Recursive Representation of Periodicity and Temporal Reasoning

Luca Anselma
Dipartimento di Informatica
Università di Torino, Italy
anselma@di.unito.it
Outline

- Introduction
- Representation formalism
- Reasoning mechanisms
- Related works
- Conclusions
Outline

- Introduction
- Representation formalism
- Reasoning mechanisms
- Related works
- Conclusions
Introduction

Repeated events are widespread in many application domains.

Often (e.g., in clinical therapies), repeated events occur at regular times, i.e., they are periodic.

A compact, intuitive, and powerful representation formalism is needed.
Introduction

Desiderata

Example:
The therapy lasts 88 weeks. In the therapy, cotrimoxazole must be given twice daily on two consecutive days every week.

(an excerpt from a clinical guideline for Childhood Acute Lymphoblastic Leukaemia)

The repetitions are not only periodic, but also:
- multiply nested (2 cotrimoxazole(s) ⊆ a day ⊆ a week ⊆ 88 weeks), and
- imprecisely located (in what days of the week they occur?).
- Also temporal constraints between repetitions are needed (“consecutive days”).
Introduction

Goals of the task

We wish to perform efficient reasoning (i.e., consistency checking): trade-off between expressiveness and decidability/tractability.

Moreover: we wish to integrate it in a framework [Terenziani&Anselma, TIME03] that supports:
- qualitative [Allen] + quantitative [Dechter] temporal constraints;
- composite events;
- classes+instances [Terenziani&Anselma, Int J Intell Sys].
Introduction
Classes+Instances

CLASSES

Reservation (RS) 1-7 days before Lab_Test (LT) 1-48 hours before Report (RP)

RS1 RS2 LT1 LT2 RSk LTk RP1 RP2 RPk

INSTANCES

instance_of

temporal constraints

Luca Anselma, Università di Torino
Recursive Representation of Periodicity and Temporal Reasoning
TIME 2004
Outline

- Introduction
- Representation formalism
- Reasoning mechanisms
- Related works
- Conclusions
Representation Formalism

Periodicity constraints:

Repetition(E, RepSpec)

where RepSpec is a recursive structure.

Each level:

\[ R_i ::= <nRep_i, I-Time_i, constrs_i> \]

i.e., \( R_i \) means:

“repeat \( nRep_i \) times \( R_{i+1} \)
in a time lapse lasting \( I-Time_i \)
following the constraints \( constrs_i \)”
Representation Formalism

Periodicity constraints:

\[
\text{Repetition}(E, <R_1, \ldots, R_n>)
\]

\[
R_i ::= <\text{nRep}_i, \text{I-Time}_i, \text{constrs}_i> 
\]

\text{constrs}_i \text{ imposes temporal constraints between repetitions. It is a (possibly empty) set of:}

- \text{inBetweenAll}(\text{min}, \text{max})
- \text{inBetween}((\text{min}_1, \text{max}_1), \ldots, (\text{min}_{\text{nRep}-1}, \text{max}_{\text{nRep}-1}))
- \text{fromStart}(\text{min}, \text{max})
- \text{toEnd}(\text{min}, \text{max})
Representation Formalism

Examples

Example:

“*Intrathecal methotrexate must be administered 7 times during 88 weeks, never less than 10 weeks apart or more than 14 weeks apart*”.

(an excerpt from a clinical guideline for Childhood Acute Lymphoblastic Leukaemia)

\[
\text{Repetition} (\text{IntrathecalMethotrexate}, \langle 7, 88 \text{wk}, \{\text{inBetweenAll}(10 \text{wk}, 14 \text{wk})}\rangle >
\]
Example (from slide # 5):

“The therapy lasts 88 weeks. In the therapy, cotrimoxazole must be given twice daily on two consecutive days every week”.

Repetition(cotrimoxazole,
\langle\_, \text{88wk}, \{}\rangle,
\langle2, \text{1wk}, \{\text{inBetweenAll}(0,0)}\rangle,\n\langle2, \text{1d}, \{}\rangle>>
Outline

- Introduction
- Representation formalism
- Reasoning mechanisms
- Related works
- Conclusions
Reasoning Mechanisms
Internal representation

Internal representation:

- STP (constraints between instances) [Dechter];

- STP-Tree (constraints between classes) [Terenziani&Anselma, TIME03].
Reasoning Mechanisms

STP-Tree

“One possible therapy for multiple myeloma is made by six cycles of 5-day treatment, each one followed by a delay of 23 days (for a total time of 24 weeks). Within each 5-day cycle, 2 inner cycles to be performed in parallel can be distinguished: the melphalan treatment, to be provided twice a day, for each of the 5 days, and the prednisone treatment, to be provided once a day, for each of the 5 days.”

(an excerpt from a clinical guideline for Multiple Mieloma)
Reasoning Mechanisms
Checking consistency on classes

procedure STPTreeConsistency(X: STPNode, RepSpec)
1) check that the periodicity constraint is well-formed
2) let Max the maximum duration of a repetition
3) impose (in X) that the maximum distance between each pair of points is the minimum between the current maximum distance and Max
4) X ← FloydWarshall(X)
5) return X

Complexity. \textit{STPTreeConsistency} \textit{runs in} \(O(\max\{n, C^3\})\), where \(n\) is the maximum number of nesting levels of the repetitions and \(C\) is the number of classes.
Reasoning Mechanisms
Checking consistency on classes

procedure STP_tree_consistency(X: STPNode, RepSpec)
1) let RepSpec = <R_1 = <nRep_1, IT_1, constrs_1>>, …, R_n = <nRep_n, IT_n, constrs_n>>
2) for i \(
\leq\) 1 to n do
3) \[\text{max}_i \leftarrow (\text{IT}_i - (\text{fromStart}_i\text{.min} + \sum_{j \in \text{inBetween}_i[j].min + \text{toEnd}_i\text{.min}})) / n\text{Rep}_i\]
4) if i < n then
5) if IT_{i+1} > \text{max}_i then return INCONSISTENT
6) else
7) impose (in X) that the maximum distance between each pair of points is the minimum between the current maximum distance and \text{max}_n
8) od
9) X \leftarrow \text{FloydWarshall}(X)
10) if X = INCONSISTENT then return INCONSISTENT else return X
Reasoning Mechanisms
Checking consistency on classes+instances

procedure integratedConsistency(T : CKB, I : IKB)
1) associate the classes with the instances
2) inherit (in I) the constraints from classes to instances
3) add to I the constraints relative to the repetition constraints
4) $I := \text{FloydWarshall}(I)$
5) return $I$

Complexity. \textit{integratedConsistency runs in $O(\max\{I^3, r^3C^3\})$, where I is the number of instances, r is the maximum number of repetitions. (assumptions: complete observability in the past and total temporal ordering of instances).}
Reasoning Mechanisms
Checking consistency on classes+instances

procedure integratedConsistency(T : CKB, I : IKB, NOW)
1) add to I the constraints that all instances start before NOW
2) unfoldNode(Root(T), I)
3) if there exists an instance in I not yet considered then return INCONSISTENT
4) I := FloydWarshall(I)
5) if I=INCONSISTENT then return INCONSISTENT
6) for each instance i not found by unfoldNode do
7) if NEC(Start(i) before NOW) then
8) return INCONSISTENT
Reasoning Mechanisms

procedure unfoldNode(X : STPNode, I : IKB)
1) add to I the placeholder class $C_X$
2) forall $C_A$ | $C_A$ is not a repeated class in X do
3) find in I the first instance i corresponding with class $C_A$
4) if i does not exist then add it to I
5) od
6) forall $C_R$ | $C_R$ is a repeated class in X do
7) let RepSpec = $<R_1, \ldots, R_n>$ be the repetition specification of class $C_R$
8) $C_{sub} :=$ unfoldRep(X, I, $C_R$, RepSpec)
9) add to I the constraints that $C_{sub} \subseteq C_X$
10) od
11) for each monadic constraint in X do
12) add the constraint to the corresponding classes in I
13) for each binary constraint in X do
14) add the constraint to the corresponding classes in I
15) return $C_X$
procedura unfoldRep(X : STPNode, I : IKB, C : Class, RepSpec = <R₁, R₂, ..., Rₙ>)
1) add to I the placeholder class C₁
2) let R₁ = <nRep₁, IT₁, constrs₁>
3) for r := 1 to nRep₁ do
   4) if R₁ is not the last level in RepSpec then
      Csubᵣ := unfoldRep(X, I, C, <R₂, ..., Rₙ>)
   5) else
      Csubᵣ := unfoldNode(child(C, X), I)
   6) od
7) add to I the constraint that duration of C₁ is IT₁
8) add to I the constraints that Csubᵣ ⊆ C₁
9) add to I the constraints that Csubᵢ₊₁ is after Csubᵣ
10) add to I the constraint fromStart in constrs₁ in R₁ between C₁ and Csubᵣ
11) add to I the constraint toEnd in constrs₁ in R₁ between CsubₙRep₁ and C₁
12) add to I the constraints inBetween and inBetweenAll in constrs₁ in R₁ between Csubᵣ and Csubᵢ₊₁
13) return C₁

TIME 2004
Outline

- Introduction
- Representation formalism
- Reasoning mechanisms
- Related works
- Conclusions
Related Works

This work is an extension of [Terenziani&Anselma, TIME03]. It did not support either an arbitrary number of nesting levels, or imprecision.

Also [Cukierman&Delgrande, TIME98;TIME00] proposed an expressive recursive language for representing periodicity and composite temporal objects related by Allen’s interval relations. It does not support either reasoning mechanisms or quantitative information.
Outline

- Introduction
- Representation formalism
- Reasoning mechanisms
- Related works
- Conclusions
Conclusions

**Recursive language** for periodicity/repetitions supporting arbitrarily nested repetitions, imprecision, and pauses between repetitions.

It integrates in a **general-purpose domain-independent temporal reasoner** supporting classes/instances, qualitative/quantitative temporal constraints.

Presented two **algorithms** for checking consistency on classes and on classes+instances.

(Sample) application areas: planning, workflow management, protocol/guideline management.
Conclusions

Applications

Currently, **application in clinical guidelines:**

- joint project with Azienda Ospedaliera San Giovanni Battista, Torino, Italy
- (starting) joint project with Cancer Research, London, UK

All examples are from Azienda Ospedaliera San Giovanni Battista and Cancer Research. Their issues **have motivated this work.**

The assumptions (complete observability (in the past), and total temporal ordering of instances) are reasonable, among others, in the domain of clinical guidelines.
Conclusions

Future work:
- dealing with (different forms of) disjunctions;
- removing assumptions.

Tractable approach as the “core” of a more general (intractable) approach.
The end.

Thanks for your attention...
Additional slides...
Constraints about classes + instances

Correlation:
\{<C1,RS1,LT1>,<C2,RS2,LT2>,...,<Ck,RSk,LTk>,<C1,LT1,RP1>,<C2,LT2,RP2>,...,<Ck,LTk,RPk>\}

In order to check the temporal consistency:

- Which actions inherit constraints? \( \Rightarrow \) correlation
- Semantic assumptions (Complete observation? Future events?)
- Prediction
THE ROLE OF PREDICTION

Temporal constraints on classes have a PREDICTIVE role
(e.g., Reservation 1-7 days BEFORE Lab-Test)

TEMPORAL CONSISTENCY CHECKING:
what happens if a “predicted” instance is not observed within the
“predicted” range of time?

INCOMPLETE OBSERVATIONS $\rightarrow$ OK (prediction not needed)

COMPLETE OBSERVATIONS $\rightarrow$ Inconsistency (prediction needed)
Constraints about classes + instances
+ composite + repeated/periodic

- Which actions inherit constraints? ⇒ association with periodicity

E.g.: “twice each day A (before B, ... C, .... D)”

- Inheritance: periodic patterns (n-ary constraints!)
Constraints about classes + instances + composite + repeated/periodic

- Semantic assumptions (complete observations)

Only “atomic” actions are observed

E.g., “A twice a day”. “A: A1 followed by A2”
Constraints about classes + instances + composite + repeated/periodic

But constraints about composite actions have to be inherited/checked!

E.g., “A twice a day”. “A: A1 followed by A2”.
“A ends at least 1h before start B”

* PROBLEM: intensional vs extensional temporal reasoning (2)
goal of the approach

Knowledge server for (constraint-based) temporal reasoning

Let users abstract from these (general, domain-independent) problems

⇒ solve these problems once and for-all

⇒ “hide” complexity
strategy of the approach

**LAYERED APPROACH**

HIGH LEVEL: interface (expressive language + reasoning facilities)

trade-off (expressiveness vs complexity)

“LOW” LEVEL: internal representation + constraint propagation

“standard” approach (STP)
**PROBLEM**

**FILLING THE GAP!!!**

HIGH LEVEL: interface (expressive language + reasoning facilities)

**NOT JUST “MAPPING”:** algo required for coping with
- correlation
- association
- semantic assumptions
- hypothesizing extensions
- inheritance (also periodic patterns)
- prediction

“LOW” LEVEL: internal representation + constraint propagation (STP)
HLL Language (Classes)

CKB = <Classes, Part-of, Constraints>

(1) list of classes (e.g., {Reservation, Lab_Test, ….})

(2) list of “part-of” relations

Constraints (ex.1): delay
C_{delay}(\text{Start}(C1), \text{Start}(C2), L1, U1) ⇔
(∀ C1', C2' (\text{Instance}_of(C1', C1) \land \text{Instance}_of(C2', C2) \land \text{COR}(C1', C2')) ⇒
(L1 ≤ \text{Start}(C2') - \text{Start}(C1') ≤ U1) \land
(∀ C1' \text{ Instance}_of(C1', C1) ⇒ (∃ C2' \text{ Instance}_of(C2', C2) \land \text{COR}(C1', C2'))))
HLL Language (Instances)

IKB ≡ <(1), (2), (3), (4)>

(1) CONSTRAINTS (examples):
\[ \text{delay}(P_1, P_2, L_1, U_1) \iff L_1 \leq P_2 - P_1 \leq U_1 \]

(2) INSTANCE-OF (e.g., Instance_of(RS1, Reservation))

(3) CORRELATION (e.g., COR(RS1,LT1))

(4) list of instances (e.g., \{RS1, RS2, LT1, ….\})
Repetition constraints

Repetition(C, <R=<nRep, IT, constrs>>) ⇔
(∃ E, E₁, …, E_{nRep} ( % duration %
Cduration(E, IT, IT) ∧
% containment %
∀ i=1, …, nRep (CNSduring(Eᵢ, E) ∧
% ordering %
∀ i=1, …, nRep-1 (CAfter(Eᵢ₊₁, Eᵢ)) ∧
% constraints in constrs must hold % … ∧
% instances %
(∃ I₁, …, IₙRep (Instance_of(Iᵢ, C) ∧ AllDifferent(I₁, …, IₙRep))))))
Repetition constraints

\[
\text{\% constrs \%}
\]

\[\text{inBetweenAll}(\text{minIBA, maxIBA}) \in \text{constrs} \Rightarrow (\forall i=1, \ldots, n\text{Rep} \ (\text{CDelay}(\text{End}(E_i), \text{Start}(E_{i+1}), \text{minIBA, maxIBA}))) \land
\]

\[\text{inBetween}((\text{minIB}_1, \text{maxIB}_1), \ldots, (\text{minIB}_{n\text{Rep}-1}, \text{maxIB}_{n\text{Rep}-1})) \in \text{constrs} \Rightarrow (\text{CDelay}(\text{End}(E_1), \text{Start}(E_1), \text{minIB}_1, \text{maxIB}_1) \land \ldots \land \text{CDelay}(\text{End}(E_{n\text{Rep}-1}), \text{Start}(E_{n\text{Rep}}, \text{minIB}_{n\text{Rep}}, \text{maxIB}_{n\text{Rep}-1})) \land
\]

\[\text{fromStart}(\text{minS, maxS}) \in \text{constrs} \Rightarrow \text{CDelay}((\text{Start}(E), \text{Start}(E_1), \text{minS, maxS})) \land
\]

\[\text{toEnd}(\text{minE, maxE}) \in \text{constrs} \Rightarrow \text{CDelay}(\text{End}(E_{n\text{Rep}}), \text{End}(E), \text{minE, maxE})) \land
\]

\[\ldots
\]