execution strategy defined as follows:

- $[\sigma_{\phi, \pi_T}(s)]^t_v \triangleq \phi(v_s)$, $\forall\ v_s \in V^{\text{Ex}}_{\Gamma}$, $s \in \Sigma_P$, $v \in V^+_s$;
- Let $s' \in \Sigma_P$ be any complete scenario. Then, s' induces exactly one path in π_T, in a natural way, i.e., by going from the root r down to some leaf s. Notice that the sequence of labels $(p_r, p_{v_2}, \ldots, p_s)$ can be seen as a bijection, i.e., $\pi_s : \mathcal{OV}^+_{s'} \rightleftharpoons \{1, \ldots, |\mathcal{OV}^+_{s'}|\}$. Then, for any $s' \in \Sigma_P$ and $v \in \mathcal{OV}^+_{s'}$, define $[\sigma_{\phi, \pi_T}(s')]^{\pi}_v \triangleq \pi_s(v)$.

It is not difficult to check from the definitions, at this point, that since ϕ is feasible for $\mathcal{H}^{\pi_T}_0(\Gamma)$, then σ_{ϕ, π_T} must be viable and π-dynamic for the CSTN Γ. Hence, the CSTN Γ is π-DC.

(2) The size bounds for $\mathcal{H}^{\pi_T}_0(\Gamma)$ follow from Definition 20. □

In Fig. 7, Algorithm 1 presents the pseudocode for constructing the HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$, as prescribed by Definition 20.

Algorithm 1: `construct_`$\mathcal{H}(\Gamma, \pi_T)$

Input: a CSTN $\Gamma \triangleq \langle V, A, L, \mathcal{O}, \mathcal{OV}, P \rangle$, a c-ps-tree π_T coherent with Γ.

foreach *($s \in \Sigma_P$)* **do**
    $V_s \leftarrow \{v_s \mid v \in V^+_s\}$;
    $A_s \leftarrow \{a_s \mid a \in A^+_s\}$;
$V^{\text{Ex}}_{\Gamma} \leftarrow \cup_{s \in \Sigma_P} V_s$;
$\Lambda^{\text{Ex}}_{\Gamma} \leftarrow \cup_{s \in \Sigma_P} A_s$;
foreach *($s_1, s_2 \in \Sigma_P,\ s_1 \neq s_2$)* **do**
    foreach *($u \in V^+_{s_1, s_2} \setminus \mathcal{OV}$)* **do**
        $t_{\alpha} \leftarrow u_{s_1}$;
        $H_{\alpha} \leftarrow \{u_{s_2}\} \cup (\Delta(s_1; s_2))$;
        $w_{\alpha}(u_{s_2}) \leftarrow 0$;
        foreach $v \in \Delta(s_1; s_2)$ **do**
            $w_{\alpha}(v_{s_1}) \leftarrow 0$;
        $\alpha_0(s_1; s_2; u) \leftarrow \langle t_{\alpha}, H_{\alpha}, w_{\alpha} \rangle$;
foreach *(x : internal node of π_T)* **do**
    $A'_x \leftarrow$ as defined in Definition 20;
$\mathcal{A}_{\mathcal{H}^{\pi_T}_0(\Gamma)} \leftarrow \Lambda^{\text{Ex}}_{\Gamma} \cup \bigcup_{\substack{s_1, s_2 \in \Sigma_P \\ u \in V^+_{s_1, s_2}}} \alpha_0(s_1; s_2; u) \cup \bigcup_{\substack{x \,:\, \text{internal} \\ \text{node of } \pi_T}} A'_x$;
$\mathcal{H}^{\pi_T}_0(\Gamma) \leftarrow \langle V^{\text{Ex}}_{\Gamma}, \mathcal{A}_{H^{\pi_T}_0(\Gamma)} \rangle$;
return $\mathcal{H}^{\pi_T}_0(\Gamma)$;

If Γ is π-DC, there is an integral π-dynamic π-ES, as below.

Proposition 5. *Assume $\Gamma = \langle V, A, L, \mathcal{O}, \mathcal{OV}, P \rangle$ to be π-DC. Then, there is some π-ES $\sigma \in \mathcal{S}_{\Gamma}$ which is viable, π-dynamic, and* integral*, namely, for every $s \in \Sigma_P$ and every $v \in V^+_s$, the following integrality property holds:*

$$[\sigma(s)]^t_v \in \{0, 1, 2, \ldots, \mathcal{M}_{\Gamma}\} \subseteq \mathbf{N},$$

where $\mathcal{M}_{\Gamma} \triangleq \big(|\Sigma_P||V| + |\Sigma_P||A| + |\Sigma_P|^2|V|\big)W$.

Proof. By Theorem 4, since Γ is π-DC, there exists some c-ps-tree π_T such that the HyTN $\mathcal{H}^{\pi_T}_0(\Gamma)$ is consistent; moreover, by Theorem 4 again, $\mathcal{H}^{\pi_T}_0(\Gamma)$ has $|V_{\mathcal{H}^{\pi_T}_0(\Gamma)}| \leq |\Sigma_P|\,|V|$ nodes and $|\mathcal{A}_{\mathcal{H}^{\pi_T}_0(\Gamma)}| \leq |\Sigma_P|\,|A| + |\Sigma_P|^2|V|$ hyperarcs. Since $\mathcal{H}^{\pi_T}_0(\Gamma)$ is consistent, it follows from Theorem 4 [also see Lemma 1 and Theorem 8 in [4]] that $\mathcal{H}^{\pi_T}_0(\Gamma)$ admits an integral and feasible schedule ϕ such that:

$$\phi : V_{\mathcal{H}^{\pi_T}_0(\Gamma)} \to \{0, 1, 2, \ldots, \mathcal{M}_{\Gamma}\},$$

where $\mathcal{M}_\Gamma \leq (|V_{\mathcal{H}_0^{\pi_T}(\Gamma)}| + |\mathcal{A}_{\mathcal{H}_0^{\pi_T}(\Gamma)}|)W$. Therefore, it holds that $\mathcal{M}_\Gamma \leq (|\Sigma_P|\,|V| + |\Sigma_P|\,|A| + |\Sigma_P|^2|V|)W$. ☐

Given a CSTN Γ and some c-ps-tree π_T, it is thus easy to check whether there exists some π-ES for Γ whose ordering relations are exactly the same as those prescribed by π_T. Indeed, it is sufficient to construct $\mathcal{H}_0^{\pi_T}(\Gamma)$ with Algorithm 1, then checking the consistency of $\mathcal{H}_0^{\pi_T}(\Gamma)$ with the algorithm mentioned in Theorem 1. This results into Algorithm 2. The corresponding time complexity is also that of Theorem 1.

Algorithm 2: `check_`π`-DC_on_c-ps-tree`(Γ, π_T)

Input: a CSTN $\Gamma \triangleq \langle V, A, L, \mathcal{O}, \mathcal{O}V, P\rangle$, a c-ps-tree π_T coherent with Γ.

$\mathcal{H}_0^{\pi_T}(\Gamma) \leftarrow$ `construct_`$\mathcal{H}(\Gamma, \pi_T)$; // ref. Algorithm 1
$\phi \leftarrow$ `check_HyTN-Consistency`$(\mathcal{H}_0^{\pi_T}(\Gamma))$; // ref. Thm 1
if *(ϕ is a feasible schedule of $\mathcal{H}_0^{\pi_T}(\Gamma)$)* **then**
  return $\langle \mathit{YES}, \phi, \pi_T\rangle$;
return *NO*;

Fig. 7: Checking π-DC given a c-ps-tree π_T, by reduction to HyTN-Consistency.

Notice that, in principle, one could generate all of the possible c-ps-trees π_T given P, one by one, meanwhile checking for the consistency state of $\mathcal{H}_0^{\pi_T}(\Gamma)$ with Algorithm 2. However, it is not difficult to see that, in general, the total number $f_{|P|}$ of possible c-ps-trees over P is not singly-exponential in $|P|$. Indeed, a moment's reflection revelas that for every $n > 1$ it holds that $f_n = n \cdot f_{n-1}^2$, and $f_1 = 1$. So, any algorithm based on the exhaustive exploration of the whole space comprising all of the possible c-ps-trees over P would not have a (pseudo) singly-exponential time complexity in $|P|$. Nevertheless, we have identified another solution, that allows us to provide a sound-and-complete (pseudo) singly-exponential time π-DC-Checking procedure: it is a simple and self-contained reduction from π-DC-Checking to DC-Checking. This allows us to provide the first sound-and-complete (pseudo) singly-exponential time π-DC-Checking algorithm which employs our previous DC-Checking algorithm (i.e., that underlying Theorem 3) in a direct manner, as a black box, thus avoiding a more fundamental restructuring of it.

C. A Singly-Exponential Time π-DC-Checking Algorithm

This section presents a sound-and-complete (pseudo) singly-exponential time algorithm for solving π-DC, also producing a viable and π-dynamic π-ES whenever the input CSTN is really π-DC. The main result of this section goes as follows.

Theorem 5. *There exists an algorithm for checking π-DC on any input given CSTN $\Gamma = (V, A, L, \mathcal{O}, \mathcal{O}V, P)$ with the following (pseudo) singly-exponential time complexity:*

$$O\Big(|\Sigma_P|^4|A|^2|V|^3 + |\Sigma_P|^5|A||V|^4|P| + |\Sigma_P|^6|V|^5|P|\Big)W.$$

Moreover, when Γ is π-DC, the algorithm also returns a viable and π-dynamic π-ES for Γ. Here, $W \triangleq \max_{a\in A}|w_a|$.

The algorithm mentioned in Theorem 5 consits of a simple reduction from π-DC to (classical) DC in CSTNs.

Basically, the idea is to give a small margin γ so that the planner can actually do before, in the sense of the time value $[\sigma(s)]_v$, what he did "before" in the ordering π. Given any ES in the relaxed network, the planner would then turn it into a π-ES for the original network (which has some more stringent constraints), by rounding-down each time value $[\sigma(s)]_v$ to the largest integer less than or equal to it, i.e., $\big\lfloor[\sigma(s)]_v\big\rfloor$. The problem is that one may (possibly) violate some constraints when there is a "leap" in the rounding (i.e., a difference of one unit, in the rounded value, w.r.t. what one would have wanted). Anyhow, we have identified a technique that allows us to get around this subtle case, provided that γ is exponentially small.

Definition 21. *Relaxed CSTN Γ'. Let $\Gamma = \langle V, A, L, \mathcal{O}, \mathcal{O}V, P\rangle$ be any CSTN with integer constraints. Let $\gamma \in (0, 1)$ be a real. Define $\Gamma'_\gamma \triangleq \langle V, A'_\gamma, L, \mathcal{O}, \mathcal{O}V, P\rangle$ to be a CSTN that differs from Γ only in the numbers appearing in the constraints. Specifically, each constraint $\langle u - v \leq \delta, \ell\rangle \in A$ is replaced in Γ'_γ by a slightly relaxed constraint, $\langle u - v \leq \delta'_\gamma, \ell\rangle \in A'_\gamma$, where:*

$$\delta'_\gamma \triangleq \delta + |V| \cdot \gamma.$$

The following two lemmata hold for any CSTN Γ.

Lemma 1. *Let γ be any real in $(0, |V|^{-1})$.*
If Γ is π-DC, then Γ'_γ is DC.

Proof. Since Γ is π-DC, by Proposition 5, there exists an integral, viable and π-dynamic, π-ES σ for Γ. Let us fix some real $\gamma \in (0, |V|^{-1})$. Define the ES $\sigma'_\gamma \in \mathcal{S}_{\Gamma'_\gamma}$ as follows, for every $s \in \Sigma_P$ and $v \in V_s^+$:

$$[\sigma'_\gamma(s)]_v \triangleq [\sigma(s)]^t_v + [\sigma(s)]^\pi_v \cdot \gamma.$$

Since $[\sigma(s)]^\pi_v \leq |V|$, then:

$$[\sigma(s)]^\pi_v \cdot \gamma < |V| \cdot |V|^{-1} = 1,$$

and so the total ordering of the values $[\sigma'_\gamma(s)]_v$, for a given $s \in \Sigma_P$, coincides with $[\sigma(s)]^\pi$. Hence, the fact that σ'_γ is dynamic follows directly from the π-dynamicity of σ. Moreover, no LTC $(u - v \leq \delta'_\gamma, \ell)$ of Γ'_γ is violated in any scenario $s \in \Sigma_P$ since, if $\Delta'_{\gamma,u,v} \triangleq [\sigma'_\gamma(s)]_u - [\sigma'_\gamma(s)]_v$ then:

$$\begin{aligned}\Delta'_{\gamma,u,v} &= \big([\sigma(s)]^t_u + [\sigma(s)]^\pi_u \cdot \gamma\big) - \big([\sigma(s)]^t_v + [\sigma(s)]^\pi_v \cdot \gamma\big)\\ &\leq [\sigma(s)]^t_u - [\sigma(s)]^t_v + |V| \cdot \gamma\\ &\leq \delta + |V| \cdot \gamma = \delta'.\end{aligned}$$

So, σ'_γ is viable. Since σ'_γ is also dynamic, then Γ'_γ is DC. ☐

The next lemma shows that the converse direction holds as well, but for (exponentially) smaller values of γ.

Lemma 2. *Let γ be any real in $(0, |\Sigma_P|^{-1} \cdot |V|^{-2})$.*
If Γ'_γ is DC, then Γ is π-DC.

Proof. Let $\sigma'_\gamma \in \mathcal{S}_{\Gamma'_\gamma}$ be some viable and dynamic ES for Γ'_γ.

Firstly, we aim at showing that, w.l.o.g., the following lower bound holds:

$$[\sigma'_\gamma(s)]_v - \big\lfloor[\sigma'_\gamma(s)]_v\big\rfloor \geq |V|\cdot\gamma, \text{ for } \textit{all } s \in \Sigma_P \text{ and } v \in V_s^+. \text{ (LB)}$$

This will allow us to simplify the rest of the proof. In order to prove it, let us pick any $\eta \in [0, 1)$ such that:

$$[\sigma'_\gamma(s)]_v - \eta - k \in [0, |V|\cdot\gamma), \quad \text{for } \textit{no } v \in V,\ s \in \Sigma_P,\ k \in \mathbf{Z}.$$

Observe that such a value η exists. Indeed, there are only $|\Sigma_P| \cdot |V|$ choices of pairs $(s, v) \in \Sigma_P \times V$ and each pair rules out a (circular) semi-open interval of length $|V| \cdot \gamma$ in $[0, 1)$, so the total measure of invalid values for η in the semi-open real interval $[0, 1)$ is at most $|\Sigma_P| \cdot |V| \cdot |V| \cdot \gamma < 1$. So η exists.

See Fig. 8 for an intuitive illustration of this fact.

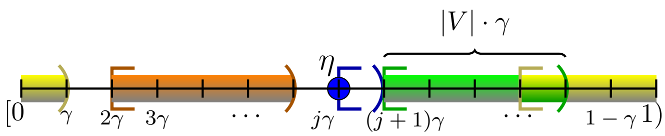

Fig. 8: An illustration of the proof of Lemma 2.

By subtracting η to all time values $\{[\sigma'_\gamma(s)]_v\}_{v\in V, s\in\Sigma_P}$ we can assume w.l.o.g. that $\eta = 0$ holds for the rest of this proof; and thus, that (LB) holds. Now, define $[\sigma(s)]^t_v \triangleq \big\lfloor[\sigma'_\gamma(s)]_v\big\rfloor$, and let $[\sigma(s)]^\pi$ be the ordering induced by $\sigma'_\gamma(s)$. Observe that σ is a well-defined π-ES (i.e., that $[\sigma(s)]^\pi$ is coherent w.r.t. $[\sigma(s)]^t$), thanks to the fact that $\lfloor\cdot\rfloor$ is a monotone operator. Since the ordering $[\sigma(s)]^\pi$ is the same as that of $\sigma'_\gamma(s)$, then σ is π-dynamic.

It remains to prove that σ is viable. For this, take any constraint $(u - v \leq \delta, \ell) \in A$ in Γ, and suppose that:

$$[\sigma'_\gamma(s)]_u - [\sigma'_\gamma(s)]_v \leq \delta'_\gamma = \delta + |V| \cdot \gamma. \quad \text{(A)}$$

If $[\sigma'_\gamma(s)]_u - [\sigma'_\gamma(s)]_v \leq \delta$, then clearly $[\sigma(s)]^t_u - [\sigma(s)]^t_v \leq \delta$. So, the interesting case that we really need to check is when:

$$0 < [\sigma'_\gamma(s)]_u - [\sigma'_\gamma(s)]_v - \delta \leq |V| \cdot \gamma.$$

For this, we observe that the following $(*)$ holds by (LB):

$$\big\lfloor[\sigma'_\gamma(s)]_u\big\rfloor \leq [\sigma'_\gamma(s)]_u - |V| \cdot \gamma. \quad (*)$$

Also, it is clear that:

$$\big\lfloor[\sigma'_\gamma(s)]_v\big\rfloor > [\sigma'_\gamma(s)]_v - 1. \qquad (**)$$

Then,

$$\begin{aligned}
[\sigma(s)]^t_u - [\sigma(s)]^t_v &= \big\lfloor[\sigma'_\gamma(s)]_u\big\rfloor - \big\lfloor[\sigma'_\gamma(s)]_v\big\rfloor \\
&< ([\sigma'_\gamma(s)]_u - |V| \cdot \gamma) - ([\sigma'_\gamma(s)]_v - 1) \\
&\leq ([\sigma'_\gamma(s)]_u - [\sigma'_\gamma(s)]_v) - |V| \cdot \gamma + 1 \\
&\leq \delta'_\gamma - |V| \cdot \gamma + 1 \\
&\leq \delta + 1,
\end{aligned}$$

where: the first equality holds by definition of $[\sigma(s)]^t_x$, for $x \in \{u, v\}$; the strict inequality holds by $(*)$ and $(**)$; the subsequent inequality is just a rewriting; finally, the last two inequalities hold by (A) and by $\delta'_\gamma = \delta + |V| \cdot \gamma$ (respectively). Now, since we have the strict inequality $[\sigma(s)]^t_u - [\sigma(s)]^t_v < \delta + 1$, and since $[\sigma(s)]^t_u - [\sigma(s)]^t_v \in \mathbf{Z}$, then $[\sigma(s)]^t_u - [\sigma(s)]^t_v \leq \delta$, as desired. So, σ is viable. Since σ is both viable and π-dynamic, then Γ is π-DC. $\square$

Fig. 8 illustrates the proof of Lemma 2, in which a family of (circular) semi-open intervals of length $|V| \cdot \gamma$ are depicted as shaded rectangles. Lemma 2 ensures that at least one chunk on length $l_\gamma \geq 1 - |\Sigma_P| \cdot |V|^2 \cdot \gamma$ is not covered by the union of those (circular) semi-open intervals, and it is therefore free to host η; in Fig. 8, this is represented by the blue interval, and $\eta = j \cdot \gamma$ for some $j \in [0, \gamma^{-1})$. Also notice that γ can be fixed as follows:

$$\gamma \triangleq \frac{1}{|\Sigma_P| \cdot |V|^2 + 1};$$

then, $l_\gamma \geq |\Sigma_P|^{-1} \cdot |V|^{-2}$.

In summary, Lemma 1 and Lemma 2 imply Theorem 6.

Theorem 6. *Let Γ be a CSTN and let $\gamma \in (0, |\Sigma_P|^{-1} \cdot |V|^{-2})$. Then, Γ is π-DC if and only if Γ'_γ is DC.*

This allows us to design a simple algorithm for solving π-DC-Checking, by reduction to DC-Checking, which is named `Check-π-DC()` (Algorithm 3). Its pseudocode follows below.

Algorithm 3: `Check-π-DC(Γ)`

```
    Input: a CSTN Γ ≜ ⟨V, A, L, O, OV, P⟩
1   γ ← 1/(|Σ_P|·|V|²+1);
2   A'_γ ← {⟨u − v ≤ δ + |V|·γ, ℓ⟩ | ⟨u − v ≤ δ, ℓ⟩ ∈ A};
3   Γ'_γ ← ⟨V, A'_γ, L, O, OV, P⟩;
4   σ'_γ ← check_DC(Γ'_γ); // see Theorem 3
5   if σ'_γ is a viable and dynamic ES for Γ'_γ then
6       η ← pick η ∈ [0, 1) as in the proof of Lemma 2;
7       foreach (s, v) ∈ Σ_P × V_s^+ do
8           [σ'_γ(s)]_v ← [σ'_γ(s)]_v − η; // shift by η;
9       let σ ∈ Σ_Γ be constructed as follows;
10      foreach s ∈ Σ_P do
11          foreach v ∈ V_s^+ do
12              [σ(s)]^t_v ← ⌊[σ'_γ(s)]_v⌋;
13          [σ(s)]^π ← the ordering on P induced by σ'_γ(s);
14      return ⟨YES, σ⟩;
15  return NO;
```

Fig. 9: Checking π-DC by reduction to DC-Checking.

Description of Algorithm 3: It takes in input a CSTN Γ. When Γ is π-DC, it aims at returning $\langle \texttt{YES}, \sigma \rangle$, where $\sigma \in \mathcal{S}_\Gamma$ is a viable and π-dynamic π-ES for Γ. Otherwise, if Γ is not π-DC, then `Check-π-DC()` (Algorithm 3) returns `NO`. Of course the algorithm implements the reduction described in Definition 21, whereas the π-ES is computed as prescribed by Lemma 2. At line 1, we set $\gamma \leftarrow \frac{1}{|\Sigma_P|\cdot|V|^2+1}$. Then, at lines 2-3, Γ'_γ is constructed as in Definition 21, i.e., $\Gamma'_\gamma \leftarrow \langle V, A'_\gamma, L, \mathcal{O}, \mathcal{OV}, P \rangle$, where $A'_\gamma \leftarrow \big\{\langle u - v \leq \delta + |V| \cdot \gamma, \ell\rangle \mid$

$\langle u-v \leq \delta, \ell\rangle \in A\}$. At this point, at line 5, the DC-Checking algorithm of Theorem 3 is invoked on input Γ'_γ. Let σ'_γ be its output. If Γ'_γ is not DC, then Check-π-DC() (Algorithm 3) returns NO at line 15. When σ'_γ is a viable and dynamic ES for Γ'_γ at line 5, then Check-π-DC() (Algorithm 3) proceeds as follows. At line 6, some $\eta \in [0, 1)$ is computed as in the proof of Lemma 2, i.e., such that $[\sigma'_\gamma(s)]_v - \eta - k \in [0, |V| \cdot \gamma)$ holds for *no* $v \in V, s \in \Sigma_P, k \in \mathbf{Z}$. Notice that it is easy to find such η in practice. Indeed, one may view the real semi-open interval $[0, 1)$ as if it was partitioned into chunks (i.e., smaller semi-open intervals) of length γ; as observed in the proof of Lemma 2, there are only $|\Sigma_P| \cdot |V|$ choices of pairs $(s, v) \in \Sigma_P \times V$, and each pair rules out a (circular) semi-open interval of length $|V| \cdot \gamma$; therefore, there is at least one chunk of length $l_\gamma \geq |\Sigma_P|^{-1} \cdot |V|^{-2}$, within $[0, 1)$, where η can be placed, and we can easily find it just by inspecting (exhaustively) the pairs $(s, v) \in \Sigma_P \times V$. In fact, the algorithm underlying Theorem 3 always deliver an earliest-ES (i.e., one in which the time values are the smallest possible, in the space of all consistent ESs), so that for each interval of length $|V| \cdot \gamma$, the only time values that we really need to check and rule out are $|V|$ multiples of γ. Therefore, at line 6, η exists and it can be easily found in time $O(|\Sigma_P| \cdot |V|^2)$. So, at line 7, for each $s \in \Sigma_P$ and $v \in V_s^+$, the value $[\sigma'_\gamma(s)]_v$ is shifted to the left by setting $[\sigma'_\gamma(s)]_v \leftarrow [\sigma'_\gamma(s)]_v - \eta$. Then, the following π-ES $\sigma \in \mathcal{S}_\Gamma$ is constructed at lines 9-13: for each $s \in \Sigma_P$ and $v \in V_s^+$, the execution-time is set $[\sigma(s)]_v^t \leftarrow \lfloor [\sigma'_\gamma(s)]_v \rfloor$, and the ordering $[\sigma(s)]^\pi$ follows the ordering on P that is induced by $\sigma'_\gamma(s)$. Finally, $\langle \text{YES}, \sigma\rangle$ is returned to output at line 14.

To conclude, we can prove the main result of this section.

Proof of Theorem 5. The correctness of Algorithm 3 follows directly from Theorems 6 and 3, plus the fact that $\eta \in [0, 1)$ can be computed easily, at line 6, as we have already mentioned above. The (pseudo) singly-exponential time complexity of Algorithm 3 follows from that of Theorem 3 plus the fact that all the integer weights in Γ are scaled-up by a factor $1/\gamma = |\Sigma_P| \cdot |V|^2 + 1$ in Γ'_γ; also notice that $\eta \in [0, 1)$ can be computed in time $O(|\Sigma_P| \cdot |V|^2)$, as we have already mentioned. Therefore, all in, the time complexity stated in Theorem 3 increases by a factor $1/\gamma = |\Sigma_P| \cdot |V|^2 + 1$. □

IV. Related Works

This section discusses of some related approaches offered in the current literature. The article of Tsamardinos, et al. [12] introduced DC for CSTNs. Subsequently, this notion has been analyzed and further formalized in [10], finally leading to a sound notion of DC for CSTNs. However, neither of these two works takes into account an instantaneous reaction-time. Cimatti, et al. [2] provided the first sound-and-complete procedure for checking the Dynamic-Controllability of CSTNs with Uncertainty (CSTNUs) and this algorithm can be employed for checking DC on CSTNs as a special case. Their approach is based on reducing the problem to solving Timed Game Automata (TGA). However, solving TGAs is a problem of much higher complexity than solving MPGs. Indeed, no upper bound is given in [2] on the time complexity of their solution. Moreover, neither ε-DC nor any other notion of DC with an instantaneous reaction-time are dealt with in that work. The first work to approach a notion of DC with an instantaneous reaction-time is [11]; its aim was to offer a sound-and-complete propagation-based DC-checking algorithm for CSTNs. The subsequent work [9] extended and amended [11] so that to check ε-DC, both for $\varepsilon > 0$ and for $\varepsilon = 0$. However, to the best of our knowledge, the worst-case complexity of those algorithms is currently unsettled. Moreover, it is not clear to us how one variant of the algorithm offered in [9], [11] (i.e., the one that aims at checking DC with an instantaneous reaction-time) can adequately handle cases like the CSTN counter-example $\Gamma_\Box$ that we have provided in Example 2. In summary, we believe that the present work can possibly help in clarifying DC with an instantaneous reaction-time also when the perspective had to be that of providing sound-and-complete algorithms based on the propagation of labelled temporal constraints.

V. Conclusion

The notion of ε-DC has been introduced and analysed in [5] where an algorithm was also given to check whether a CSTN is ε-DC. By the interplay between ε-DC and the standard notion of DC, also disclosed in [5], this delivered the first (pseudo) singly-exponential time algorithm checking whether a CSTN is DC (essentially, DC-Checking reduces to ε-DC-Checking for a suitable value of ε). In this paper, we proposed and formally defined π-DC, a natural and sound notion of DC for CSTNs in which the planner is allowed to react instantaneously to the observations that are made during the execution. A neat counter-example shows that π-DC with instantaneous reaction-time is not just the special case of ε-DC with $\varepsilon = 0$. Therefore, to conclude, we offer the first sound-and-complete π-DC-Checking algorithm for CSTNs. The time complexity of the procedure is still (pseudo) singly-exponential in $|P|$. The solution is based on a simple reduction from π-DC-Checking to DC-Checking of CSTNs.

References

[1] L. Brim, J. Chaloupka, L. Doyen, R. Gentilini, and J.F. Raskin. Faster algorithms for mean-payoff games. *Formal Methods in System Design*, 38(2):97–118, 2011.

[2] A. Cimatti, L. Hunsberger, A. Micheli, R. Posenato, and M. Roveri. Sound and complete algorithms for checking the dynamic controllability of temporal networks with uncertainty, disjunction and observation. In *21st Intern. Symp. on Temp. Repres. and Reasoning, TIME 2014, Verona, Italy*, pages 27–36, 2014.

[3] C. Comin, R. Posenato, and R. Rizzi. A tractable generalization of simple temporal networks and its relation to mean payoff games. In *21th International Symposium on Temporal Representation and Reasoning (TIME 2014)*, Verona, Italy, Sept 2014.

[4] C. Comin, R. Posenato, and R. Rizzi. Hyper temporal networks. *Constraints*, pages 1–39, 2016.

[5] C. Comin and R. Rizzi. Dynamic consistency of conditional simple temporal networks via mean payoff games: a singly-exponential time DC-Checking. In *22nd International Symposium on Temporal Representation and Reasoning (TIME 2015)*, pages 19–28. IEEE CPS, September 2015.

[6] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49(1–3):61–95, 1991.

[7] A. Ehrenfeucht and J. Mycielski. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8(2):109–113, 1979.

[8] E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*. Springer-Verlag New York, Inc., New York, NY, USA, 2002.

[9] L. Hunsberger and R. Posenato. Checking the dynamic consistency of conditional temporal networks with bounded reaction times. In *Proceedings of the Twenty-Sixth International Conference on Automated Planning and Scheduling, ICAPS 2016, London, UK, June 12-17, 2016.*, pages 175–183, 2016.

[10] L. Hunsberger, R. Posenato, and C. Combi. The dynamic controllability of conditional stns with uncertainty. In *Proc. of the Plan. and Plan Exec. for Real-World Syst.: Princip. and Pract. (PlanEx)*, ICAPS-2012, page 121–128, Atibaia, Sao Paulo, Brazil, 2012.

[11] L. Hunsberger, R. Posenato, and C. Combi. A sound-and-complete propagation-based algorithm for checking the dynamic consistency of conditional simple temporal networks. In *22nd International Symposium on Temporal Representation and Reasoning, TIME 2015, Kassel, Germany, September 23-25, 2015*, pages 4–18, 2015.

[12] I. Tsamardinos, T. Vidal, and M. Pollack. Ctp: A new constraint-based formalism for conditional, temporal planning. *Constraints*, 8(4):365–388, 2003.

[13] U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158:343–359, 1996.