Timed Collaborative Systems with Real Time

Max Kanovich[1], Tajana Ban Kirigin[2], Vivek Nigam[3], Andre Scedrov[4]

[1] Queen Mary, University of London, UK

[2] University of Rijeka, HR

3 Federal University of Paraíba, João Pessoa, Brazil

[4] University of Pennsylvania, USA

Why time?

In collaborative systems time is often essential for expressing:

- goals of the collaboration
- initial conditions
- undesired states
- procedures and rules of the collaboration

These specifications mention time explicitly.

Example: complying with deadlines; prompt reactions to some events

Why real time?

In many collaborative systems it is enough to use discrete time, e.g. days or hours.

We deal with discrete time with TLSTSes in [RTA,12]

Why real time?

In many collaborative systems it is enough to use discrete time, e.g. days or hours.

- We deal with discrete time with TLSTSes in [RTA,12]

However, some systems require real time.

- Example: Distance Bounding Protocols

Why real time?

Distance Bounding Protocols

Time challenge:

$A \xrightarrow{m} B$ A sends at time t_0

$A \xleftarrow{f(m)} B$ A receives at time t_1

If the round time $t_1 - t_0 < R$, then B is within a radius r from A.

Otherwise, there is no information on B's location.

Distance Bounding Protocols

Verify whether an intruder can impersonate someone else?

Is it possible for an intruder to appear to be closer than he actually is?

Timed Collaborative Systems with Real Time

Distance Bounding Protocols

Verify whether an intruder can impersonate someone else?

Is it possible for an intruder to appear to be closer than he actually is?

Formal specification and verification of such systems requires explicit real time and fresh values.

Agenda

- **Local State Transition Systems**
- Fresh Values
- Timed Collaborative Systems
- Real Time

Collaborative Systems

Closed Room

Examples: administrative tasks, protocols

- Agents collaborate to achieve some common goal.
- No intruder can enter the system.
- However, agents do not completely trust other agents.
- Therefore, while collaborating, an agent might not want some confidential information to be leaked.

Collaborative Systems [Kanovich, Rowe, and Scedrov]

Model – Local State Transition System (LSTS)

- FOL signature
- Configurations are multisets of facts:

$$\{\text{Nurse}(\text{Tom}, \text{id1}, \text{blood}), \text{Nurse}(\text{Sam}, \text{id2}, \text{blood})\}$$

- Actions are rewrite rules:

$$\text{Nurse}(X, Y, \text{blood}) \to \text{Nurse}(\text{blank}, Y, \text{blood})$$

$$\text{Lab}(\text{id}, \text{blood}) \to \text{Lab}(\text{id}, \text{testResults})$$

- Goals are multisets of facts:

$$\{\text{Doctor}(\text{testResults}, \text{Tom})\}$$

- Critical configurations are configurations that have to be avoided

$$\{\text{Lab}(\text{testResults}, \text{Tom})\} \qquad \{\text{Nurse}(\text{Tom}, \text{id1}, \text{blood}), \text{Nurse}(\text{Sam}, \text{id1}, \text{blood})\}$$

Previous results [Kanovich, Rowe, and Scedrov, CSF'07, CSF'09, Rowe PhD Dissertation UPENN'09]

The planning problem

Is there a plan from an initial configuration to a configuration containing a goal such that no critical configuration is reached along the plan?

Example:
the test results of a patient should not be publicly leaked with the patient's name.

Previous results [Kanovich, Rowe, and Scedrov, CSF'07, CSF'09, Rowe PhD Dissertation UPENN'09]

The planning problem

Is there a plan from an initial configuration to a configuration containing a goal such that no critical configuration is reached along the plan?

Example:
the test results of a patient should not be publicly leaked with the patient's name.

Assumption

Balanced actions, that is actions that have the same number of facts in their pre and post conditions.

Along a plan, configurations have the same number of facts.
Intuitively, agents have bounded memory.

Previous results [Kanovich, Rowe, and Scedrov, CSF'07, CSF'09, Rowe PhD Dissertation UPENN'09]

The planning problem

Is there a plan from an initial configuration to a configuration containing a goal such that no critical configuration is reached along the plan?

Example:
the test results of a patient should not be publicly leaked with the patient's name.

Assumption

Balanced actions, that is actions that have the same number of facts in their pre and post conditions.

Along a plan, configurations have the same number of facts.
Intuitively, agents have bounded memory.

Complexity Results

Balanced actions:

PSPACE-complete

Not necessarily balanced actions:

Undecidable

Systems with balanced actions

Challenge

- Although checking for the existence of plan is in PSPACE, it turns out that to write down the entire plan may require exponential space because the plan might be exponentially long.

Challenge

- Although checking for the existence of plan is in PSPACE, it turns out that to write down the entire plan may require exponential space because the plan might be exponentially long.

Example: Towers of Hanoi

$$Clear(x)\ On(x,y)\ Clear(z)\ S(x,z) \rightarrow Clear(x)\ Clear(y)\ On(x,z)\ S(x,z)$$

Given n disks plans must be of exponential length $2^n - 1$, at least.

Systems with balanced actions

Challenge

- Although checking for the existence of plan is in PSPACE, it turns out that to write down the entire plan may require exponential space because the plan might be exponentially long.

Example: Towers of Hanoi

$$Clear(x)\ On(x,y)\ Clear(z)\ S(x,z) \rightarrow Clear(x)\ Clear(y)\ On(x,z)\ S(x,z)$$

Given n disks plans must be of exponential length $2^n - 1$, at least.

Solution

- [CSF'07] Scheduling a plan in PSPACE

Agenda

- Local State Transition Systems
- **Fresh Values**
- Timed Collaborative Systems
- Real Time

Fresh values [Kanovich, Ban Kirigin, Nigam, and Scedrov, FAST'10]

Motivation

Agents might need to create fresh values or *nonces*:

$$\mathrm{nurse}(\mathrm{Tom},\ \mathrm{blank},\ \mathrm{blood}) \rightarrow \exists\ \mathrm{testNo}.\mathrm{nurse}(\mathrm{Tom},\ \mathrm{testNo},\ \mathrm{blood})$$

Each sample should have a different number assigned.

- opening a new bank account;
- changing a customer's password;
- creating a transaction number or a case number;
- security protocols.

Balanced actions that create fresh values

$$\mathrm{nurse}(\mathrm{Tom},\ \mathrm{blank},\ \mathrm{blood}) \rightarrow \exists\ \mathrm{testNo}.\mathrm{nurse}(\mathrm{Tom},\ \mathrm{testNo},\ \mathrm{blood})$$

The fresh value uses the memory slot previously used by the updated value.

Balanced actions that create fresh values

$$\mathrm{nurse}(\mathrm{Tom},\ \mathrm{blank},\ \mathrm{blood}) \to \exists\ \mathrm{testNo}.\mathrm{nurse}(\mathrm{Tom},\ \mathrm{testNo},\ \mathrm{blood})$$

The fresh value uses the memory slot previously used by the updated value.

Agents have a bounded memory even when they can create fresh values.

Balanced actions that create fresh values

$$\text{nurse}(\text{Tom, blank, blood}) \to \exists\ \text{testNo}.\text{nurse}(\text{Tom, testNo, blood})$$

The fresh value uses the memory slot previously used by the updated value.

Agents have a bounded memory even when they can create fresh values.

$$\to \exists n.A(n)$$

Whenever such an unbalanced rule is used, an extra memory slot is required to store the nonce created.
That is, agents possess unbounded memory.

Systems with balanced actions

Challenge

- Although checking for the existence of plan is in PSPACE, it turns out that to write down the entire plan may require exponential space and exponentially many mutually distinct nonces.

Example: Towers of Hanoi, suitably modified to have balanced actions that always creates fresh values.

Systems with balanced actions

Challenge

- Although checking for the existence of plan is in PSPACE, it turns out that to write down the entire plan may require exponential space and exponentially many mutually distinct nonces.

  Example: Towers of Hanoi, suitably modified to have balanced actions that always creates fresh values.

Solution

- [FAST 10] We exploit the fact that the number of constants in a configuration is bounded and a priori fix a small number of nonce names. We then show how to reuse obsolete constants instead of updating with fresh constants.

Summary of results

Planning Problem		
Balanced Actions	Nonces are not allowed	PSPACE-complete [Kanovich et al., CSF'07]
	Nonces are allowed	PSPACE-complete [Kanovich et al., FAST'10]
Actions not necessarily balanced		Undecidable [Kanovich et al., CSF'09]

Agenda

- Local StateTransition Systems
- Fresh Values
- **Timed Collaborative Systems**
- Real Time

Timed Collaborative Systems

M. Kanovich, T. Ban Kirigin, V. Nigam, A. Scedrov, C. L. Talcott, R. Perović.

- Towards an automated assistant for clinical investigations. IHI, 2012.
- A rewriting framework for activities subject to regulations. RTA, 2012.

Motivational application: Clinical Investigations

- Before drugs can be made available to the general public, their effectiveness has to be experimentally validated. At the final stages human subjects are involved. These tests are called **Clinical Investigations**.
- Pharmaceutical companies (Sponsor), clinical research organizations (CRO), health institutions (HI) and government regulatory agencies collaborate in order to carry out Cis.

Motivational application: Clinical Investigations

- Before drugs can be made available to the general public, their effectiveness has to be experimentally validated. At the final stages human subjects are involved. These tests are called **Clinical Investigations**.
- Pharmaceutical companies (Sponsor), clinical research organizations (CRO), health institutions (HI) and government regulatory agencies collaborate in order to carry out CIs

Key Concerns

Safety of Subjects

One should avoid at all costs that the health of subjects is compromised during the tests.

Conclusive Data Collection

CI's should be carried in order to obtain the most conclusive results/data without compromising the health of subjects.

Motivational application: Clinical Investigations

Regulations

"Any adverse experience associated with the use of the drug that is both serious and unexpected; […]
Each notification shall be made as soon as possible and *in no event later than 15 calendar days* after the sponsor's initial receipt of the information."

Procedures

Procedures are elaborated by specialists explaining how one should carry out CIs, so that the most conclusive data is collected and the health of subjects is not compromised.

Both procedures and regulations mention time explicitly.

Timestamps and time constraints [IHI'12]

Motivation

$$\mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{no})@T_1 \mid \{T_1 - 5 \leq T \leq T_1 + 5\} \longrightarrow \mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{yes})@T$$

Timestamps and time constraints [IHI'12]

Motivation

$$\mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{no})@T_1 \mid \{T_1 - 5 \leq T \leq T_1 + 5\} \longrightarrow \mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{yes})@T$$

Global Time

Timestamps and time constraints [IHI'12]

Motivation

$$\mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{no})@T_1 \mid \{T_1 - 5 \leq T \leq T_1 + 5\} \longrightarrow \mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{yes})@T$$

Global Time

a scheduled visit has a tolerance of 5 days

Timestamps and time constraints [IHI'12]

Motivation

$$\mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{no})@T_1 \mid \{T_1 - 5 \leq T \leq T_1 + 5\} \longrightarrow \mathrm{Time}@T,\ \mathrm{Visit}(\mathrm{I},\ \mathrm{ID},\ \mathrm{yes})@T$$

Global Time

a scheduled visit has a tolerance of 5 days

Other examples:

- time constraints often appear in legislation

  e.g. medical, financial;

- timestamps are also used in protocols.

Timestamps and time constraints [IHI'12]

Timed Goal Configurations

Data of the subjects have to be collected at the correct times:

Configuration

$$\{\mathrm{Time}@T, \mathrm{Data}(Id, 1)@T_1, \ldots, \mathrm{Data}(Id, 25)@T_{25}\}$$

Time constraints

$$T_i + 23 \leq T_{i+1} \leq T_i + 33$$

$$\text{and that } T > T_i, \text{ for } 1 \leq i \leq 25$$

Timestamps and time constraints [IHI'12]

Timed Goal Configurations

Data of the subjects have to be collected at the correct times:

Configuration $\{\mathrm{Time}@T, \mathrm{Data}(Id, 1)@T_1, \ldots, \mathrm{Data}(Id, 25)@T_{25}\}$

Time constraints $T_i + 23 \leq T_{i+1} \leq T_i + 33$

and that $T > T_i$, for $1 \leq i \leq 25$

Timed Critical Configurations

regulatory agency is not informed within 15 days an unexpected event is detected:

Configuration $\{\mathrm{Detect}(Id)@T_1, \mathrm{Report}(Id)@T_2\}$

Time constraints $\{T_2 > T_1 + 15\}$

Timestamps and time constraints [IHI'12]

Assumptions:

- Discrete time: timestamps are natural numbers.

  For example, a timestamp can denote the time when the fact was created or the time until the fact is valid.

- Global time: *Time@T*

- Time constraints are arithmetic comparisons of the form:

$$T_1 \circ T_2 + D, \text{ where } \circ \in \{<, \leq, =, \geq, >\}$$

where D is a natural number and T_1 and T_2 are time variables.

Time constraints are relative i.e. they are invariant with respect to time translation $t \rightarrow t + t_0$.

Timestamps and time constraints [IHI'12]

Assumptions:

- Actions are balanced.
- Time tick action: $Time@T \rightarrow_{clock} Time@(T+1)$
- Time constraints are attached to actions.

$$Time@T,W \mid \Upsilon \rightarrow \exists \mathbf{x}.Time@T,W'$$

- Timestamps of created facts in an action at the moment T are of the form:

$T + D$, where D is a non-negative integer.

Handling the unboundedness of time

Challenge

- Overcome the fact that the domain of timestamps is unbounded.

  Example: a plan where the global time advances eagerly.

$$\mathrm{Time@0}, W \longrightarrow_{clock} \mathrm{Time@1}, W \longrightarrow_{clock} \mathrm{Time@2}, W \longrightarrow_{clock} \cdots$$

Solution

We propose an equivalence relation on configurations based on the time differences of facts:

Handling the unboundedness of time

Solution

We propose an equivalence relation on configurations based on the time differences of facts:

Truncated time difference of two facts $P@T_1$ and $Q@T_2$:

$$\delta_{P,Q} = \begin{cases} T_2 - T_1, & \text{provided } T_2 - T_1 \leq D_{max} \\ \infty, & \text{otherwise} \end{cases}$$

where D_{max} is an upper bound on the numbers appearing in the TLSTS.

Handling the unboundedness of time

Solution

We propose an equivalence relation on configurations based on the time differences of facts:

Truncated time difference of two facts $P@T_1$ and $Q@T_2$:

$$\delta_{P,Q} = \begin{cases} T_2 - T_1, \text{ provided } T_2 - T_1 \leq D_{max} \\ \infty, \text{ otherwise} \end{cases}$$

where D_{max} is an upper bound on the numbers appearing in the TLSTS.

Informally: Two configurations are equivalent if they have the same facts and the same truncated time differences.

Example

Assume $D_{max} = 3$, then the following configurations are equivalent:

Example

Assume $D_{max} = 3$, then the following configurations are equivalent:

	Time Differences	Truncated Time Differences	Time Differences	
$R@3$				$R@0$
$P@4$				$P@1$
$\mathrm{Time}@11$				$\mathrm{Time}@6$
$Q@12$				$Q@7$
$S@14$				$S@9$

Example

Assume D_{max} = *3*, then the following configurations are equivalent*:*

	Time Differences	Truncated Time Differences	Time Differences	
$R@3$				$R@0$
	1		1	
$P@4$				$P@1$
	7		5	
$\mathrm{Time}@11$				$\mathrm{Time}@6$
	1		1	
$Q@12$				$Q@7$
	2		2	
$S@14$				$S@9$

Example

Assume $D_{max} = 3$, then the following configurations are equivalent:

	Time Differences	Truncated Time Differences	Time Differences	
$R@3$				$R@0$
	1	1	1	
$P@4$				$P@1$
	7	∞	5	
$\mathrm{Time}@11$				$\mathrm{Time}@6$
	1	1	1	
$Q@12$				$Q@7$
	2	2	2	
$S@14$				$S@9$

Example

Assume $D_{max} = 3$, then the following configurations are equivalent:

	Time Differences:	Truncated Time Differences:	Time Differences:	
$R@3$				$R@0$
	1	1	1	
$P@4$				$P@1$
	7	∞	5	
$\mathrm{Time}@11$				$\mathrm{Time}@6$
	1	1	1	
$Q@12$				$Q@7$
	2	2	2	
$S@14$				$S@9$

Canonical form called δ-representation:

$$\langle R, 1, P, \infty, \mathrm{Time}, 1, Q, 2, S \rangle$$

Equivalent configurations and relative time constraints

Lemma: Let S and S' be equivalent configurations and let C be a relative time constraint. S satisfies C if and only if S' satisfies C.

Hence, if an action is applicable in the configuration S it will also be applicable in the configuration S'.
Moreover, if S is a goal (respectively, critical) configuration, then S' is also a goal (respectively, critical) configuration.

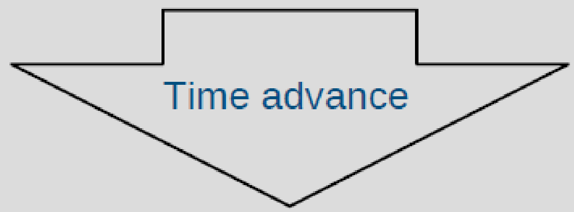
Future bounded configurations

Handling Time Advances

Time advances may cause problems for the *bisimulation* that we intend to provide with our equivalence:

Assume $D_{max} = 3$ and the following configurations:

Equivalent: $\{\mathrm{Time}@0, P@5\}$ $\{\mathrm{Time}@0, P@4\}$



Not Equivalent: $\{\mathrm{Time}@1, P@5\}$ $\{\mathrm{Time}@1, P@4\}$

Future bounded configurations

Handling Time Advances

Time advances may cause problems for the *bisimulation* that we intend to provide with our equivalence:

Assume $D_{max} = 3$ and the following configurations that are not future bounded:

Equivalent: $\{\mathrm{Time}@0, P@5\}$ $\{\mathrm{Time}@0, P@4\}$

Time advance

Not Equivalent: $\{\mathrm{Time}@1, P@5\}$ $\{\mathrm{Time}@1, P@4\}$

We manage this problem by taking a future bounded initial configuration where the time differences between each of the future facts and the current global time is bounded by D_{max}.

Handling Time Advances

Lemma: Actions preserve future boundedness of configurations.

This is because of the following condition on actions:

> The timestamps of created facts in an action at a moment T are of the form $T + D$, where D is non-negative integer.

Actions preserve equivalences

Theorem: For a given Timed Local State Transition System (TLSTS) any plan starting from a future bounded configuration can be conceived as a plan over its δ-representations.

We only need to consider the planning problem with a bounded number of δ-representations with respect to:

- the number of facts in the future bounded initial configuration;
- the upper bound on the size of facts;
- the upper bound, D_{max}, of the numbers appearing in the theory.

Summary of results for Timed Collaborative Systems

Planning Problem		
Balanced Actions	LSTS	PSPACE-complete
	TLSTS	PSPACE-complete
Actions not necessarily balanced		Undecidable

The above PSPACE result also relates to TLSTSes with fresh values.

Agenda

- Local State Transition Systems
- Fresh Values
- Timed Collaborative Systems
- **Real Time**

Timed Collaborative Systems with Real Time

- Extending our discrete time model TLSTSes [RTA'12] to meet the needs for real time

- Motivation: Distance Bounding Protocols
  Cyberphisical systems - autonomous robots that move around and often need to know where other agents are and also need to plan taking time into account
  Time syncrhonization mechanisms - algorithms that are used to synchronize time of several machines according to master time, such as an atomic clock.

- Investigating the complexity of the planning problem for the new model with real time

Timestamps and time constraints

Assumptions:

- Real time: timestamps are non-negative real numbers.
- Time constraints are arithmetic comparisons of the form:

$$T_1 \circ T_2 + D, \text{ where } \circ \in \{<, \leq, =, \geq, >\}$$

where D is a natural number and T_1 and T_2 are time variables.

- Timed goal and critical configurations:
  time constraints attached to configurations.

Timestamps and time constraints

Assumptions:

- Actions are balanced and can create fresh values.
- Time tick action: $\mathit{Time@T} \rightarrow \mathit{Time@(T{+}t)}$
  where t is a positive real number.
- Time constraints are attached to actions:

$$\mathit{Time@T,W} \mid \Upsilon \rightarrow \exists \mathbf{x}.\mathit{Time@T,W'}$$

- The timestamps of created facts in an action at the moment T are of the form $T + D$, where D is non-negative integer.

Handling the unboundedness of time

Challenge

- Overcome the fact that the domain of timestamps is unbounded.

  Example: A plan where the global time advances eagerly.

When time is discrete, we handle the unboundedness of time with a bounded number of δ-representations based on the time differences of facts.
With real numbers as timestamps, there would be an infinite number of such representations.

Handling the unboundedness and density of time

Challenge

- Additionally, deal with the density of the domain of timestamps:

$$Time@T \;\rightarrow\; Time@(T+t)$$

where *t* is a positive real number.

Handling the unboundedness and density of time

Solution

We propose a novel equivalence relation on configurations.

Let D_{max} be a natural number that is an upper bound on the numbers appearing in the specification of the given a TLSTS T. Configurations S_1 and S_2 are equivalent w.r.t. D_{max} if the following conditions are satisfied:

Handling the unboundedness and density of time

Solution

We propose a novel equivalence relation on configurations.

Let D_{max} be a natural number that is an upper bound on the numbers appearing in the specification of the given a TLSTS T. Configurations S_1 and S_2 are equivalent w.r.t. D_{max} if the following conditions are satisfied:

(1-δ) S_1 and S_2 have the same δ-representations w.r.t. D_{max} when considering only the integer part of the truncated time differences.

Truncated time difference of two facts $P@T_1$ and $Q@T_2$:

$$\delta_{P,Q} = \begin{cases} T_2 - T_1, & \text{provided } T_2 - T_1 \leq D_{max} \\ \infty, & \text{otherwise} \end{cases}$$

Handling the unboundedness and density of time

Solution

We propose a novel equivalence relation on configurations.

Let D_{max} be a natural number that is an upper bound on the numbers appearing in the specification of the given a TLSTS T. Configurations S_1 and S_2 are equivalent w.r.t. D_{max} if the following conditions are satisfied:

(1-δ) S_1 and S_2 have the same δ-representations w.r.t. D_{max} when considering only the integer part of the truncated time differences.

(2-circle) when ordering their facts considering only the decimal part of timestamps, one obtains the same list of facts for S_1 and S_2.

Solution – Circle Abstractions

We propose a novel equivalence relation on configurations.

Let D_{max} be a natural number that is an upper bound on the numbers appearing in the specification of the given a TLSTS T. Configurations S_1 and S_2 are equivalent w.r.t. D_{max} if the following conditions are satisfied:

(1-δ) S_1 and S_2 have the same δ-representations w.r.t. D_{max} when considering only the integer part of the truncated time differences.

(2-circle) when ordering their facts considering only the decimal part of timestamps, one obtains the same list of facts for S_1 and S_2.

Circle Abstractions - Example

Assume $D_{max}=2$, then the following configurations are equivalent:

$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\ \}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\ \}$$

	Time differences	Time differences	
$P_0@0.4$			$P_0@3.2$
	1.1	1.3	
$P_1@1.5$			$P_1@5.4$
	3.9	3.7	
$Time@5.4$			$Time@8.2$
	2.2	2.4	
$P_2@7.6$			$P_2@10.6$

Circle Abstractions - Example

Assume $D_{max}=2$, then the following configurations are equivalent:

$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\ \}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\ \}$$

	Time differences	Integer part	Integer part	Time differences	
$P_0@0.4$					$P_0@3.2$
	1.1	1	1	1.3	
$P_1@1.5$					$P_1@5.4$
	3.9	3	3	3.7	
$Time@5.4$					$Time@8.2$
	2.2	2	2	2.4	
$P_2@7.6$					$P_2@10.6$

Circle Abstractions - Example

Assume $D_{max}=2$, then the following configurations are equivalent:

$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\ \}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\ \}$$

	Time differences	Integer part	Truncated time differences	Integer part	Time differences	
$P_0@0.4$						$P_0@3.2$
	1.1	1	1	1	1.3	
$P_1@1.5$						$P_1@5.4$
	3.9	3	∞	3	3.7	
$Time@5.4$						$Time@8.2$
	2.2	2	2	2	2.4	
$P_2@7.6$						$P_2@10.6$

Circle Abstractions - Example

Assume D_{max}=*2*, then the following configurations are equivalent*:*

$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\ \}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\ \}$$

	Time differences	Integer part	Truncated time differences	Integer part	Time differences	
$P_0@0.4$						$P_0@3.2$
	1.1	1	1	1	1.3	
$P_1@1.5$						$P_1@5.4$
	3.9	3	∞	3	3.7	
$Time@5.4$						$Time@8.2$
	2.2	2	2	2	2.4	
$P_2@7.6$						$P_2@10.6$

(1-δ) for both S_1 and S_2 we obtain the representation:

$$[\ P_0,\ 1,\ P_1\ ,\ \infty\ ,\ Time,\ 2,\ P_2\ ]$$

Circle Abstractions - Example

Assume D_{max}=2, then the following configurations are equivalent:

$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\ \}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\ \}$$

(2-circle) Ordering the facts considering only the decimal part of timestamps for both S_1 and S_2, we obtain the list:

$$[\ P_0 = Time,\ P_1,\ P_2\ ]$$

Circle Abstractions - Example

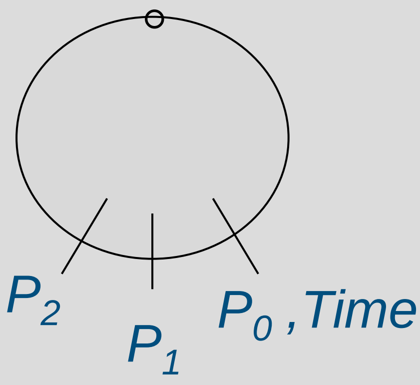
Assume D_{max}=2, then the following configurations are equivalent:

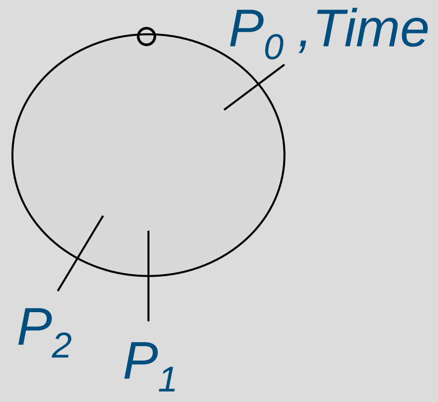
$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\ \}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\ \}$$

(2-circle) Ordering the facts considering only the decimal part of timestamps for both S_1 and S_2, we obtain the list:

$$[\ P_0 = Time,\ P_1,\ P_2\ ]$$





Circle Abstractions - Example

Assume $D_{max}=2$, then the following configurations are equivalent:

$$S_1 = \{P_0@0.4,\ P_1@1.5,\ Time@5.4,\ P_2@6.6\}$$

$$S_2 = \{P_0@3.2,\ P_1@4.5,\ Time@8.2,\ P_2@9.6\}$$

(1-δ) for both S_1 and S_2 we obtain the representation:

$$[\ P_{0,}\ 1,\ P_1\ ,\ \infty\ ,\ Time,\ 2,\ P_2\ ]$$

(2-circle) Ordering the facts considering only the decimal part of timestamps for both S_1 and S_2 we obtain the list:

$$[\ P_0 = Time,\ P_1,\ P_2\ ]$$

Circle Abstractions

Lemma: The equivalence relation among configurations is well defined w.r.t. time constraints, goal and critical configurations and action application for TLSTSes with a future bounded initial configuration.

Circle Abstractions

Lemma: The equivalence relation among configurations is well defined w.r.t. time constraints, goal and critical configurations and action application for TLSTSes with a future bounded initial configuration.

For a given planning problem we obtain a finite number of circle abstractions with which we are able to represent the infinite space of configurations.

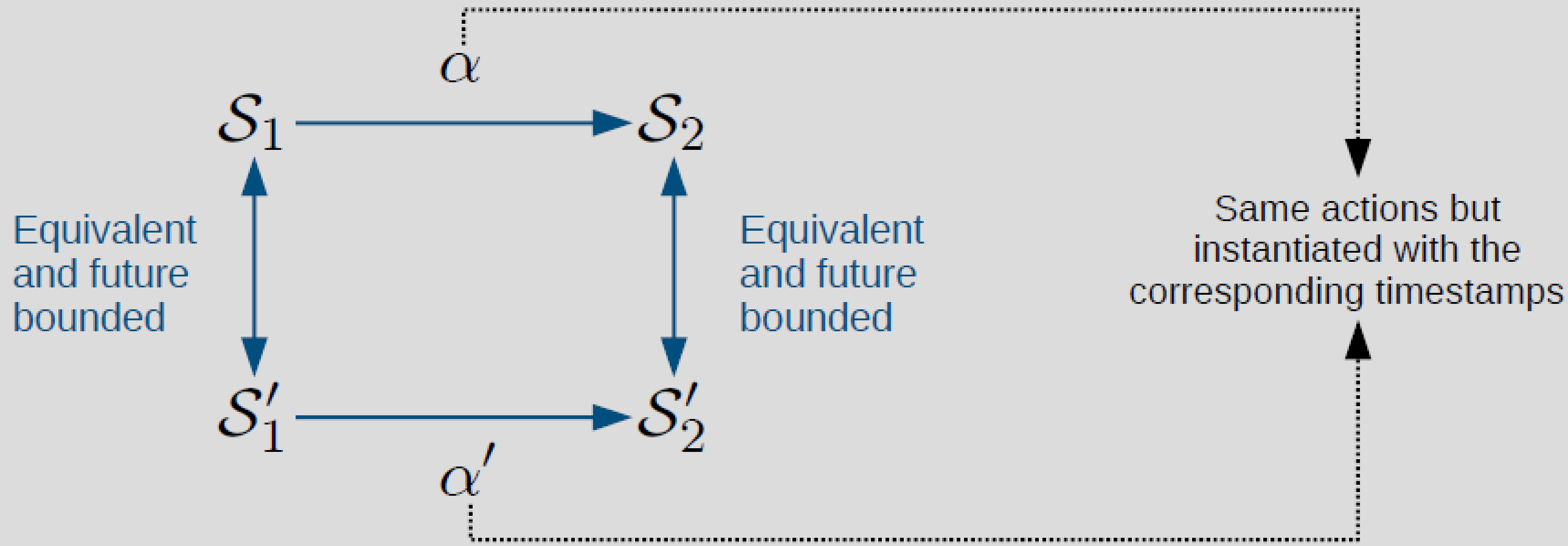
Circle Abstractions

Lemma: The equivalence relation among configurations is well defined w.r.t. time constraints, goal and critical configurations and action application for TLSTSes with a future bounded initial configuration.

Theorem: The planning problem for TLSTSes with real time is PSPACE-complete.

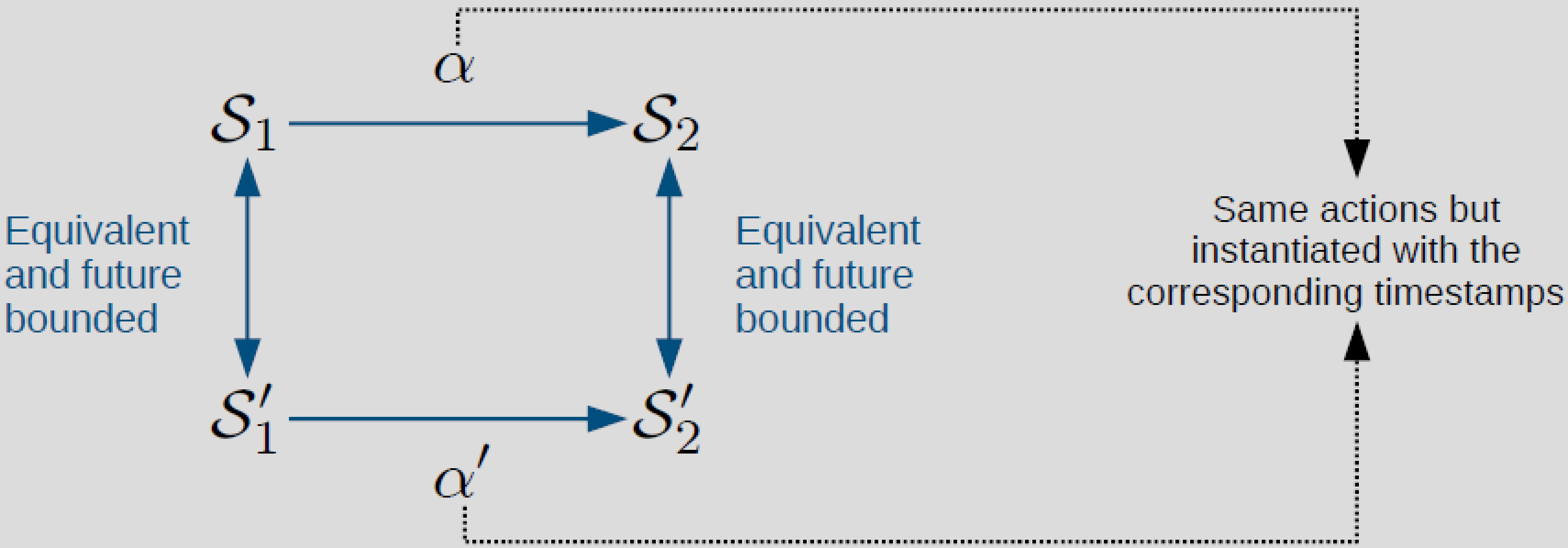
Proof Sketch

Simulation Argument



Proof sketch

Simulation Argument



Any plan starting from a future bounded configuration can be conceived as a plan over circle abstractions.

Proof Sketch

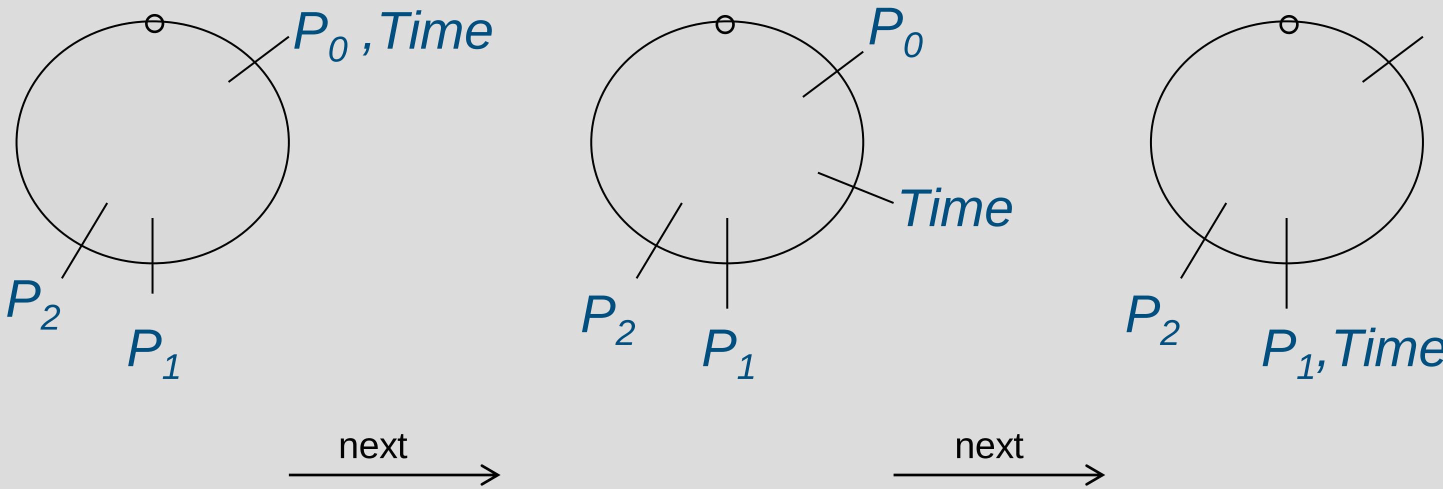
Handling Time Advances

Circle abstractions do not contain the information of exact time differences between timestamps and the global time.
In order to model time advancement on circle abstractions we add a special action ***next***.

Proof Sketch

Handling Time Advances

Circle abstractions do not contain the information of exact time differences between timestamps and the global time.
In order to model time advancement on circle abstractions we add a special action ***next***.
Application of *next* results in the circle abstraction in which time has advanced just enough to change the abstraction, not to jump over some abstractions:



Summary of Results for Timed Collaborative Systems

Planning Problem		
Balanced Actions	TLSTS with discrete time	PSPACE-complete
	TLSTS with real time	PSPACE-complete
Actions not necessarily balanced		Undecidable

The above PSPACE result also relates to TLSTSes with fresh values.

Future work

- Verification of systems that require explicit real time:

  Distance Bounding Protocols

  Cyberphisical systems

- Specification of asynchronous systems

  Time syncrhonization mechanisms

- Analysis of security protocols

  – timestamps, timing channels

Related work

- N. A. Durgin, P. Lincoln, J. C. Mitchell, and A. Scedrov. Multiset rewriting and the complexity of bounded security protocols. 1999.

- M. I. Kanovich, M. Okada, and A. Scedrov. Specifying real-time finite-state systems in linear logic, 1998.

- R. Alur and D.L. Dill. A theory of timed automata., 1994.

- S. Brands and D. Chaum. Distance-bounding protocols, 1993.