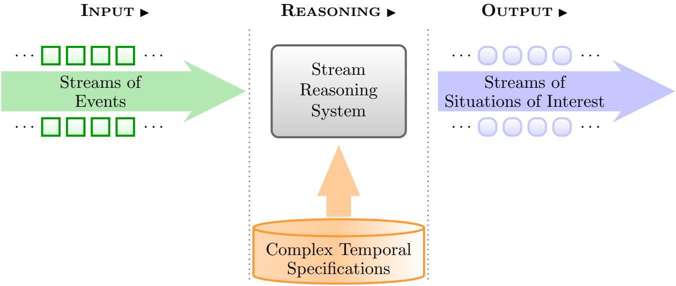
Reasoning over Complex Temporal Specifications and Noisy Data Streams

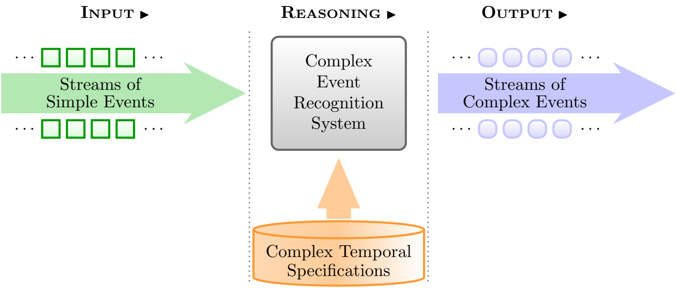
Periklis Mantenoglou

National and Kapodistrian University of Athens, Greece
NCSR Demokritos, Greece

Stream Reasoning



Complex Event Recognition



Requirements and Motivation

Requirements:

Requirements and Motivation

Requirements:

- Temporal specification language:

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Logic-based temporal frameworks:

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Logic-based temporal frameworks:

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Logic-based temporal frameworks:

- Event Calculus:
  - Temporal formalism based on logic programming.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Logic-based temporal frameworks:

- Event Calculus:
  - Temporal formalism based on logic programming.
- Run-Time Event Calculus (RTEC):
  - Event Calculus + windowing and caching

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Logic-based temporal frameworks:

- Event Calculus:
  - Temporal formalism based on logic programming.
- Run-Time Event Calculus (RTEC):
  - Event Calculus + windowing and caching
- Probabilistic Interval-based Event Calculus (PIEC):
  - Event Calculus + noisy events.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Motivation: No framework fulfills all requirements.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Motivation: No framework fulfills all requirements.

Contributions:

- $\text{RTEC}_{\circ}$: RTEC + cyclic dependencies.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Motivation: No framework fulfills all requirements.

Contributions:

- $\text{RTEC}_{\circ}$: RTEC + cyclic dependencies.
- $\text{RTEC}^{\rightarrow}$: RTEC + events with delayed effects.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Motivation: No framework fulfills all requirements.

Contributions:

- RTEC_{o}: RTEC + cyclic dependencies.
- $\text{RTEC}^{\rightarrow}$: RTEC + events with delayed effects.
- RTEC_{A}: RTEC + Allen relations.

Requirements and Motivation

Requirements:

- Temporal specification language:
  - Formal, declarative formalism.
  - Wide range of temporal specifications.
  - Hierarchical and relational patterns.
  - Background Knowledge.
- Highly efficient reasoning algorithms:
  - Windowing.
  - Caching.
- Noisy event streams.

Motivation: No framework fulfills all requirements.

Contributions:

- $\mathsf{RTEC}_{\circ}$: RTEC + cyclic dependencies.
- $\mathsf{RTEC}^{\rightarrow}$: RTEC + events with delayed effects.
- $\mathsf{RTEC}_{\mathsf{A}}$: RTEC + Allen relations.
- oPIEC: PIEC + data streams.

Publications

Journal Publications:

- Mantenoglou P., Pitsikalis M., Artikis A., *Reasoning over Streams of Events with Delayed Effects.*
  In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, **under review since January 2024**.
- Mantenoglou P., Artikis A., Paliouras G., *Online Event Recognition over Noisy Data Streams.*
  In *International Journal of Approximate Reasoning (IJAR)*, 161, 2023.
  DOI: *https://doi.org/10.1016/j.ijar.2023.108993*

Conference Publications:

- Mantenoglou P., Kelesis D., Artikis A., *Complex Event Recognition with Allen Relations.*
  In *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pp. 502–511, 2023.
  DOI: *https://doi.org/10.24963/kr.2023/49*
- Mantenoglou P., Pitsikalis M., Artikis A., *Stream Reasoning with Cycles.*
  In *Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pp. 533–553, 2022.
  DOI: *https://doi.org/10.24963/kr.2022/56*
- Mantenoglou P., Artikis A., Paliouras G., *Online Probabilistic Interval-based Event Calculus.*
  In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI)*, pp. 2624–2631, 2020.
  DOI: *https://doi.org/10.3233/FAIA200399*

Peripheral Publication:

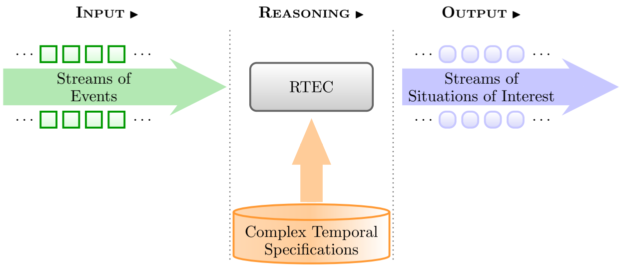
- Andrienko N., Andrienko G., Artikis A., Mantenoglou P., Rinzivillo S., *Human-in-the-Loop: Visual Analytics for Building Models Recognising Behavioural Patterns in Time Series.*
  In *IEEE Computer Graphics and Applications (CG&A)*, pp. 1–15, 2024.
  DOI: *https://doi.org/10.1109/MCG.2024.3379851*

Event Calculus

Predicate	Meaning
happensAt(E, T)	Event E occurs at time T
initiatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is initiated
terminatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is terminated
holdsAt$(F = V, T)$	The value of fluent F is V at time T
holdsFor$(F = V, I)$	I is the list of the maximal intervals for which $F = V$ holds continuously

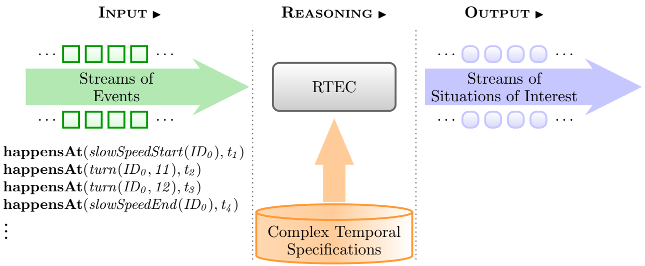
Kowalski et al., A Logic-based Calculus of Events. New Gener. Comput. 4(1): 67–95, 1986.

Stream Reasoning with RTEC



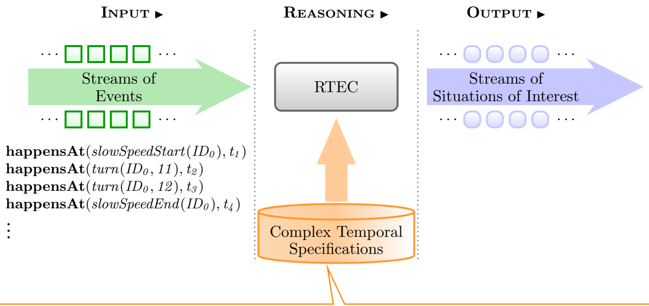
Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Stream Reasoning with RTEC



Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Stream Reasoning with RTEC



$\mathbf{initiatedAt}(stopped(Vessel) = nearPorts, T) \leftarrow$
$\quad \mathbf{happensAt}(stop\_start(Vessel), T),$
$\quad \mathbf{holdsAt}(withinArea(Vessel, nearPorts) = \mathsf{true}, T).$

$\mathbf{terminatedAt}(stopped(Vessel) = nearPorts, T) \leftarrow$
$\quad \mathbf{happensAt}(stop\_end(Vessel), T).$

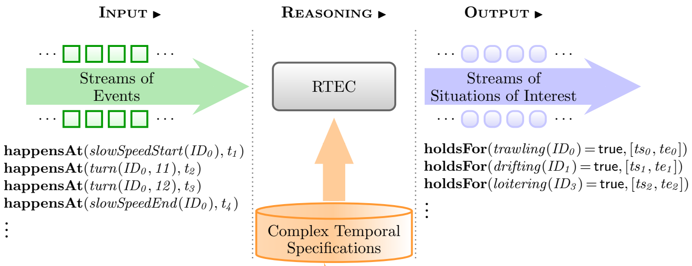
⋮

$\mathbf{holdsFor}(trawling(Vessel) = \mathsf{true}, I) \leftarrow$
$\quad \mathbf{holdsFor}(trawlingMovement(Vessel) = \mathsf{true}, I_{tc}),$
$\quad \mathbf{holdsFor}(trawlingSpeed(Vessel) = \mathsf{true}, I_t),$
$\quad \mathbf{intersect\_all}([I_{tc}, I_t], I_i),$
$\quad threshold(v_{trawl}, V_{trawl}),$
$\quad intDurGreater(I_i, V_{trawl}, I).$

⋮

Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Stream Reasoning with RTEC



$\textbf{initiatedAt}(stopped(Vessel) = nearPorts, T) \leftarrow$
$\quad \textbf{happensAt}(stop\_start(Vessel), T),$
$\quad \textbf{holdsAt}(withinArea(Vessel, nearPorts) = \mathsf{true}, T).$

$\textbf{terminatedAt}(stopped(Vessel) = nearPorts, T) \leftarrow$
$\quad \textbf{happensAt}(stop\_end(Vessel), T).$

⋮

$\textbf{holdsFor}(trawling(Vessel) = \mathsf{true}, I) \leftarrow$
$\quad \textbf{holdsFor}(trawlingMovement(Vessel) = \mathsf{true}, I_{tc}),$
$\quad \textbf{holdsFor}(trawlingSpeed(Vessel) = \mathsf{true}, I_t),$
$\quad \textbf{intersect\_all}([I_{tc}, I_t], I_i),$
$\quad threshold(v_{trawl}, V_{trawl}),$
$\quad intDurGreater(I_i, V_{trawl}, I).$

⋮

Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Simple Fluent: High Speed Near Coast

$$
\begin{aligned}
&\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}, \_\mathit{CoG}, \_\mathit{TrueHeading}), T), \\
&\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}, T), \\
&\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} > V.
\end{aligned}
$$

Simple Fluent: High Speed Near Coast

$$\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}, \_\mathit{CoG}, \_\mathit{TrueHeading}), T),$$
$$\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}, T),$$
$$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} > V.$$

$$\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}), T),$$
$$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} \leq V.$$

$$\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{end}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}), T).$$

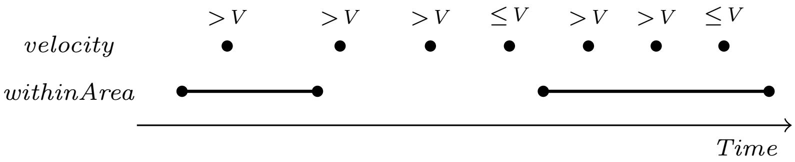
Simple Fluent: High Speed Near Coast

initiatedAt(*highSpeedNC*(*Vessel*) = true, T) $\leftarrow$
  happensAt(*velocity*(*Vessel*, *Speed*, *_CoG*, *_TrueHeading*), T),
  holdsAt(*withinArea*(*Vessel*, *nearCoast*) = true, T),
  threshold(v_{hs}, V), *Speed* $> V$.

terminatedAt(*highSpeedNC*(*Vessel*) = true, T) $\leftarrow$
  happensAt(*velocity*(*Vessel*, *Speed*), T),
  threshold(v_{hs}, V), *Speed* $\leq V$.

terminatedAt(*highSpeedNC*(*Vessel*) = true, T) $\leftarrow$
  happensAt(*end*(*withinArea*(*Vessel*, *nearCoast*) = true), T).

Reasoning: **holdsFor**(*highSpeedNC*(*Vessel*) = true, I)



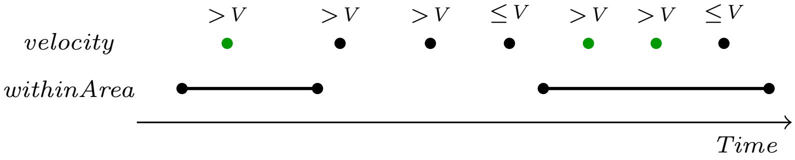
Simple Fluent: High Speed Near Coast

$\mathbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathrm{true},\ T) \leftarrow$
$\quad \mathbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}, \_\mathit{CoG}, \_\mathit{TrueHeading}), T),$
$\quad \mathbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \mathrm{true},\ T),$
$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} > V.$

$\mathbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathrm{true},\ T) \leftarrow$
$\quad \mathbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}), T),$
$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} \leq V.$

$\mathbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathrm{true},\ T) \leftarrow$
$\quad \mathbf{happensAt}(\mathit{end}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \mathrm{true}), T).$

Reasoning: $\mathbf{holdsFor}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathrm{true},\ I)$



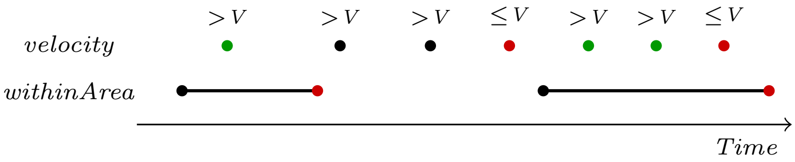
Simple Fluent: High Speed Near Coast

$\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}, \_CoG, \_TrueHeading), T),$
$\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}, T),$
$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} > V.$

$\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}), T),$
$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} \leq V.$

$\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{end}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}), T).$

Reasoning: $\textbf{holdsFor}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, I)$



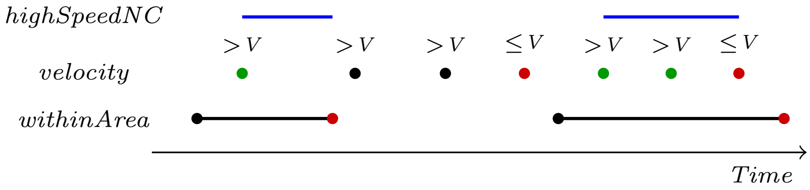
Simple Fluent: High Speed Near Coast

$\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}, \_\mathit{CoG}, \_\mathit{TrueHeading}), T),$
$\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}, T),$
$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} > V.$

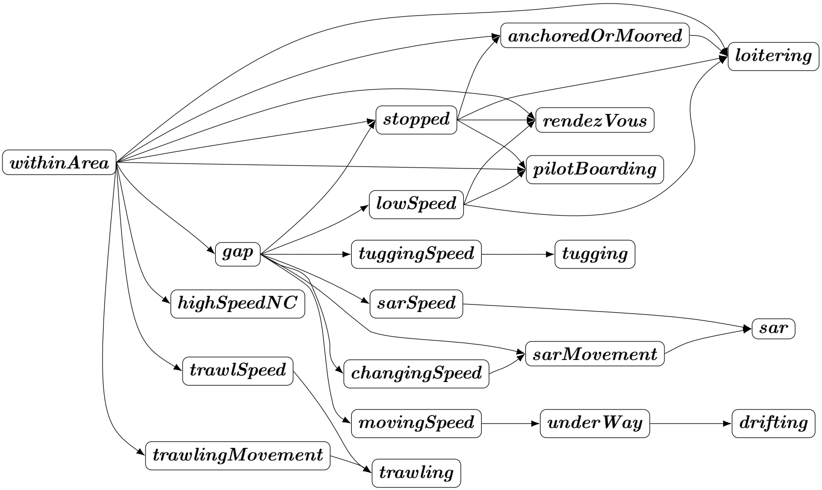
$\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}), T),$
$\quad \mathit{threshold}(v_{hs}, V), \mathit{Speed} \leq V.$

$\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{end}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}), T).$

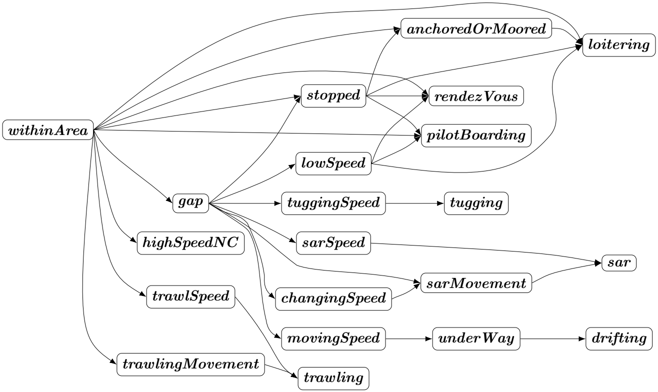
Reasoning: $\textbf{holdsFor}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true}, I)$



Hierarchical Knowledge Bases



Hierarchical Knowledge Bases

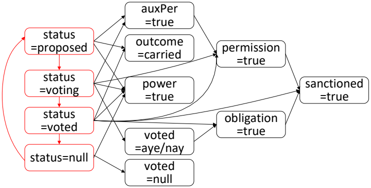


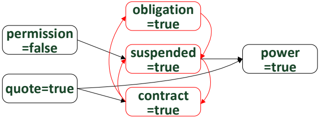
Semantics

An event description of RTEC is a locally stratified logic program.

Cyclic Dependencies in Temporal Patterns

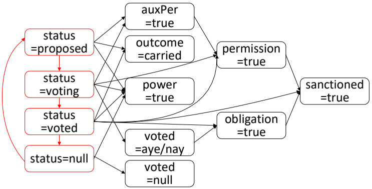
- Multi-Agent Systems: Voting & NetBill.

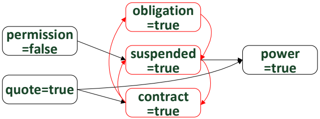




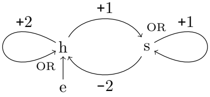
Cyclic Dependencies in Temporal Patterns

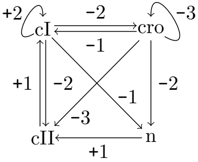
- Multi-Agent Systems: Voting & NetBill.





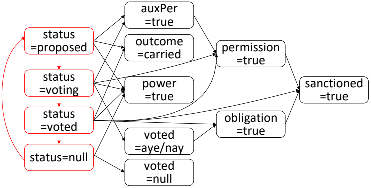
- Biological Feedback Processes.

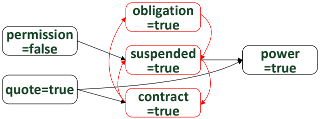




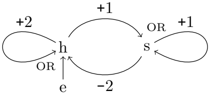
Cyclic Dependencies in Temporal Patterns

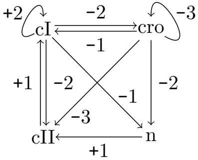
- Multi-Agent Systems: Voting & NetBill.





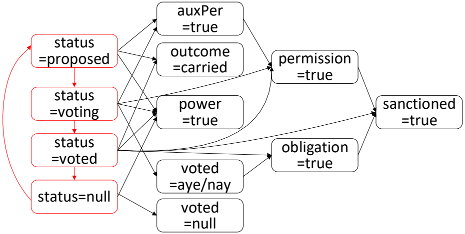
- Biological Feedback Processes.





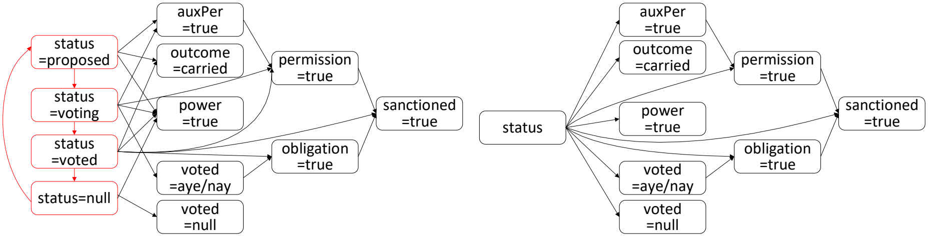
- Maritime Situational Awareness: the stages of a fishing trip, i.e., *started*, *fishing*, *returning*, *ended*, form a cycle.

1. RTEC$_o$: Cyclic Dependencies



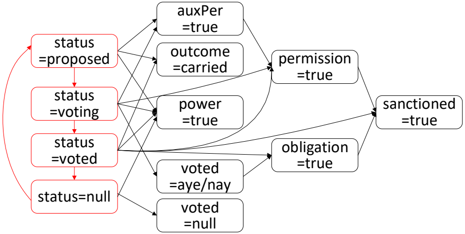
Mantenoglou et al., Stream Reasoning with Cycles. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 544–553, 2022.

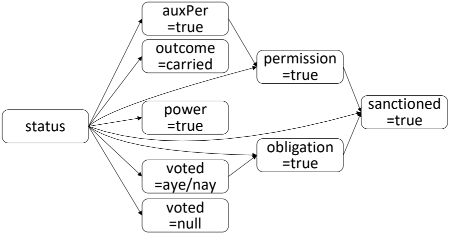
1. RTEC$_o$: Cyclic Dependencies

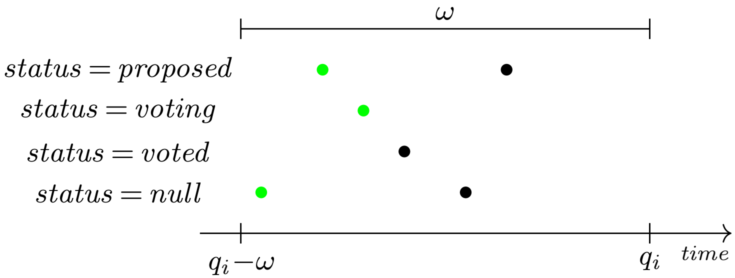


Mantenoglou et al., Stream Reasoning with Cycles. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 544–553, 2022.

1. RTEC$_\circ$: Cyclic Dependencies

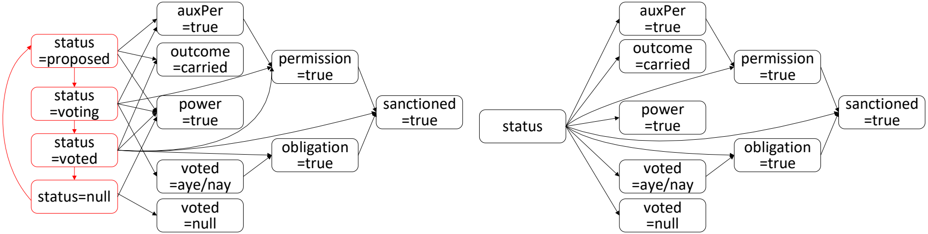


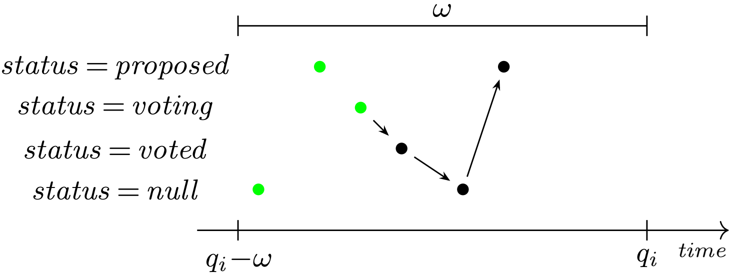




Mantenoglou et al., Stream Reasoning with Cycles. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 544–553, 2022.

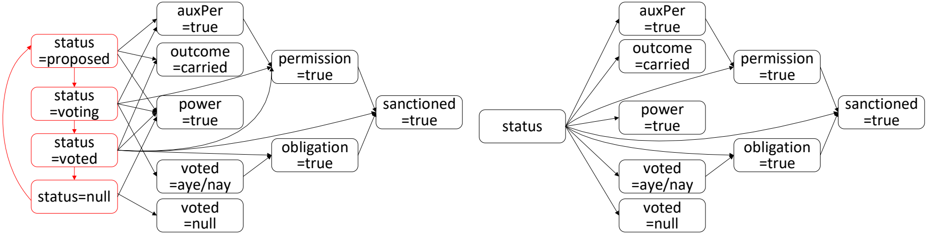
1. RTEC$_o$: Cyclic Dependencies

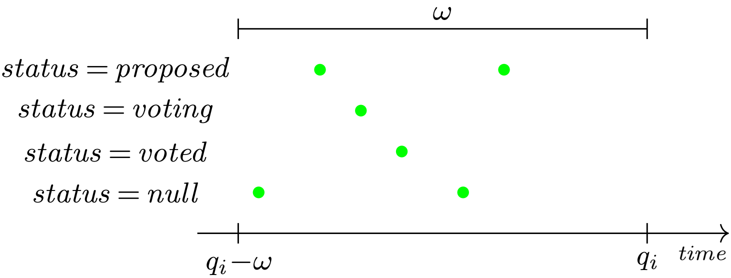




Mantenoglou et al., Stream Reasoning with Cycles. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 544–553, 2022.

1. RTEC$_\circ$: Cyclic Dependencies





Mantenoglou et al., Stream Reasoning with Cycles. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 544–553, 2022.

1. $\mathrm{RTEC}_{\mathrm{o}}$: Formal Properties

Semantics

An event description of $\mathrm{RTEC}_{\mathrm{o}}$ is a locally stratified logic program.

1. $RTEC_o$: Formal Properties

Semantics

An event description of $RTEC_o$ is a locally stratified logic program.

Correctness

$RTEC_o$ computes all maximal intervals of the fluents of an event description with cyclic dependencies, and no other interval.

1. $RTEC_o$: Formal Properties

Semantics

An event description of $RTEC_o$ is a locally stratified logic program.

Correctness

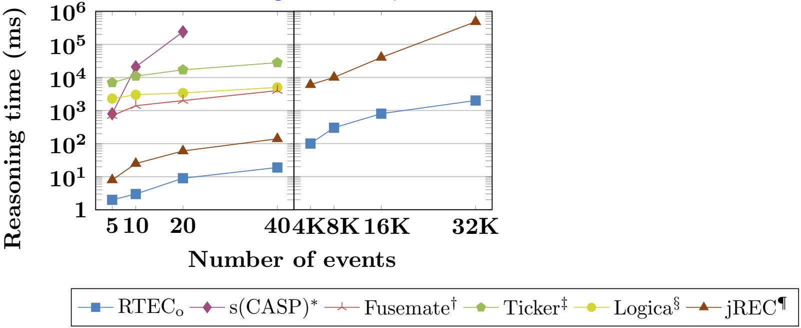
$RTEC_o$ computes all maximal intervals of the fluents of an event description with cyclic dependencies, and no other interval.

Complexity

In $RTEC_o$, the worst-case time complexity of maximal interval computation for a fluent definition with cyclic dependencies is $\mathcal{O}(\omega log(\omega))$, where ω is the size of the window.

1. RTEC$_o$: Indicative Experimental Results

NetBill: monitoring active quotes



*Arias et al., Modeling and reasoning in event calculus using goal-directed constraint answer set programming. Theory and Practice of Logic Programming, 2022.

†Baumgartner, Combining Event Calculus and Description Logic Reasoning via Logic Programming. Frontiers of Combining Systems, 2021.

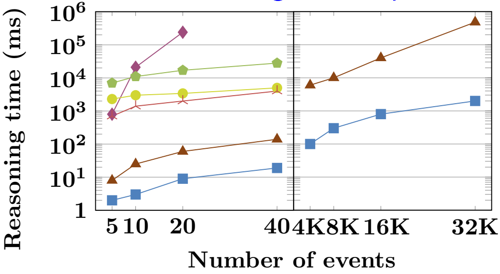
‡Beck et al., Ticker: A system for incremental asp-based stream reasoning. Theory and Practice of Logic Programming, 2017.

§Logica: Language of Big Data, `https://github.com/EvgSkv/logica`.

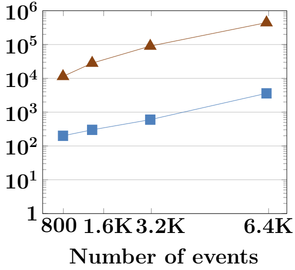
¶Falcionelli et al., Indexing the Event Calculus: Towards practical human-readable Personal Health Systems. Artificial Intelligence in Medicine, 2019.

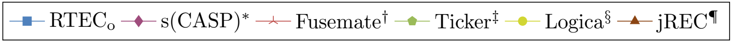
1. RTEC$_o$: Indicative Experimental Results

NetBill: monitoring active quotes



Voting: monitoring the status of motions (cycles)





*Arias et al., Modeling and reasoning in event calculus using goal-directed constraint answer set programming. Theory and Practice of Logic Programming, 2022.

†Baumgartner, Combining Event Calculus and Description Logic Reasoning via Logic Programming. Frontiers of Combining Systems, 2021.

‡Beck et al., Ticker: A system for incremental asp-based stream reasoning. Theory and Practice of Logic Programming, 2017.

§Logica: Language of Big Data, `https://github.com/EvgSkv/logica`.

¶Falcionelli et al., Indexing the Event Calculus: Towards practical human-readable Personal Health Systems. Artificial Intelligence in Medicine, 2019.

Events with Delayed Effects

- Biological Feedback Processes: the values of the biological variables in a feedback loop change with time delays.

Events with Delayed Effects

- Biological Feedback Processes: the values of the biological variables in a feedback loop change with time delays.
- Maritime Situational Awareness: a fishing activity is terminated with a delay after multiple changes in heading.

Events with Delayed Effects

- Biological Feedback Processes: the values of the biological variables in a feedback loop change with time delays.
- Maritime Situational Awareness: a fishing activity is terminated with a delay after multiple changes in heading.
- Multi-Agent Systems: agents may be suspended temporarily. Further violations may extend the period of suspension.

Events with Delayed Effects

- Biological Feedback Processes: the values of the biological variables in a feedback loop change with time delays.
- Maritime Situational Awareness: a fishing activity is terminated with a delay after multiple changes in heading.
- Multi-Agent Systems: agents may be suspended temporarily. Further violations may extend the period of suspension.

- RTEC supports only immediate initiations:

$\textbf{initiatedAt}(F = V, T) \leftarrow$
$\quad \textbf{happensAt}(E, T)[,$
$\quad \text{conditions}].$

where conditions:
$^{0-K}$[not] $\textbf{happensAt}(E_k, T)$,
$^{0-M}$[not] $\textbf{holdsAt}(F_m = V_m, T)$,
$^{0-N}$atemporal-constraint$_n$.

2. RTEC$^{\rightarrow}$: Events with Delayed Effects

- RTEC$^{\rightarrow}$: Representation of future initiations:
  - $\mathbf{fi}(quote(M, C, G) = in\_effect, quote(M, C, G) = expiring, 50)$.

Mantenoglou et al., Reasoning over streams of events with delayed effects. In IEEE Transactions on Knowledge and Data Engineering (TKDE), **under review since January 2024**.

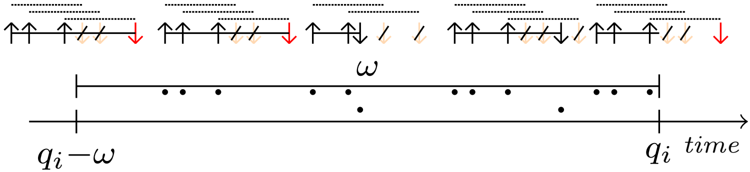
2. RTEC$^{\rightarrow}$: Events with Delayed Effects

- RTEC$^{\rightarrow}$: Representation of future initiations:
  - $\mathbf{fi}(\mathit{quote}(M, C, G) = \mathit{in\_effect}, \mathit{quote}(M, C, G) = \mathit{expiring}, 50)$.
  - $\mathbf{p}(\mathit{quote}(M, C, G) = \mathit{in\_effect})$.

Mantenoglou et al., Reasoning over streams of events with delayed effects. In IEEE Transactions on Knowledge and Data Engineering (TKDE), **under review since January 2024**.

2. RTEC$^{\rightarrow}$: Events with Delayed Effects

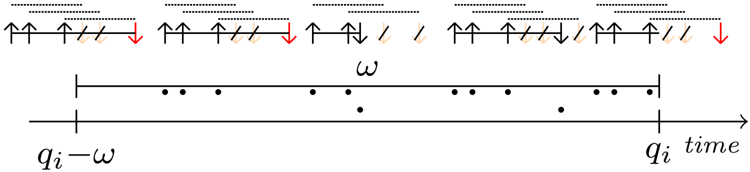
- RTEC$^{\rightarrow}$: Representation of future initiations:
  - $\mathbf{fi}(quote(M, C, G) = in\_effect, quote(M, C, G) = expiring, 50)$.
  - $\mathbf{p}(quote(M, C, G) = in\_effect)$.



Mantenoglou et al., Reasoning over streams of events with delayed effects. In IEEE Transactions on Knowledge and Data Engineering (TKDE), **under review since January 2024**.

2. RTEC$^{\rightarrow}$: Events with Delayed Effects

- RTEC$^{\rightarrow}$: Representation of future initiations:
  - $\textbf{fi}(quote(M, C, G) = in\_effect, quote(M, C, G) = expiring, 50)$.
  - $\textbf{p}(quote(M, C, G) = in\_effect)$.

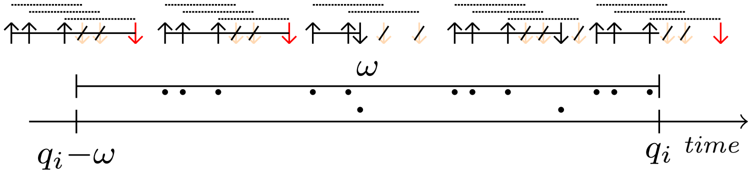


- RTEC$^{\rightarrow}$: Reasoning over future initiations:
  - Compile-time detection of optimal processing order.

Mantenoglou et al., Reasoning over streams of events with delayed effects. In IEEE Transactions on Knowledge and Data Engineering (TKDE), **under review since January 2024**.

2. RTEC$^{\rightarrow}$: Events with Delayed Effects

- RTEC$^{\rightarrow}$: Representation of future initiations:
  - **fi**($quote(M, C, G) = in\_effect, quote(M, C, G) = expiring, 50$).
  - **p**($quote(M, C, G) = in\_effect$).

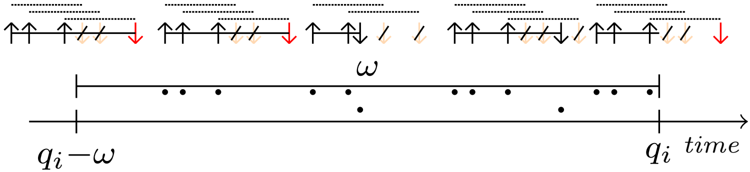


- RTEC$^{\rightarrow}$: Reasoning over future initiations:
  - Compile-time detection of optimal processing order.
  - Incremental caching in ascending temporal order at run-time.

Mantenoglou et al., Reasoning over streams of events with delayed effects. In IEEE Transactions on Knowledge and Data Engineering (TKDE), **under review since January 2024**.

2. RTEC$^{\rightarrow}$: Events with Delayed Effects

- RTEC$^{\rightarrow}$: Representation of future initiations:
  - $\textbf{fi}(quote(M, C, G) = in\_effect, quote(M, C, G) = expiring, 50)$.
  - $\textbf{p}(quote(M, C, G) = in\_effect)$.



- RTEC$^{\rightarrow}$: Reasoning over future initiations:
  - Compile-time detection of optimal processing order.
  - Incremental caching in ascending temporal order at run-time.
  - Minimal information transfer between windows.

Mantenoglou et al., Reasoning over streams of events with delayed effects. In IEEE Transactions on Knowledge and Data Engineering (TKDE), **under review since January 2024**.

2. RTEC$^{\rightarrow}$: Formal Properties

Semantics

An event description of RTEC$^{\rightarrow}$ is a locally stratified logic program.

2. RTEC$^{\rightarrow}$: Formal Properties

Semantics

An event description of RTEC$^{\rightarrow}$ is a locally stratified logic program.

Correctness

RTEC$^{\rightarrow}$ computes all maximal intervals of the fluents of an event description with events with delayed effects, and no other interval.

2. RTEC$^{\rightarrow}$: Formal Properties

Semantics

An event description of RTEC$^{\rightarrow}$ is a locally stratified logic program.

Correctness

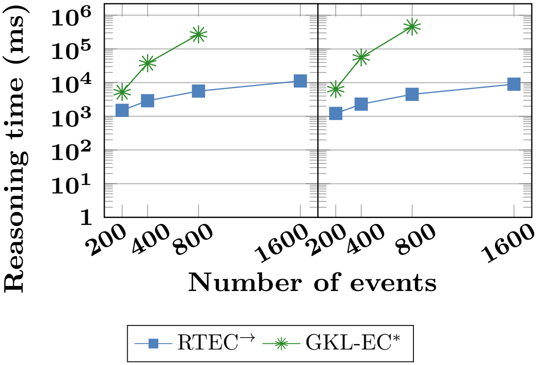
RTEC$^{\rightarrow}$ computes all maximal intervals of the fluents of an event description with events with delayed effects, and no other interval.

Complexity

In RTEC$^{\rightarrow}$, the worst-case time complexity of maximal interval computation for a fluent definition with events with delayed effects is $\mathcal{O}(\omega log(\omega))$, where ω is the size of the window.

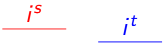
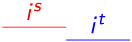
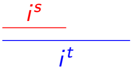
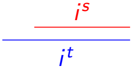
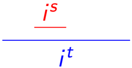
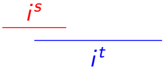
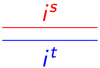
2. RTEC$^{\rightarrow}$: Indicative Experimental Results

Biological Processes: Immune Response and Phage Infection (delayed effects & cycles)



*Srinivasan et al., Learning explanations for biological feedback with delays using an event calculus. Machine Learning, 2022.

The Relations of Allen's Interval Algebra

Relation	**Illustration**
before(i^s, i^t)	
meets(i^s, i^t)	
starts(i^s, i^t)	
finishes(i^s, i^t)	
during(i^s, i^t)	
overlaps(i^s, i^t)	
equal(i^s, i^t)	

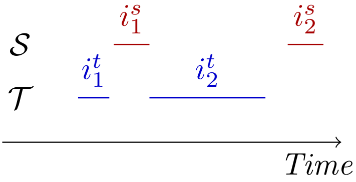
3. RTEC$_A$: Allen Relations

$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{disappearedInArea}(\mathit{Vessel}, \mathit{AreaType}) = \mathrm{true}, I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{AreaType}) = \mathrm{true}, \mathcal{S}), \\
&\quad \textbf{holdsFor}(\mathit{gap}(\mathit{Vessel}) = \mathit{farFromPorts}, \mathcal{T}), \\
&\quad \textbf{allen}(\mathrm{meets}, \mathcal{S}, \mathcal{T}, \mathrm{target}, I).
\end{aligned}
$$

Mantenoglou et al., Complex Event Recognition with Allen Relations. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 502–511, 2023.

3. RTEC$_A$: Allen Relations

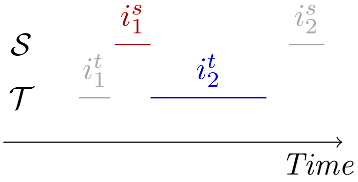
holdsFor(*disappearedInArea*(*Vessel*, *AreaType*) = true, *I*) $\leftarrow$
holdsFor(*withinArea*(*Vessel*, *AreaType*) = true, $\mathcal{S}$),
holdsFor(*gap*(*Vessel*) = *farFromPorts*, $\mathcal{T}$),
allen(meets, $\mathcal{S}$, $\mathcal{T}$, target, *I*).



Mantenoglou et al., Complex Event Recognition with Allen Relations. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 502–511, 2023.

3. RTEC$_A$: Allen Relations

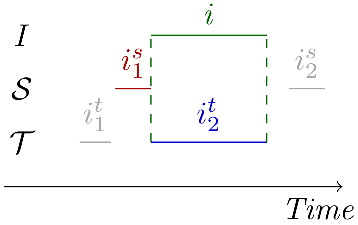
$\textbf{holdsFor}(\textit{disappearedInArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, I) \leftarrow$
$\quad \textbf{holdsFor}(\textit{withinArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, \mathcal{S}),$
$\quad \textbf{holdsFor}(\textit{gap}(\textit{Vessel}) = \textit{farFromPorts}, \mathcal{T}),$
$\quad \textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$



Mantenoglou et al., Complex Event Recognition with Allen Relations. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 502–511, 2023.

3. RTEC_A: Allen Relations

holdsFor(*disappearedInArea*(*Vessel*, *AreaType*) = true, *I*) ←
  holdsFor(*withinArea*(*Vessel*, *AreaType*) = true, $\mathcal{S}$),
  holdsFor(*gap*(*Vessel*) = *farFromPorts*, $\mathcal{T}$),
  allen(meets, $\mathcal{S}$, $\mathcal{T}$, target, *I*).



Mantenoglou et al., Complex Event Recognition with Allen Relations. In International Conference on Principles of Knowledge Representation and Reasoning (KR), 502–511, 2023.

3. $RTEC_A$: Formal Properties

Semantics

An event description of $RTEC_A$ is a locally stratified logic program.

3. RTEC_A: Formal Properties

Semantics

An event description of RTEC_A is a locally stratified logic program.

Correctness

RTEC_A computes all maximal intervals of the fluents of an event description with Allen relations, and no other interval.

3. RTEC_A: Formal Properties

Semantics

An event description of RTEC_A is a locally stratified logic program.

Correctness

RTEC_A computes all maximal intervals of the fluents of an event description with Allen relations, and no other interval.

Complexity

In RTEC_A, the worst-case time complexity of maximal interval computation for a fluent definition with Allen relations $\mathcal{O}(\omega)$, where ω is the size of the window.

3. $RTEC_A$: Indicative Experimental Results

Monitoring maritime activities with Allen relations

Window size		Reasoning Time (ms)		Output Intervals	
Days	Input Intervals	$RTEC_A$	D^2IA^*	$RTEC_A$	D^2IA^*
1	19K	40	410	6K	6K
2	37K	65	592	9K	9K
4	74K	99	1.1K	16K	16K
8	148K	156	1.6K	32K	31K
16	297K	285	2.7K	77K	76K

*Awad et al, D^2IA: User-defined interval analytics on distributed streams. Information Systems, 2022.

3. RTEC_A: Indicative Experimental Results

Monitoring maritime activities with Allen relations

Window size		Reasoning Time (ms)		Output Intervals	
Days	Input Intervals	RTEC_A	D^2IA^*	RTEC_A	D^2IA^*
1	19K	40	410	6K	6K
2	37K	65	592	9K	9K
4	74K	99	1.1K	16K	16K
8	148K	156	1.6K	32K	31K
16	297K	285	2.7K	77K	76K

*Awad et al, D^2IA: User-defined interval analytics on distributed streams. Information Systems, 2022.

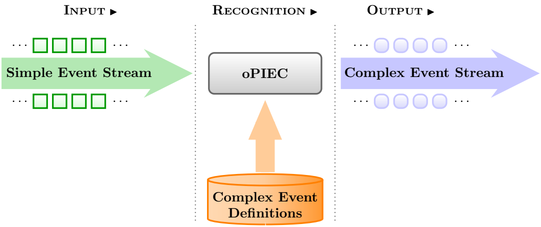
3. RTEC_A: Indicative Experimental Results

Monitoring maritime activities with Allen relations

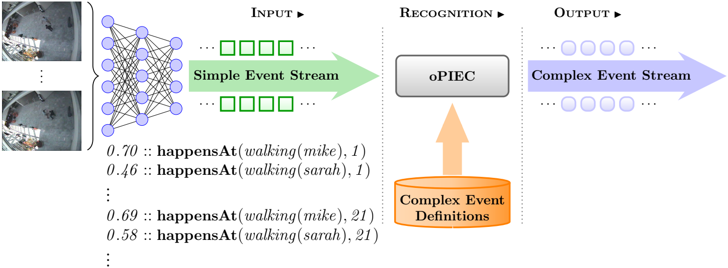
Window size		Reasoning Time (ms)		Output Intervals	
Days	Input Intervals	RTEC_A	D^2IA^*	RTEC_A	D^2IA^*
1	19K	40	410	6K	6K
2	37K	65	592	9K	9K
4	74K	99	1.1K	16K	16K
8	148K	156	1.6K	32K	31K
16	297K	285	2.7K	77K	76K

* Awad et al, D^2IA: User-defined interval analytics on distributed streams. Information Systems, 2022.

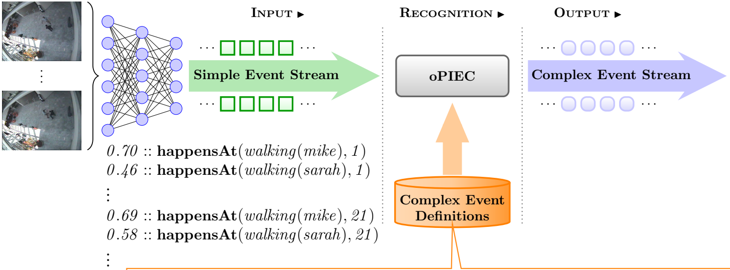
4. Stream Reasoning with oPIEC



4. Stream Reasoning with oPIEC



4. Stream Reasoning with oPIEC



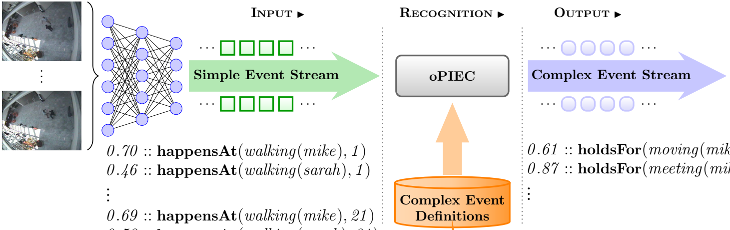
$\textbf{initiatedAt}(moving(P_1, P_2) = \mathsf{true},\ T) \leftarrow$
$\quad \textbf{happensAt}(walking(P_1),\ T),$
$\quad \textbf{happensAt}(walking(P_2),\ T),$
$\quad \textbf{holdsAt}(close(P_1, P_2) = \mathsf{true},\ T),$
$\quad \textbf{holdsAt}(orientation(P_1, P_2) = \mathsf{true},\ T).$

⋮

$\textbf{terminatedAt}(moving(P_1, P_2) = \mathsf{true},\ T) \leftarrow$
$\quad \textbf{happensAt}(walking(P_1),\ T),$
$\quad \textbf{holdsAt}(close(P_1, P_2) = \mathsf{false},\ T).$

⋮

4. Stream Reasoning with oPIEC



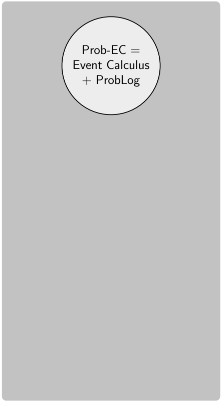
0.70 :: **happensAt**$(walking(mike), 1)$
0.46 :: **happensAt**$(walking(sarah), 1)$
⋮
0.69 :: **happensAt**$(walking(mike), 21)$
0.58 :: **happensAt**$(walking(sarah), 21)$
⋮

0.61 :: **holdsFor**$(moving(mike, sarah) = \text{true}, [2, 18])$
0.87 :: **holdsFor**$(meeting(mike, sarah) = \text{true}, [16, 29])$
⋮

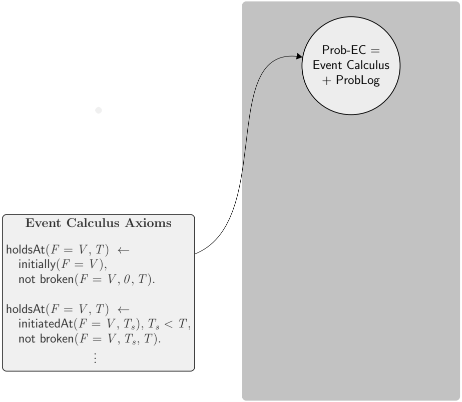
initiatedAt$(moving(P_1, P_2) = \text{true},\ T) \leftarrow$
  happensAt$(walking(P_1),\ T)$,
  happensAt$(walking(P_2),\ T)$,
  holdsAt$(close(P_1, P_2) = \text{true},\ T)$,
  holdsAt$(orientation(P_1, P_2) = \text{true},\ T)$.
⋮

terminatedAt$(moving(P_1, P_2) = \text{true},\ T) \leftarrow$
  happensAt$(walking(P_1),\ T)$,
  holdsAt$(close(P_1, P_2) = \text{false},\ T)$.
⋮

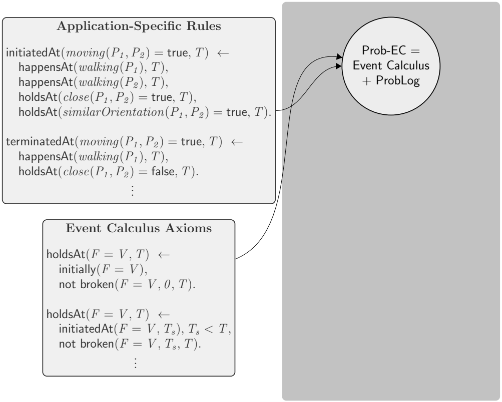
4. Architecture of oPIEC



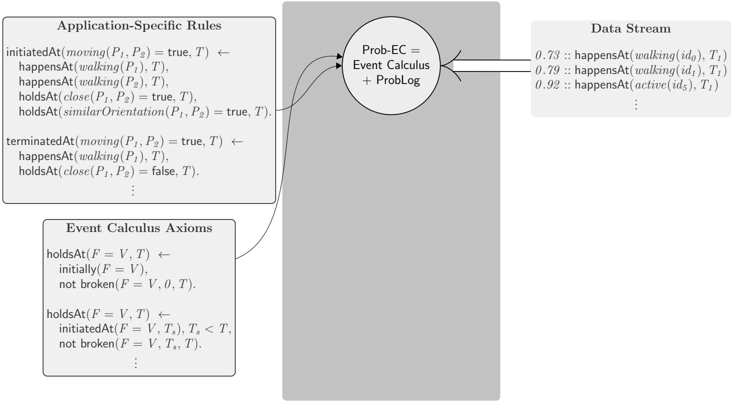
4. Architecture of oPIEC



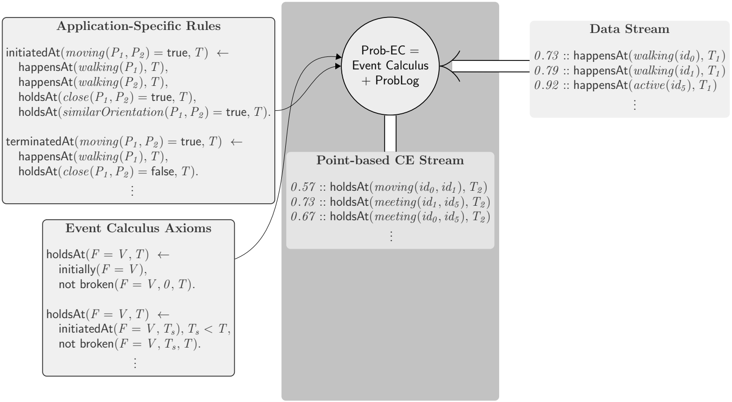
4. Architecture of oPIEC



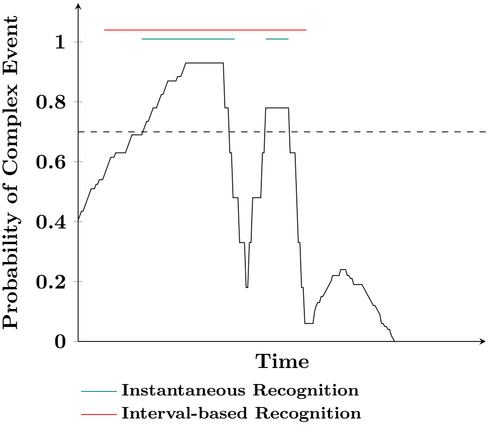
4. Architecture of oPIEC



4. Architecture of oPIEC

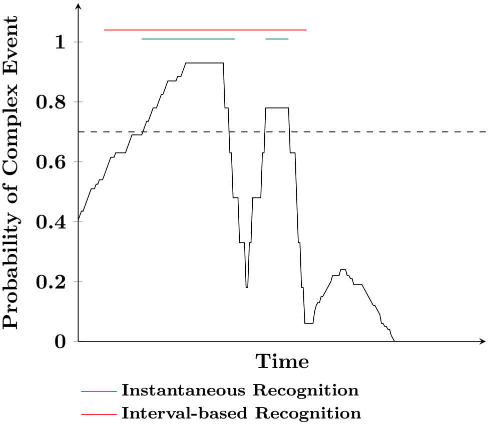


Time-points vs Temporal Intervals



Artikis et al., A Probabilistic Interval-based Event Calculus for Activity Recognition. Annals of Mathematics and Artificial Intelligence, 2021.

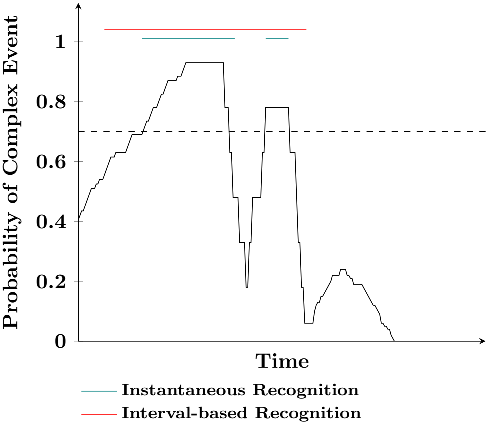
Time-points vs Temporal Intervals



- Interval Probability: average probability of the time-points it contains.

Artikis et al., A Probabilistic Interval-based Event Calculus for Activity Recognition. Annals of Mathematics and Artificial Intelligence, 2021.

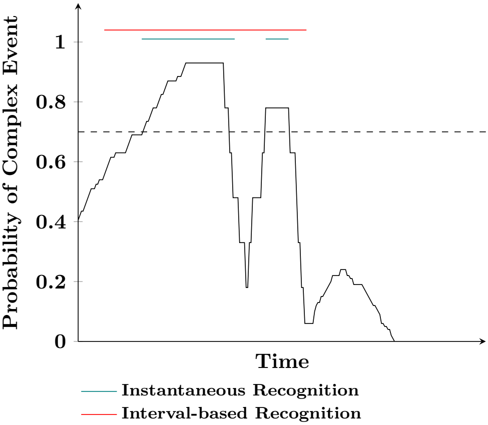
Time-points vs Temporal Intervals



- Interval Probability: average probability of the time-points it contains.
- Probabilistic Maximal Interval:
  - interval probability above a given threshold;
  - no super-interval with probability above the threshold.

Artikis et al., A Probabilistic Interval-based Event Calculus for Activity Recognition. Annals of Mathematics and Artificial Intelligence, 2021.

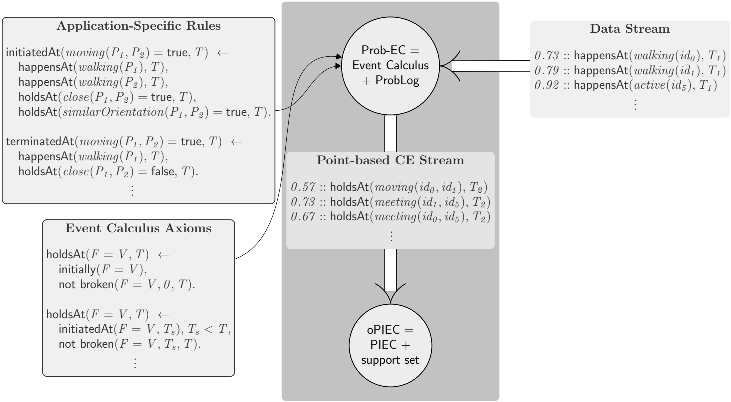
Time-points vs Temporal Intervals



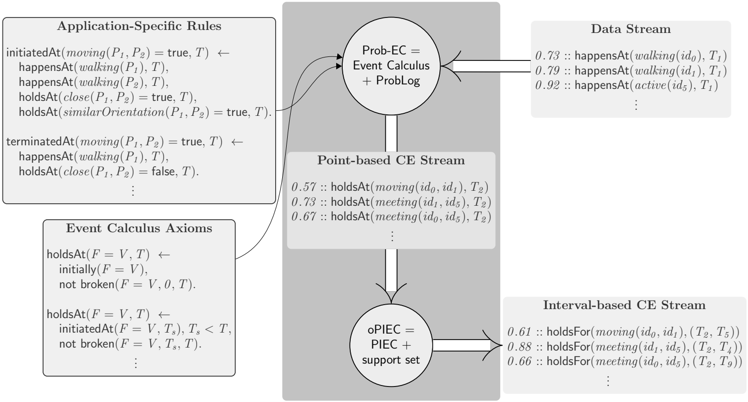
- Interval Probability: average probability of the time-points it contains.
- Probabilistic Maximal Interval:
  - interval probability above a given threshold;
  - no super-interval with probability above the threshold.
- Probabilistic maximal interval computation via maximal non-negative sum interval computation.

Artikis et al., A Probabilistic Interval-based Event Calculus for Activity Recognition. Annals of Mathematics and Artificial Intelligence, 2021.

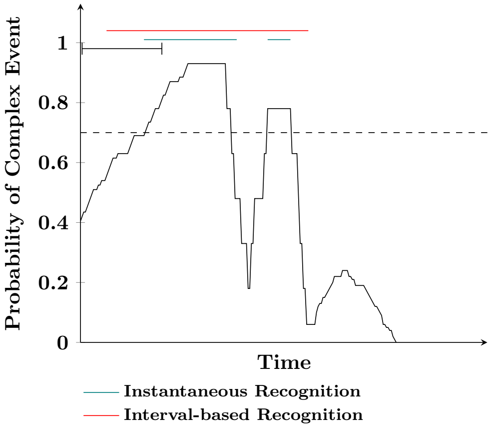
4. Architecture of oPIEC



4. Architecture of oPIEC

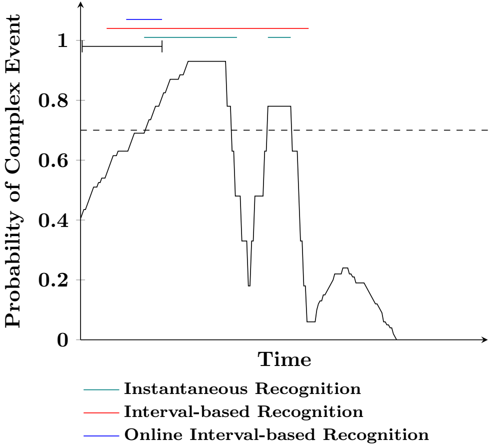


4. Online Interval-based Recognition with oPIEC



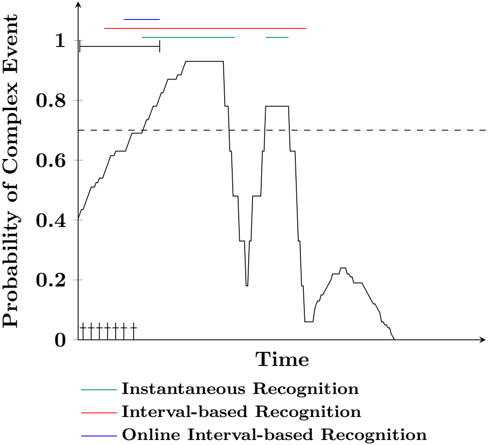
- Windowing.

4. Online Interval-based Recognition with oPIEC



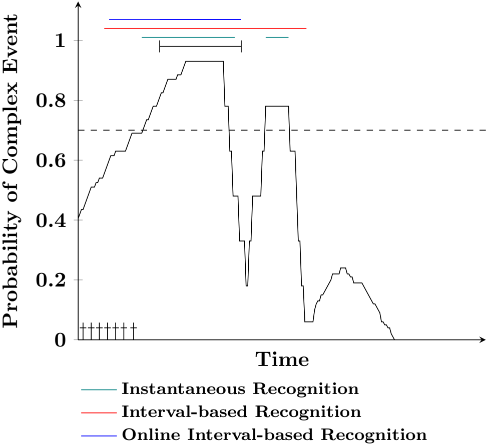
- Windowing.
- Probabilistic maximal interval computation.

4. Online Interval-based Recognition with oPIEC



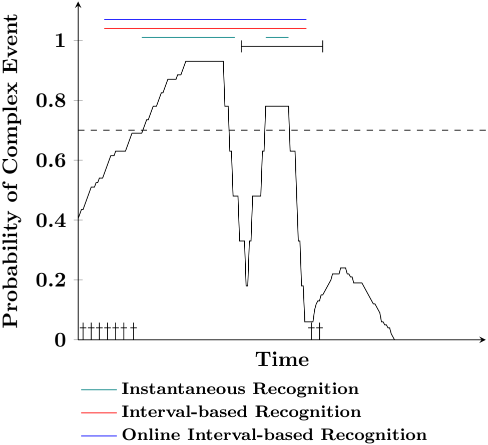
- Windowing.
- Probabilistic maximal interval computation.
- Caching potential starting points.
  - Discard time-point t iff there is a $t' < t$ that can be the starting point of a probabilistic maximal interval including t.

4. Online Interval-based Recognition with oPIEC



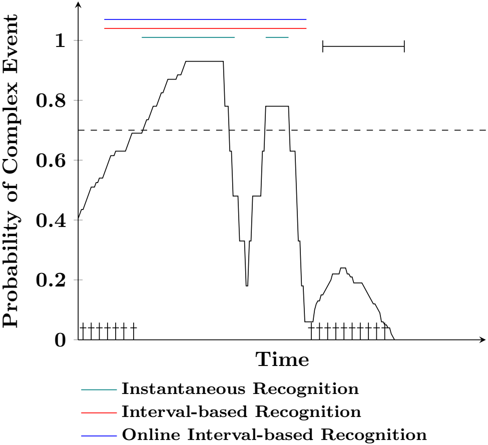
- Windowing.
- Probabilistic maximal interval computation.
- Caching potential starting points.
  - Discard time-point t iff there is a $t' < t$ that can be the starting point of a probabilistic maximal interval including t.

4. Online Interval-based Recognition with oPIEC



- Windowing.
- Probabilistic maximal interval computation.
- Caching potential starting points.
  - Discard time-point t iff there is a $t' < t$ that can be the starting point of a probabilistic maximal interval including t.

4. Online Interval-based Recognition with oPIEC



- Windowing.
- Probabilistic maximal interval computation.
- Caching potential starting points.
  - Discard time-point t iff there is a $t' < t$ that can be the starting point of a probabilistic maximal interval including t.

4. Online Interval-based Recognition with oPIEC: Properties

Memory Minimality

A time-point is cached iff it may be the starting point of a future probabilistic maximal interval.

4. Online Interval-based Recognition with oPIEC: Properties

Memory Minimality

A time-point is cached iff it may be the starting point of a future probabilistic maximal interval.

Interval Computation Correctness

An interval is computed iff it is a probabilistic maximal interval given the data seen so far.

4. Online Interval-based Recognition with oPIEC: Properties

Memory Minimality

A time-point is cached iff it may be the starting point of a future probabilistic maximal interval.

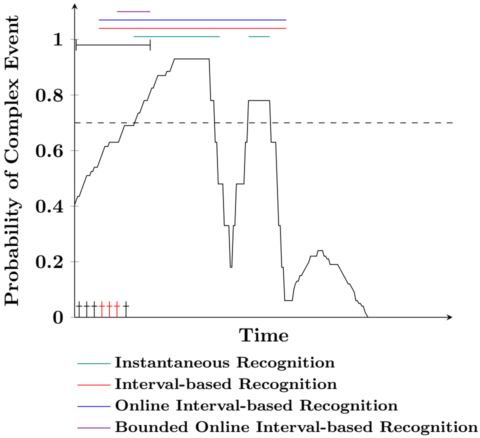
Interval Computation Correctness

An interval is computed iff it is a probabilistic maximal interval given the data seen so far.

Complexity

The computation of probabistic maximal intervals is linear to the window and memory size.

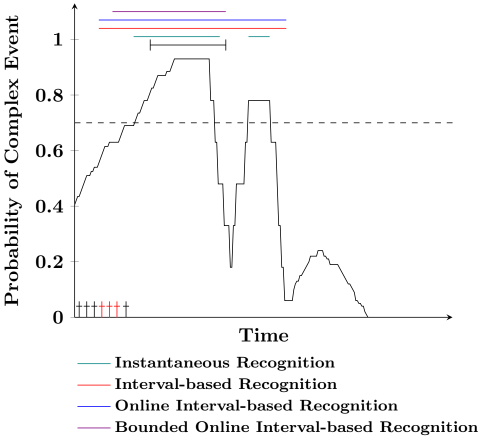
4. Bounded Online Interval-based Recognition with oPIEC



- Complex event duration statistics favor more recent potential starting points.

Mantenoglou et al., Online Event Recognition over Noisy Data Streams. International Journal of Approximate Reasoning, 2023.

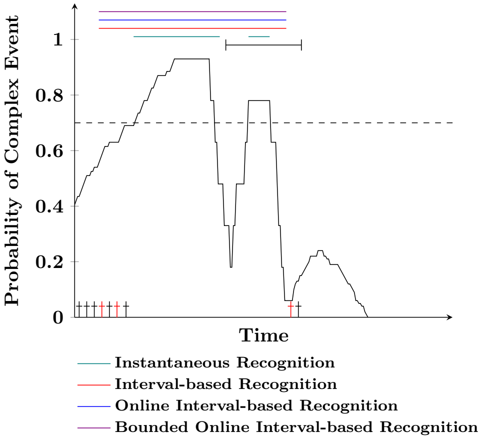
4. Bounded Online Interval-based Recognition with oPIEC



- Complex event duration statistics favor more recent potential starting points.

Mantenoglou et al., Online Event Recognition over Noisy Data Streams. International Journal of Approximate Reasoning, 2023.

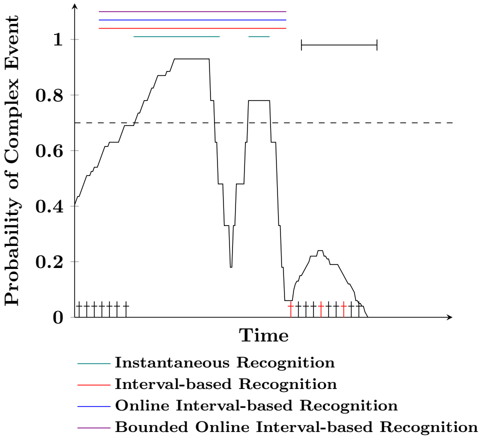
4. Bounded Online Interval-based Recognition with oPIEC



- Complex event duration statistics favor more recent potential starting points.

Mantenoglou et al., Online Event Recognition over Noisy Data Streams. International Journal of Approximate Reasoning, 2023.

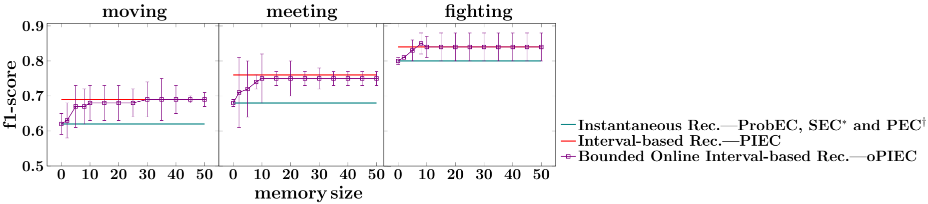
4. Bounded Online Interval-based Recognition with oPIEC

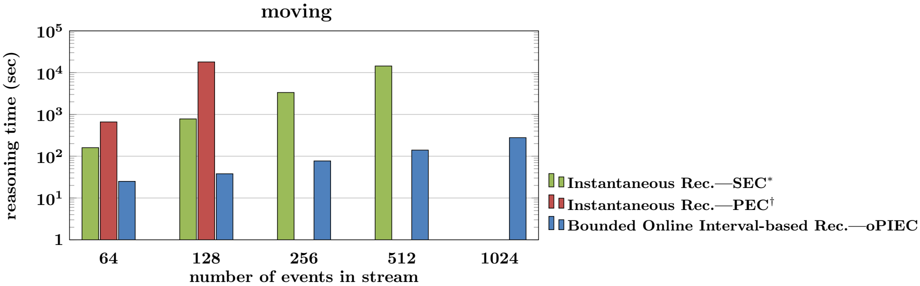


- Complex event duration statistics favor more recent potential starting points.
- Comparable accuracy to batch reasoning.

Mantenoglou et al., Online Event Recognition over Noisy Data Streams. International Journal of Approximate Reasoning, 2023.

4. oPIEC: Indicative Experimental Results





* McAreavey et al., The event calculus in probabilistic logic programming with annotated disjunctions. AAMAS, 2017.

† D'Asaro et al., Probabilistic reasoning about epistemic action narratives. Artificial Intelligence, 2021.

Summary

- Stream Reasoning over Complex Temporal Specifications*:
  - $\text{RTEC}_{\circ}$ supports cyclic dependencies.
  - $\text{RTEC}^{\rightarrow}$ supports events with delayed effects.
  - RTEC_{A} supports Allen relations.

*https://github.com/aartikis/rtec

Summary

- Stream Reasoning over Complex Temporal Specifications*:
  - $\text{RTEC}_{\circ}$ supports cyclic dependencies.
  - $\text{RTEC}^{\rightarrow}$ supports events with delayed effects.
  - RTEC_{A} supports Allen relations.
- Reasoning over Noisy Data Streams†:
  - oPIEC: interval-based reasoning over noisy data streams.

*https://github.com/aartikis/rtec

†https://github.com/periklismant/opiec

Summary

- Stream Reasoning over Complex Temporal Specifications*:
  - $\text{RTEC}_{\circ}$ supports cyclic dependencies.
  - $\text{RTEC}^{\rightarrow}$ supports events with delayed effects.
  - RTEC_{A} supports Allen relations.
- Reasoning over Noisy Data Streams†:
  - oPIEC: interval-based reasoning over noisy data streams.
- Formal Properties.

*https://github.com/aartikis/rtec

†https://github.com/periklismant/opiec

Summary

- Stream Reasoning over Complex Temporal Specifications*:
  - $\text{RTEC}_\circ$ supports cyclic dependencies.
  - $\text{RTEC}^{\rightarrow}$ supports events with delayed effects.
  - RTEC_A supports Allen relations.
- Reasoning over Noisy Data Streams†:
  - oPIEC: interval-based reasoning over noisy data streams.
- Formal Properties.
- Open-source stream reasoning frameworks.

*https://github.com/aartikis/rtec

†https://github.com/periklismant/opiec

Summary

- Stream Reasoning over Complex Temporal Specifications*:
  - $\text{RTEC}_{\circ}$ supports cyclic dependencies.
  - $\text{RTEC}^{\rightarrow}$ supports events with delayed effects.
  - RTEC_{A} supports Allen relations.
- Reasoning over Noisy Data Streams†:
  - oPIEC: interval-based reasoning over noisy data streams.
- Formal Properties.
- Open-source stream reasoning frameworks.
- Reproducible empirical evaluation on large data streams: considerable improvement wrt state-of-the-art.

---

*https://github.com/aartikis/rtec

†https://github.com/periklismant/opiec

Future Work

Possible Directions:

- Explanations for derived situations.

Future Work

Possible Directions:

- Explanations for derived situations.
- Comparison with automata-based complex event recognition formalisms.

Future Work

Possible Directions:

- Explanations for derived situations.
- Comparison with automata-based complex event recognition formalisms.
- Distributed and parallel stream reasoning.

Future Work

Possible Directions:

- Explanations for derived situations.
- Comparison with automata-based complex event recognition formalisms.
- Distributed and parallel stream reasoning.
- Stream reasoning in tensor spaces.

Future Work

Possible Directions:

- Explanations for derived situations.
- Comparison with automata-based complex event recognition formalisms.
- Distributed and parallel stream reasoning.
- Stream reasoning in tensor spaces.
- Neuro-symbolic stream reasoning.

Appendix

Logic Programming

Logic program:

- A set of rules $a \leftarrow b_1, \ldots, b_m, \text{not } c_1, \ldots, \text{not } c_k$.

Przymusinski T. C., On declarative semantics of deductive databases and logic programs. In Foundations of Deductive Databases and Logic Programming. 193–216, 1988.

Lloyd J. W., Foundations of logic programming, 2nd edition. 1987.

Logic Programming

Logic program:

- A set of rules $a \leftarrow b_1, \ldots, b_m, \text{not } c_1, \ldots, \text{not } c_k$.

Herbrand models of logic program P:

- The sets of ground atoms for which all rules in P are true.

Przymusinski T. C., On declarative semantics of deductive databases and logic programs. In Foundations of Deductive Databases and Logic Programming. 193–216, 1988.

Lloyd J. W., Foundations of logic programming, 2nd edition. 1987.

Logic Programming

Logic program:

- A set of rules $a \leftarrow b_1, \ldots, b_m, \text{not } c_1, \ldots, \text{not } c_k$.

Herbrand models of logic program P:

- The sets of ground atoms for which all rules in P are true.

Semantics of logic program P:

- Identifies the model of P that denotes its intended meaning.

Przymusinski T. C., On declarative semantics of deductive databases and logic programs. In Foundations of Deductive Databases and Logic Programming. 193–216, 1988.

Lloyd J. W., Foundations of logic programming, 2nd edition. 1987.

Logic Programming

Logic program:

- A set of rules $a \leftarrow b_1, \ldots, b_m, \text{not } c_1, \ldots, \text{not } c_k$.

Herbrand models of logic program P:

- The sets of ground atoms for which all rules in P are true.

Semantics of logic program P:

- Identifies the model of P that denotes its intended meaning.

Locally stratified logic program P:

- There is a partitioning $P_0, \ldots, P_n$ of the ground atoms of P, such that, for each ground rule, if $a \in P_i$, then
  - $b_1, \ldots, b_m \in P_j$, where $j \leq i$, and
  - $c_1, \ldots, c_k \in P_j$, where $j < i$.
- Semantics of P: the unique perfect model of P.

Przymusinski T. C., On declarative semantics of deductive databases and logic programs. In Foundations of Deductive Databases and Logic Programming. 193–216, 1988.

Lloyd J. W., Foundations of logic programming, 2nd edition. 1987.

Logic Programming

Logic program:

- A set of rules $a \leftarrow b_1, \ldots, b_m, \text{not } c_1, \ldots, \text{not } c_k$.

Herbrand models of logic program P:

- The sets of ground atoms for which all rules in P are true.

Semantics of logic program P:

- Identifies the model of P that denotes its intended meaning.

Locally stratified logic program P:

- There is a partitioning $P_0, \ldots, P_n$ of the ground atoms of P, such that, for each ground rule, if $a \in P_i$, then
  - $b_1, \ldots, b_m \in P_j$, where $j \leq i$, and
  - $c_1, \ldots, c_k \in P_j$, where $j < i$.
- Semantics of P: the unique perfect model of P.

Our frameworks operate on locally stratified logic programs.

Przymusinski T. C., On declarative semantics of deductive databases and logic programs. In Foundations of Deductive Databases and Logic Programming. 193–216, 1988.

Lloyd J. W., Foundations of logic programming, 2nd edition. 1987.

Run-Time Event Calculus (RTEC)

Predicate	Meaning
happensAt(E, T)	Event E occurs at time T
initiatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is initiated
terminatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is terminated
holdsFor$(F = V, I)$	I is the list of the maximal intervals for which $F = V$ holds continuously
holdsAt$(F = V, T)$	The value of fluent F is V at time T
union_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cup \ldots \cup J_n)$
intersect_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cap \ldots \cap J_n)$
relative_complement_all $(I',\ [J_1, \ldots, J_n],\ I)$	$I = I' \setminus (J_1 \cup \ldots \cup J_n)$

Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Run-Time Event Calculus (RTEC)

Predicate	Meaning
happensAt(E, T)	Event E occurs at time T
initiatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is initiated
terminatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is terminated
holdsFor$(F = V, I)$	I is the list of the maximal intervals for which $F = V$ holds continuously
holdsAt$(F = V, T)$	The value of fluent F is V at time T
union_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cup \ldots \cup J_n)$
intersect_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cap \ldots \cap J_n)$
relative_complement_all $(I',\ [J_1, \ldots, J_n],\ I)$	$I = I' \setminus (J_1 \cup \ldots \cup J_n)$

Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Run-Time Event Calculus (RTEC): Fluent Specification

Simple Fluents:

$\textbf{initiatedAt}(F = V, T) \leftarrow$
$\quad \textbf{happensAt}(E_{In_1}, T)[,$
$\quad \text{conditions}].$
$\quad \vdots$

$\textbf{terminatedAt}(F = V, T) \leftarrow$
$\quad \textbf{happensAt}(E_{T_1}, T)[,$
$\quad \text{conditions}].$
$\quad \vdots$

where conditions:
$^{0-K}[\text{not}]\ \textbf{happensAt}(E_k, T),$
$^{0-M}[\text{not}]\ \textbf{holdsAt}(F_m = V_m, T),$
$^{0-N}\text{atemporal-constraint}_n$

Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Run-Time Event Calculus (RTEC): Fluent Specification

Simple Fluents:

$\textbf{initiatedAt}(F = V, T) \leftarrow$
$\quad \textbf{happensAt}(E_{In_1}, T)[,$
$\quad$ conditions].
$\quad \vdots$
$\textbf{terminatedAt}(F = V, T) \leftarrow$
$\quad \textbf{happensAt}(E_{T_1}, T)[,$
$\quad$ conditions].
$\quad \vdots$

where conditions:
$^{0-K}$[not] $\textbf{happensAt}(E_k, T)$,
$^{0-M}$[not] $\textbf{holdsAt}(F_m = V_m, T)$,
$^{0-N}\text{atemporal-constraint}_n$

Statically Determined Fluents:

$\textbf{holdsFor}(F = V, I) \leftarrow$
$\quad \textbf{holdsFor}(F_1 = V_1, I_1)[,$
$\quad \textbf{holdsFor}(F_2 = V_2, I_2), \ldots$
$\quad \textbf{holdsFor}(F_n = V_n, I_n),$
$\quad \text{intervalOperation}(L_1, I_{n+1}), \ldots$
$\quad \text{intervalOperation}(L_m, I)].$

where intervalOperation:
union_all or
intersect_all or
relative_complement_all

Artikis et al., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

Fluent-Value Pair Computation

Definition:

$$\textbf{initiatedAt}(F=V,\ T) \leftarrow$$
$$\quad \textbf{happensAt}(E_{In_1},\ T),$$
$$\quad [\text{conditions}]$$

$\ldots$

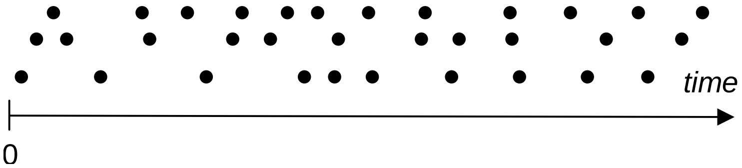
$$\textbf{initiatedAt}(F=V,\ T) \leftarrow$$
$$\quad \textbf{happensAt}(E_{In_i},\ T),$$
$$\quad [\text{conditions}]$$

$$\textbf{terminatedAt}(F=V,\ T) \leftarrow$$
$$\quad \textbf{happensAt}(E_{T_1},\ T),$$
$$\quad [\text{conditions}]$$

$\ldots$

$$\textbf{terminatedAt}(F=V,\ T) \leftarrow$$
$$\quad \textbf{happensAt}(E_{T_j},\ T),$$
$$\quad [\text{conditions}]$$

Reasoning:



Fluent-Value Pair Computation

Definition:

$$\textbf{initiatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{In_1},\ T), \\ \quad [\text{conditions}]$$

$\dots$

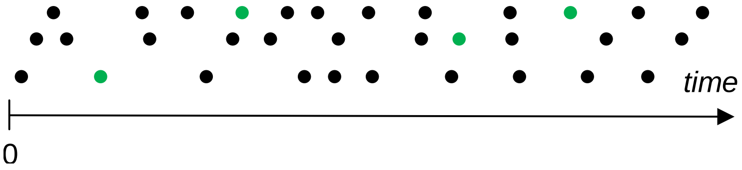
$$\textbf{initiatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{In_i},\ T), \\ \quad [\text{conditions}]$$

$$\textbf{terminatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{T_1},\ T), \\ \quad [\text{conditions}]$$

$\dots$

$$\textbf{terminatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{T_j},\ T), \\ \quad [\text{conditions}]$$

Reasoning:



Fluent-Value Pair Computation

Definition:

$$\textbf{initiatedAt}(F = V,\ T) \leftarrow \textbf{happensAt}(E_{In_1},\ T),\ [\text{conditions}]$$

$\dots$

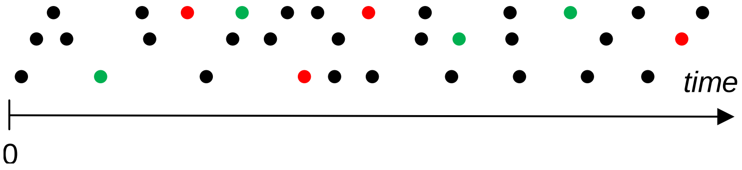
$$\textbf{initiatedAt}(F = V,\ T) \leftarrow \textbf{happensAt}(E_{In_i},\ T),\ [\text{conditions}]$$

$$\textbf{terminatedAt}(F = V,\ T) \leftarrow \textbf{happensAt}(E_{T_1},\ T),\ [\text{conditions}]$$

$\dots$

$$\textbf{terminatedAt}(F = V,\ T) \leftarrow \textbf{happensAt}(E_{T_j},\ T),\ [\text{conditions}]$$

Reasoning:



Fluent-Value Pair Computation

Definition:

$$\mathbf{initiatedAt}(F=V,\ T) \leftarrow \mathbf{happensAt}(E_{In_1},\ T),\ [\text{conditions}]$$

$\ldots$

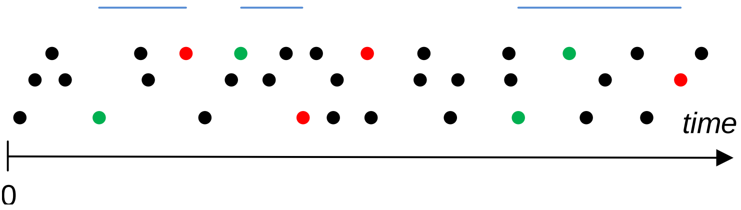
$$\mathbf{initiatedAt}(F=V,\ T) \leftarrow \mathbf{happensAt}(E_{In_i},\ T),\ [\text{conditions}]$$

$$\mathbf{terminatedAt}(F=V,\ T) \leftarrow \mathbf{happensAt}(E_{T_1},\ T),\ [\text{conditions}]$$

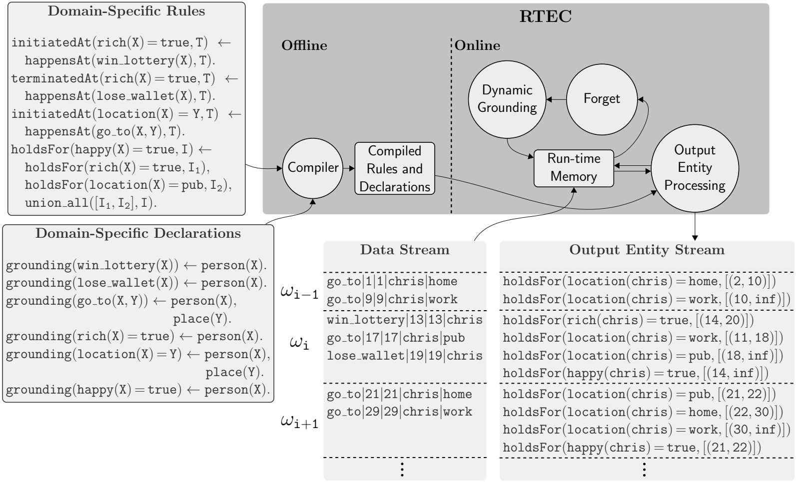
$\ldots$

$$\mathbf{terminatedAt}(F=V,\ T) \leftarrow \mathbf{happensAt}(E_{T_j},\ T),\ [\text{conditions}]$$

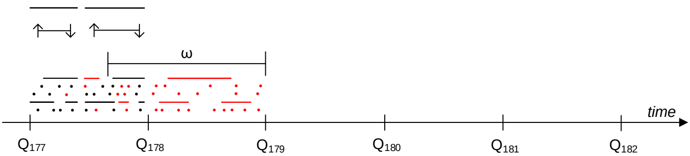
Reasoning: $\mathbf{holdsFor}(F=V,\ I)$



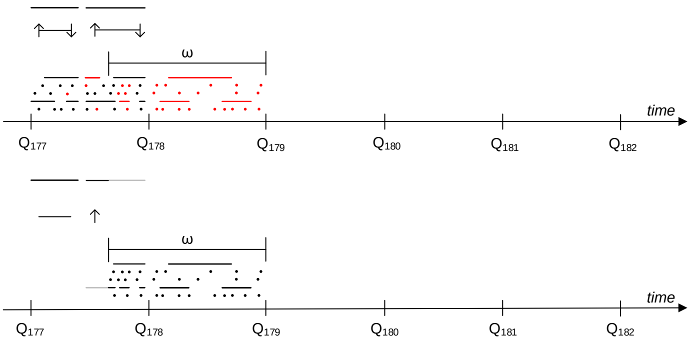
RTEC Architecture



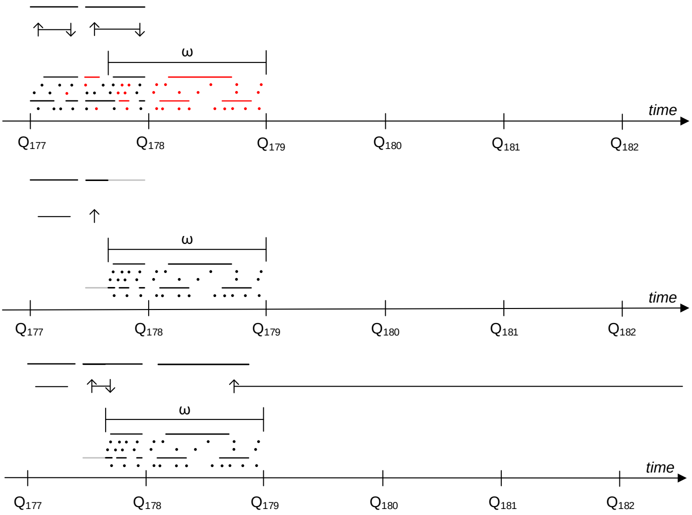
RTEC: Windowing



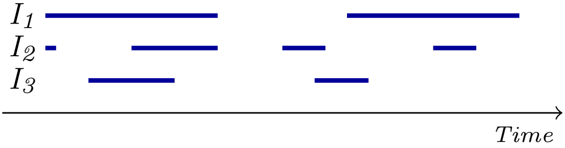
RTEC: Windowing



RTEC: Windowing

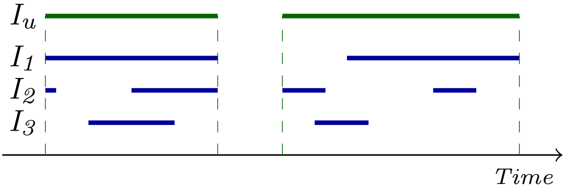


RTEC: Interval-based Reasoning



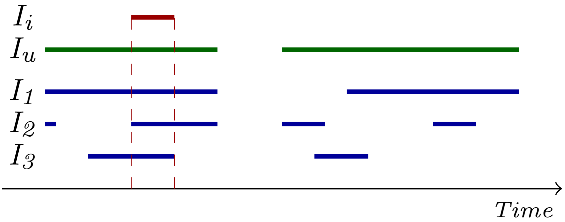
RTEC: Interval-based Reasoning

union_all$([I_1, I_2, I_3], I_u)$



RTEC: Interval-based Reasoning

intersect_all$([I_1, I_2, I_3], I_i)$
union_all$([I_1, I_2, I_3], I_u)$

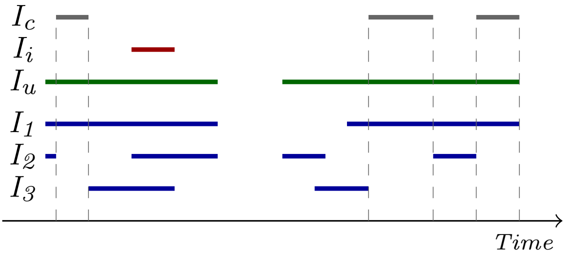


RTEC: Interval-based Reasoning

relative_complement_all$(I_1, [I_2, I_3], I_c)$

intersect_all$([I_1, I_2, I_3], I_i)$

union_all$([I_1, I_2, I_3], I_u)$

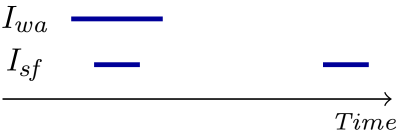


Statically Determined Fluent: Anchored or Moored

$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{anchoredOrMoored}(\mathit{Vessel})=\mathrm{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel})=\mathit{farFromPorts},\ I_{sf}), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel},\mathit{anchorage})=\mathit{true},\ I_{wa}), \\
&\quad \textbf{intersect\_all}([I_{sf}, I_{wa}],\ I_{sa}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel})=\mathit{nearPorts},\ I_{sn}), \\
&\quad \textbf{union\_all}([I_{sa}, I_{sn}],\ I).
\end{aligned}
$$

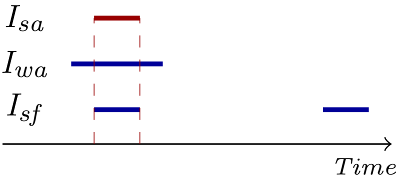
Statically Determined Fluent: Anchored or Moored

$$\begin{aligned}
&\textbf{holdsFor}(\mathit{anchoredOrMoored}(\mathit{Vessel}) = \text{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{farFromPorts},\ I_{sf}), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{anchorage}) = \mathit{true},\ I_{wa}), \\
&\quad \textbf{intersect_all}([I_{sf}, I_{wa}],\ I_{sa}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{nearPorts},\ I_{sn}), \\
&\quad \textbf{union_all}([I_{sa}, I_{sn}],\ I).
\end{aligned}$$



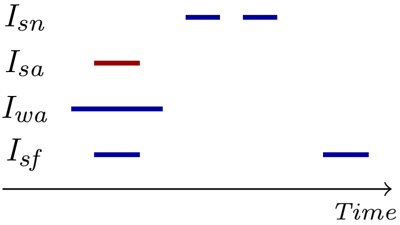
Statically Determined Fluent: Anchored or Moored

$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{anchoredOrMoored}(\mathit{Vessel}) = \mathrm{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{farFromPorts},\ I_{sf}), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{anchorage}) = \mathit{true},\ I_{wa}), \\
&\quad \textbf{intersect\_all}([I_{sf}, I_{wa}],\ I_{sa}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{nearPorts},\ I_{sn}), \\
&\quad \textbf{union\_all}([I_{sa}, I_{sn}],\ I).
\end{aligned}
$$



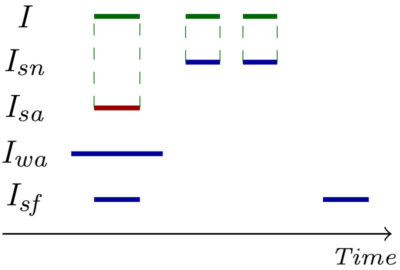
Statically Determined Fluent: Anchored or Moored

$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{anchoredOrMoored}(\mathit{Vessel}) = \mathrm{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{farFromPorts},\ I_{sf}), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{anchorage}) = \mathit{true},\ I_{wa}), \\
&\quad \textbf{intersect\_all}([I_{sf}, I_{wa}],\ I_{sa}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{nearPorts},\ I_{sn}), \\
&\quad \textbf{union\_all}([I_{sa}, I_{sn}],\ I).
\end{aligned}
$$



Statically Determined Fluent: Anchored or Moored

$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{anchoredOrMoored}(\mathit{Vessel}) = \mathrm{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{farFromPorts},\ I_{sf}), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{anchorage}) = \mathit{true},\ I_{wa}), \\
&\quad \textbf{intersect\_all}([I_{sf}, I_{wa}],\ I_{sa}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{nearPorts},\ I_{sn}), \\
&\quad \textbf{union\_all}([I_{sa}, I_{sn}],\ I).
\end{aligned}
$$



Voting: Cyclic Dependencies

$$\begin{aligned}&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{proposed}, T) \leftarrow\\ &\quad \textbf{happensAt}(\mathit{propose}(P, M), T),\\ &\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{null}, T).\end{aligned}$$

Voting: Cyclic Dependencies

$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{proposed}, T) \leftarrow\\
&\quad \textbf{happensAt}(\mathit{propose}(P, M), T),\\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{null}, T).\\
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voting}, T) \leftarrow\\
&\quad \textbf{happensAt}(\mathit{second}(S, M), T),\\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{proposed}, T).
\end{aligned}$$

Voting: Cyclic Dependencies

$$
\begin{aligned}
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{proposed}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{propose}(P, M), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{null}, T). \\
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voting}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{second}(S, M), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{proposed}, T). \\
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voted}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{close\_ballot}(C, M), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{voting}, T).
\end{aligned}
$$

Voting: Cyclic Dependencies

$$
\begin{aligned}
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{proposed}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{propose}(P, M), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{null}, T). \\
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voting}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{second}(S, M), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{proposed}, T). \\
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voted}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{close\_ballot}(C, M), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{voting}, T). \\
&\textbf{initiatedAt}(\mathit{status}(M) = \mathit{null}, T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{declare}(C, M, \mathit{Res}), T), \\
&\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{voted}, T).
\end{aligned}
$$

Voting: Cyclic Dependencies

$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{proposed}, T) \leftarrow$$
$$\textbf{happensAt}(\mathit{propose}(P, M), T),$$
$$\textbf{holdsAt}(\mathit{status}(M) = \mathit{null}, T).$$

$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voting}, T) \leftarrow$$
$$\textbf{happensAt}(\mathit{second}(S, M), T),$$
$$\textbf{holdsAt}(\mathit{status}(M) = \mathit{proposed}, T).$$

$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voted}, T) \leftarrow$$
$$\textbf{happensAt}(\mathit{close\_ballot}(C, M), T),$$
$$\textbf{holdsAt}(\mathit{status}(M) = \mathit{voting}, T).$$

$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{null}, T) \leftarrow$$
$$\textbf{happensAt}(\mathit{declare}(C, M, \mathit{Res}), T),$$
$$\textbf{holdsAt}(\mathit{status}(M) = \mathit{voted}, T).$$

Voting: Cyclic Dependencies

$$
\begin{aligned}
&\textbf{initiatedAt}(status(M)=proposed,\ T) \leftarrow \\
&\quad \textbf{happensAt}(propose(P, M),\ T), \\
&\quad \textbf{holdsAt}(status(M)=null,\ T). \\
&\textbf{initiatedAt}(status(M)=voting,\ T) \leftarrow \\
&\quad \textbf{happensAt}(second(S, M),\ T), \\
&\quad \textbf{holdsAt}(status(M)=proposed,\ T). \\
&\textbf{initiatedAt}(status(M)=voted,\ T) \leftarrow \\
&\quad \textbf{happensAt}(close\_ballot(C, M),\ T), \\
&\quad \textbf{holdsAt}(status(M)=voting,\ T). \\
&\textbf{initiatedAt}(status(M)=null,\ T) \leftarrow \\
&\quad \textbf{happensAt}(declare(C, M, Res),\ T), \\
&\quad \textbf{holdsAt}(status(M)=voted,\ T).
\end{aligned}
$$

Voting: Cyclic Dependencies

$$
\begin{aligned}
&\mathbf{initiatedAt}(status(M) = proposed, T) \leftarrow \\
&\quad \mathbf{happensAt}(propose(P, M), T), \\
&\quad \mathbf{holdsAt}(status(M) = null, T). \\
&\mathbf{initiatedAt}(status(M) = voting, T) \leftarrow \\
&\quad \mathbf{happensAt}(second(S, M), T), \\
&\quad \mathbf{holdsAt}(status(M) = proposed, T). \\
&\mathbf{initiatedAt}(status(M) = voted, T) \leftarrow \\
&\quad \mathbf{happensAt}(close\_ballot(C, M), T), \\
&\quad \mathbf{holdsAt}(status(M) = voting, T). \\
&\mathbf{initiatedAt}(status(M) = null, T) \leftarrow \\
&\quad \mathbf{happensAt}(declare(C, M, Res), T), \\
&\quad \mathbf{holdsAt}(status(M) = voted, T).
\end{aligned}
$$

Voting: Cyclic Dependencies

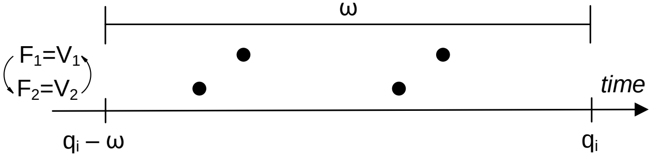
$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{proposed}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{propose}(P, M), T),$$
$$\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{null}, T).$$

$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voting}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{second}(S, M), T),$$
$$\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{proposed}, T).$$

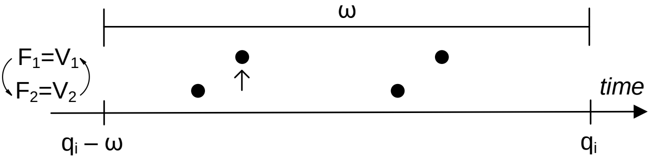
$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{voted}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{close\_ballot}(C, M), T),$$
$$\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{voting}, T).$$

$$\textbf{initiatedAt}(\mathit{status}(M) = \mathit{null}, T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{declare}(C, M, \mathit{Res}), T),$$
$$\quad \textbf{holdsAt}(\mathit{status}(M) = \mathit{voted}, T).$$

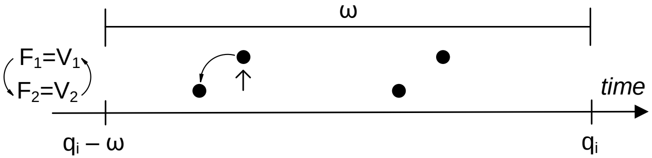
Handling Cyclic Dependencies



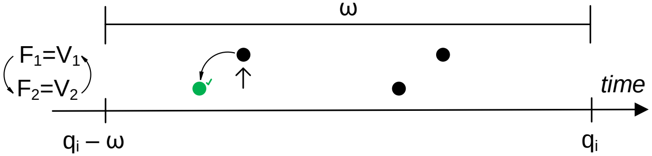
Handling Cyclic Dependencies



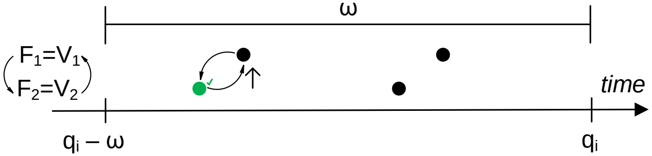
Handling Cyclic Dependencies



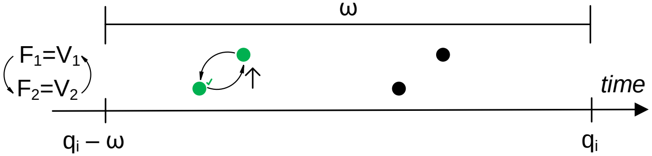
Handling Cyclic Dependencies



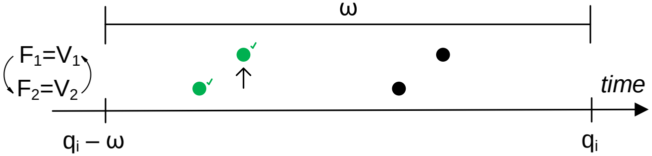
Handling Cyclic Dependencies



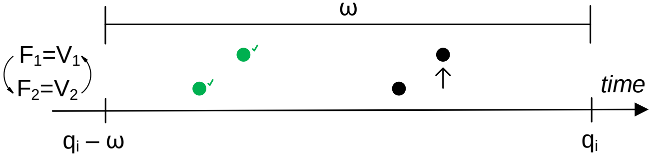
Handling Cyclic Dependencies



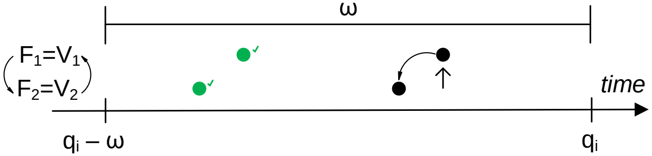
Handling Cyclic Dependencies



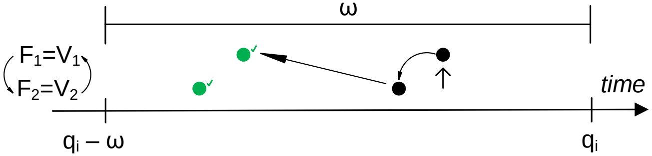
Handling Cyclic Dependencies



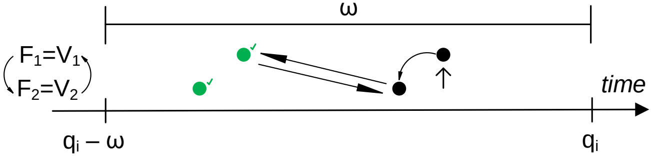
Handling Cyclic Dependencies



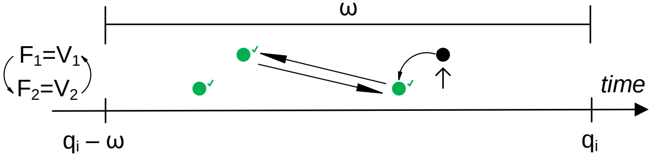
Handling Cyclic Dependencies



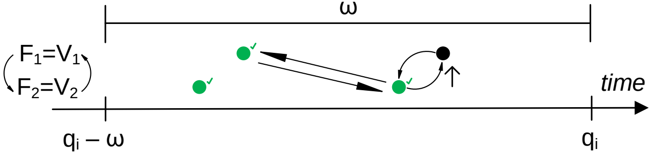
Handling Cyclic Dependencies



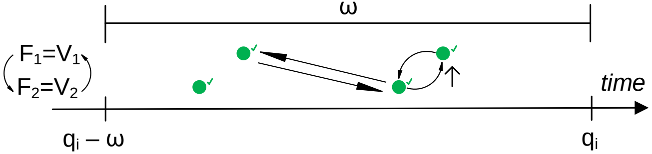
Handling Cyclic Dependencies



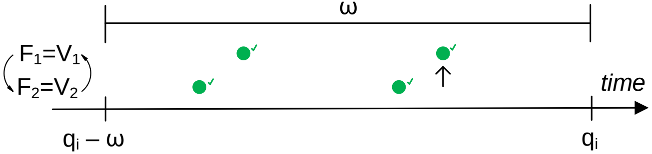
Handling Cyclic Dependencies



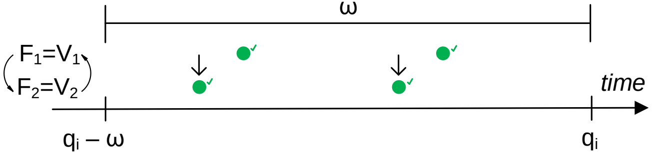
Handling Cyclic Dependencies



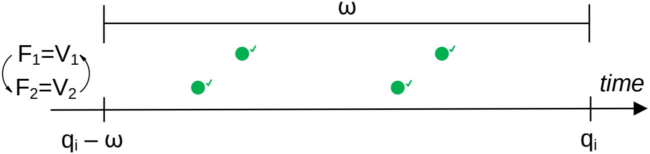
Handling Cyclic Dependencies



Handling Cyclic Dependencies

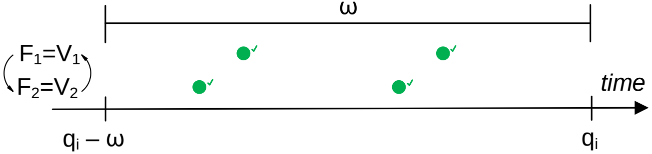


Handling Cyclic Dependencies

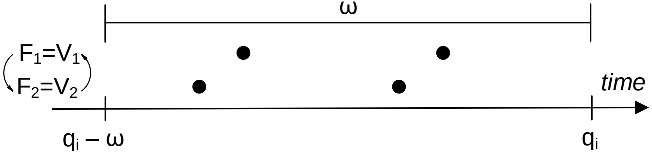


Handling Cyclic Dependencies

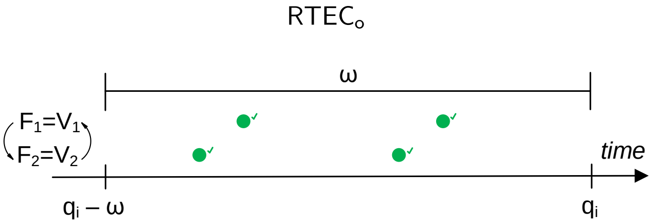
RTEC$_o$

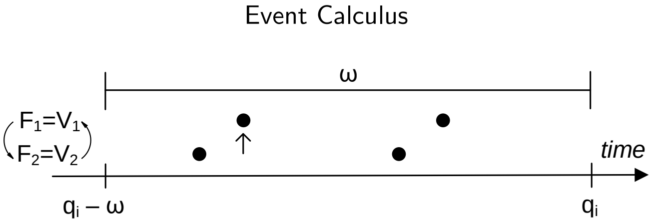


Event Calculus



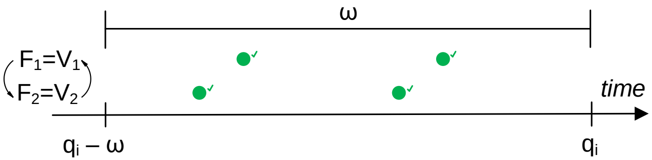
Handling Cyclic Dependencies



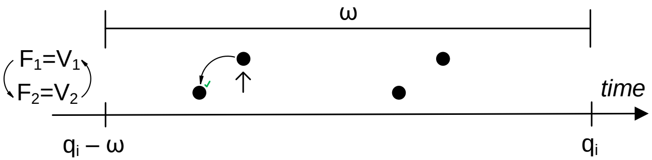


Handling Cyclic Dependencies

RTEC$_\circ$

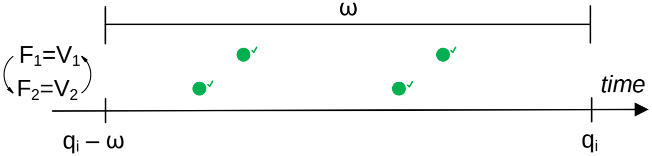


Event Calculus

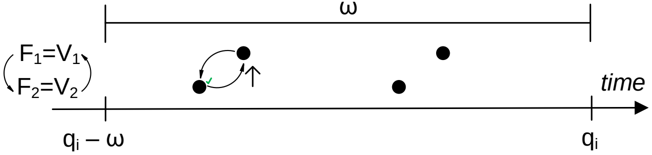


Handling Cyclic Dependencies

RTEC$_\circ$

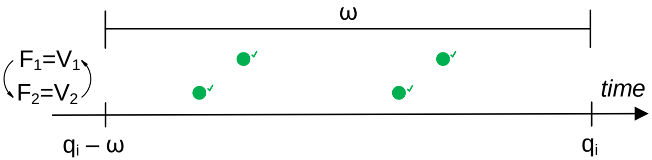


Event Calculus

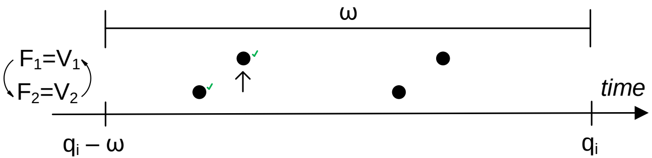


Handling Cyclic Dependencies

RTEC$_o$

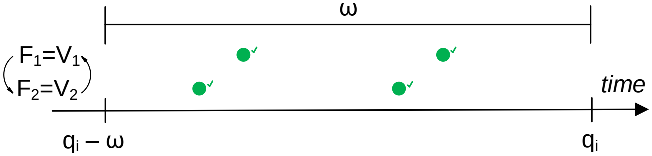


Event Calculus

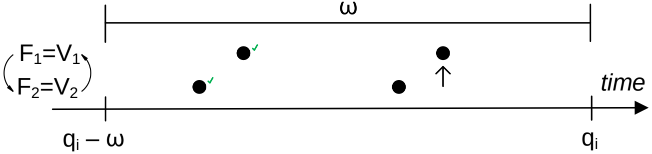


Handling Cyclic Dependencies

RTEC$_{\circ}$

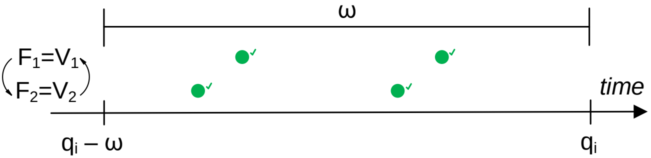


Event Calculus

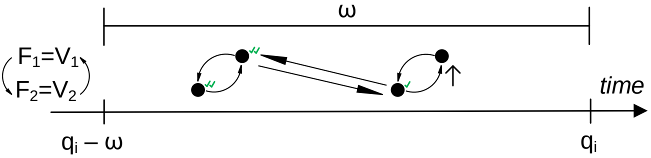


Handling Cyclic Dependencies

RTEC$_o$

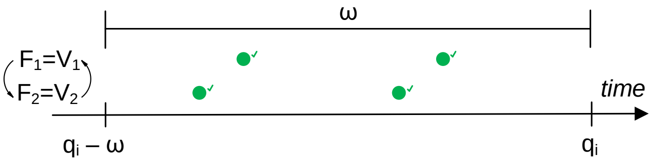


Event Calculus

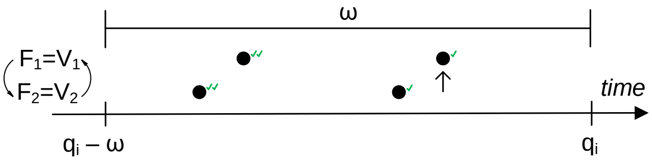


Handling Cyclic Dependencies

RTEC$_o$

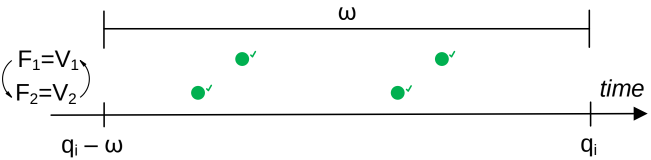


Event Calculus

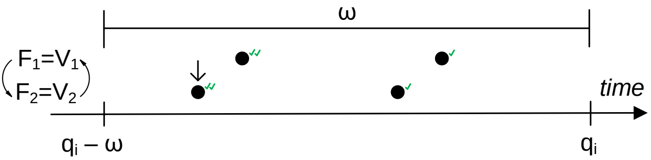


Handling Cyclic Dependencies

RTEC$_o$

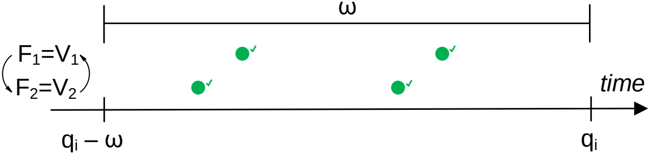


Event Calculus

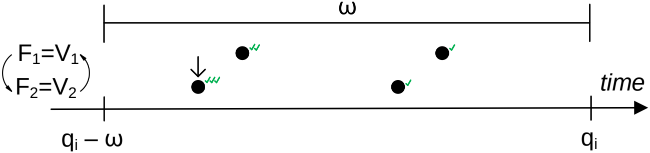


Handling Cyclic Dependencies

RTEC$_{\circ}$

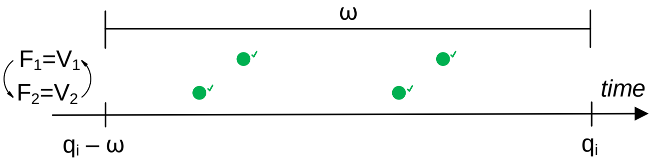


Event Calculus

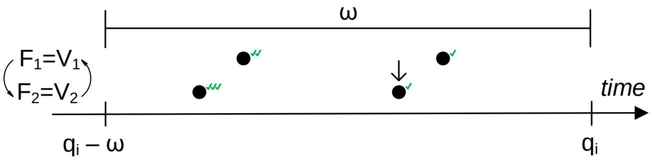


Handling Cyclic Dependencies

RTEC$_\circ$

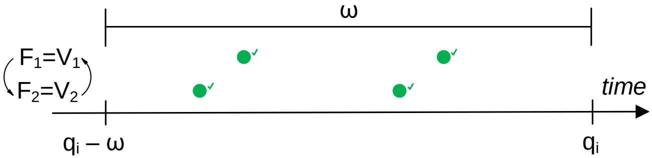


Event Calculus

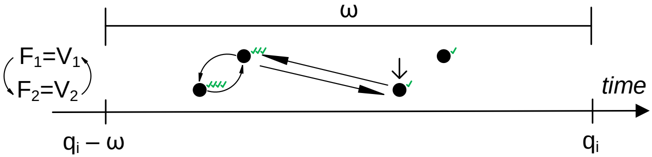


Handling Cyclic Dependencies

RTEC$_\circ$

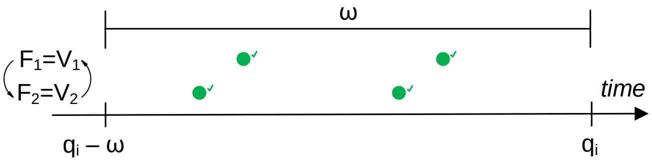


Event Calculus

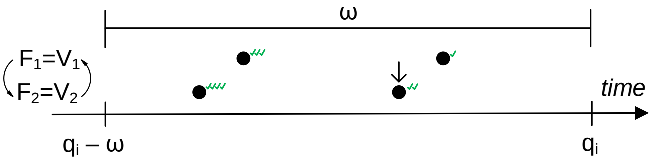


Handling Cyclic Dependencies

RTEC$_o$

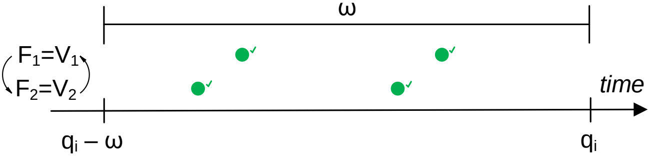


Event Calculus

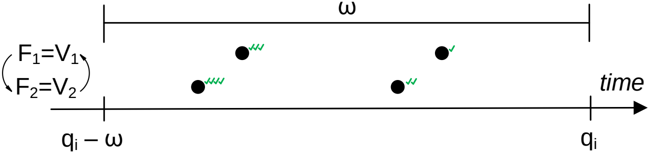


Handling Cyclic Dependencies

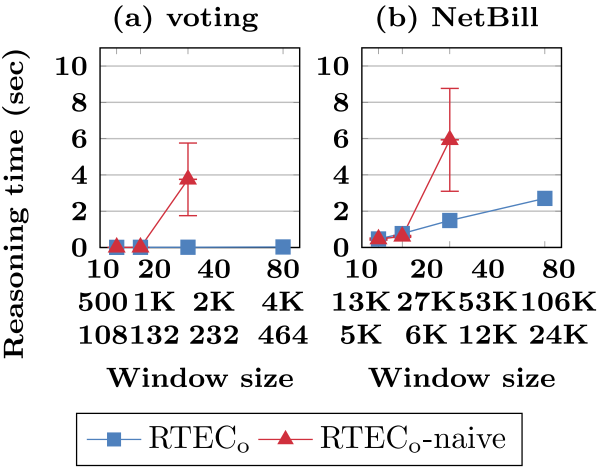
RTEC$_\circ$



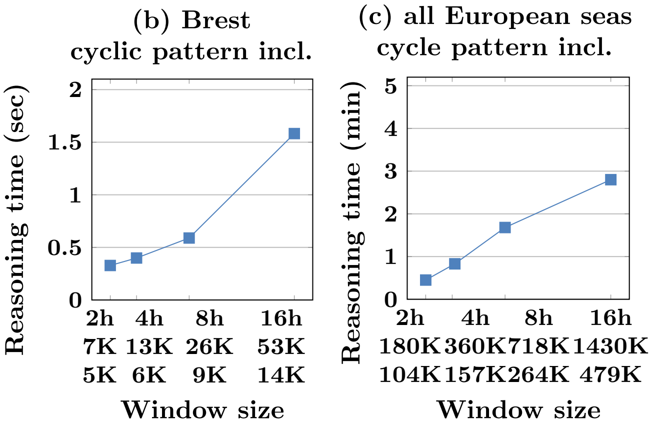
Event Calculus



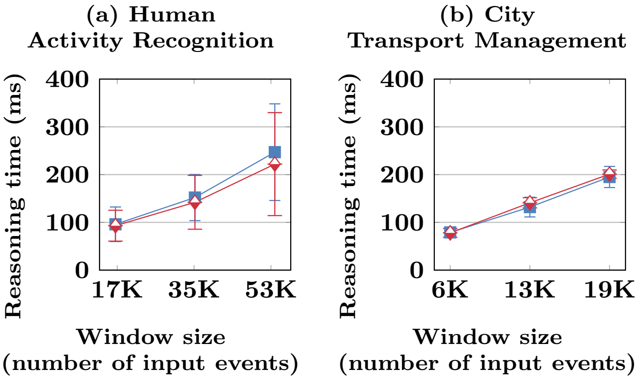
RTEC$_\circ$: Experimental Results



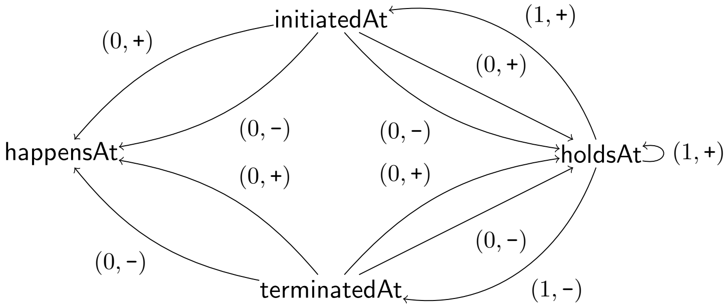
RTEC$_{\circ}$: Experimental Results



RTEC$_o$: Experimental Results

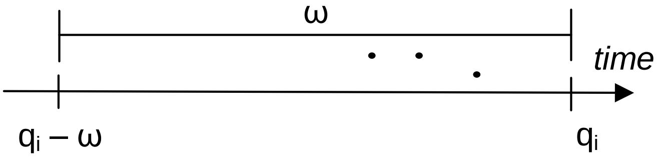


Semantics of RTEC$_\circ$

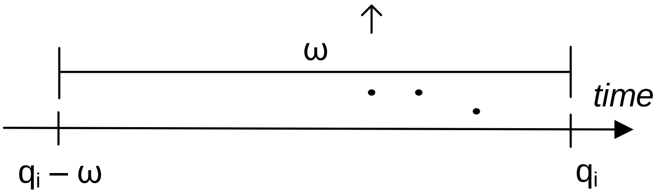


The cycle-sum graph of an RTEC$_\circ$ program.

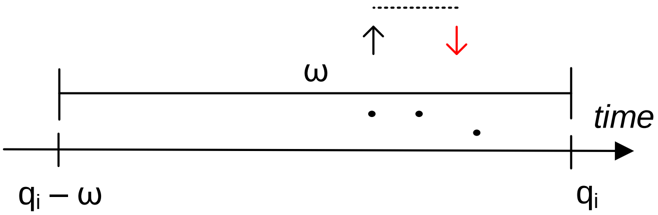
RTEC$^{\rightarrow}$: Deadlines



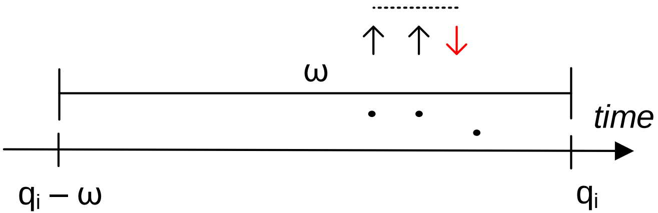
RTEC$^{\rightarrow}$: Deadlines



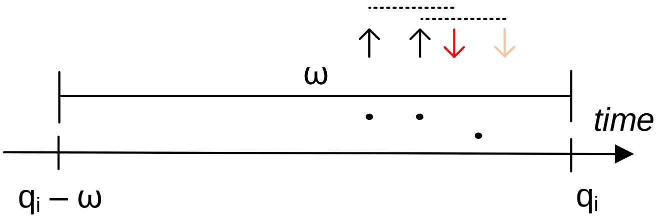
RTEC$^{\rightarrow}$: Deadlines



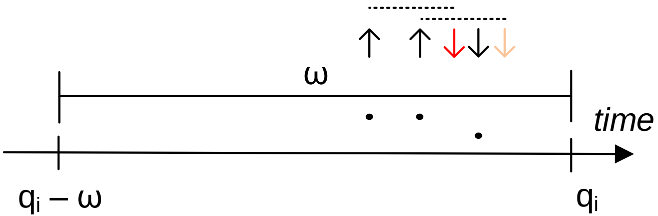
RTEC$^{\rightarrow}$: Deadlines



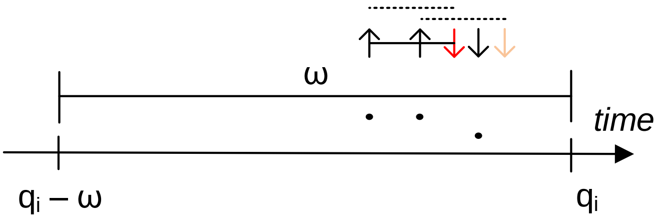
RTEC$^{\rightarrow}$: Deadlines



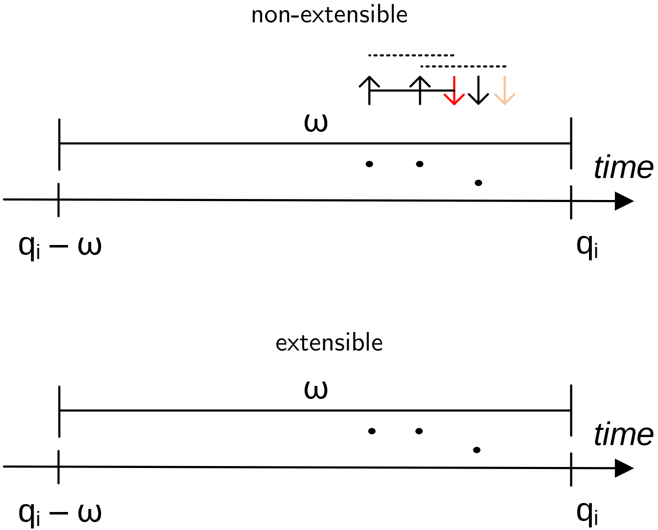
RTEC$^{\rightarrow}$: Deadlines



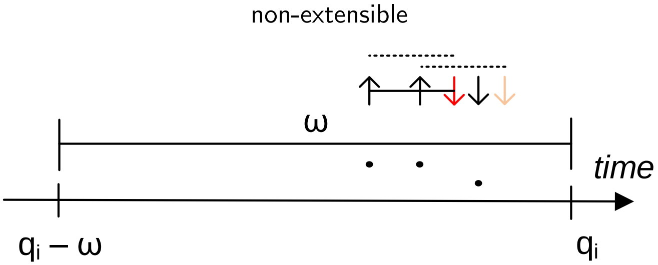
RTEC$^{\rightarrow}$: Deadlines

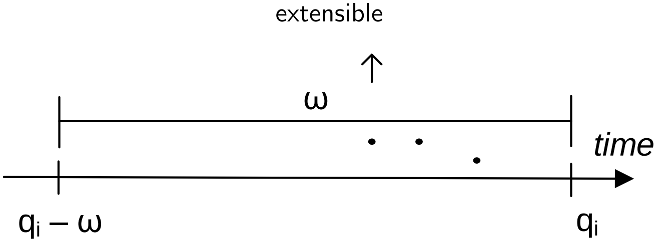


RTEC$^{\rightarrow}$: Deadlines

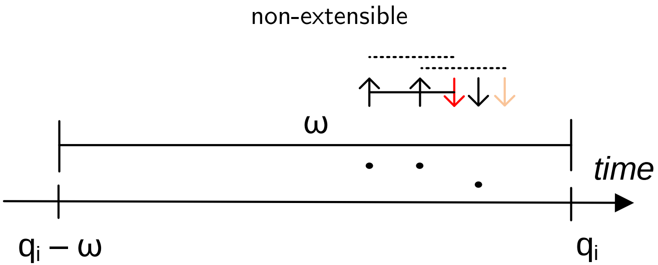


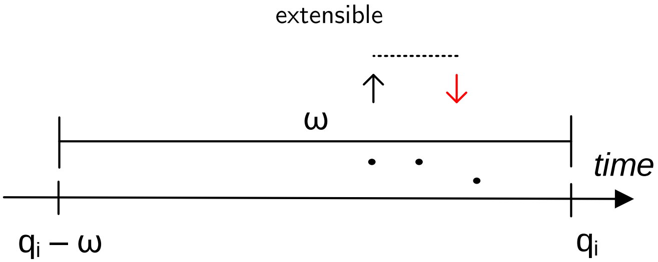
RTEC$^{\rightarrow}$: Deadlines





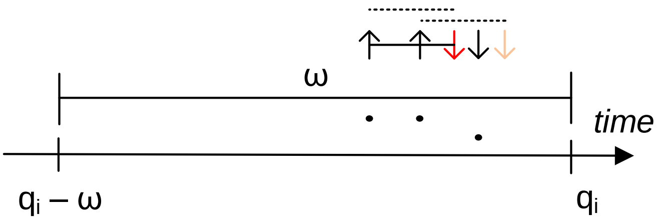
RTEC$^{\rightarrow}$: Deadlines



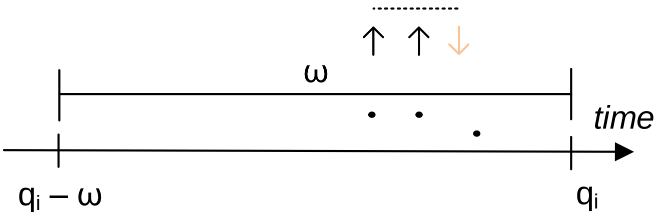


RTEC$^{\rightarrow}$: Deadlines

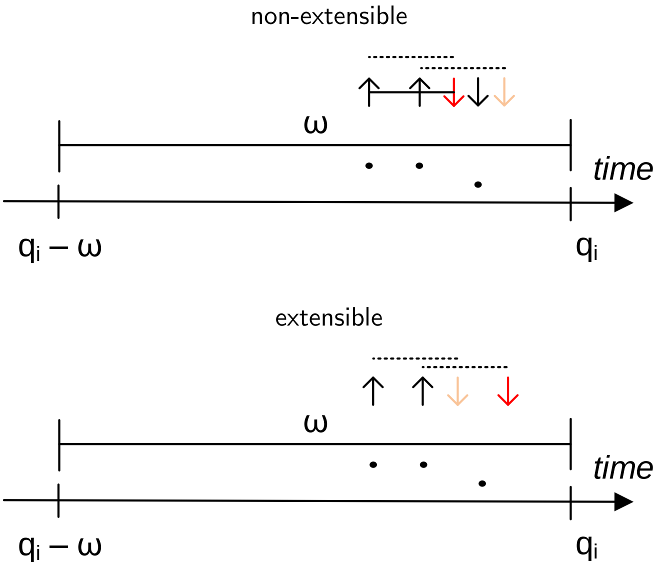
non-extensible



extensible

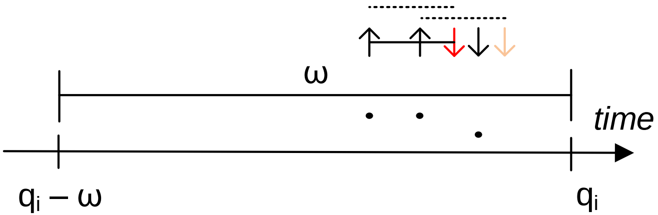


RTEC$^{\rightarrow}$: Deadlines

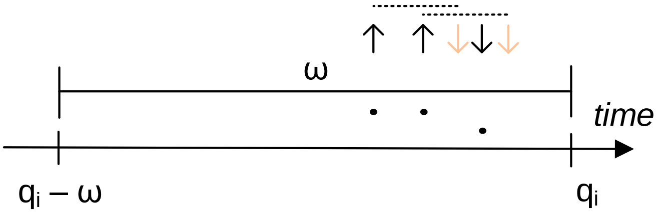


RTEC$^{\rightarrow}$: Deadlines

non-extensible

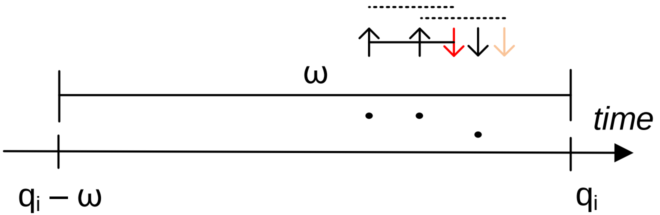


extensible

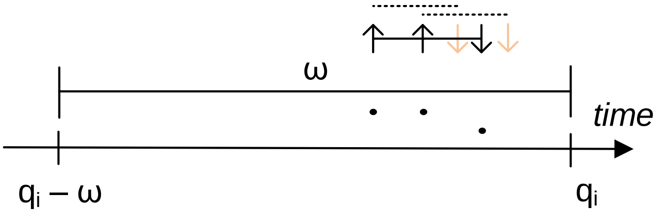


RTEC$^{\rightarrow}$: Deadlines

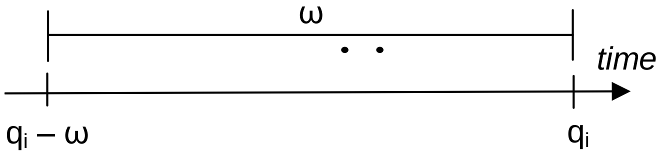
non-extensible



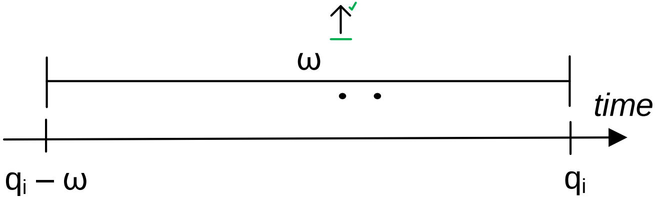
extensible



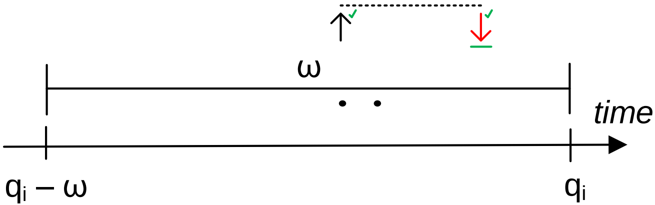
Handling Deadlines: Caching



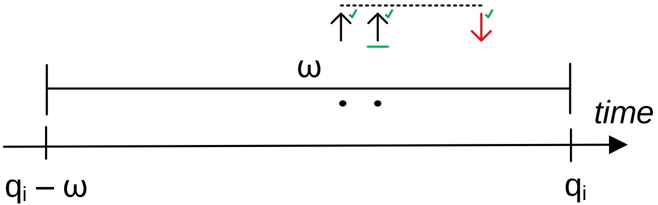
Handling Deadlines: Caching



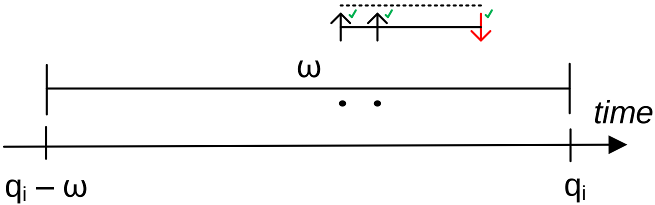
Handling Deadlines: Caching



Handling Deadlines: Caching

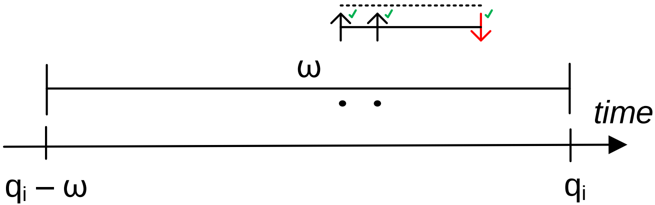


Handling Deadlines: Caching

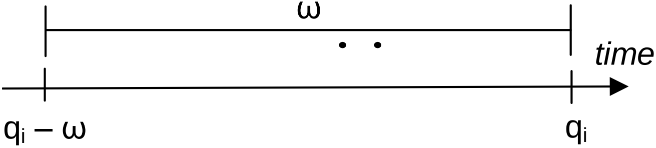


Handling Deadlines: Caching

RTEC$^{\rightarrow}$

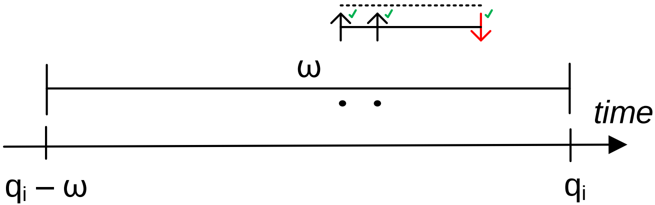


RTEC$^{\rightarrow}$-naive

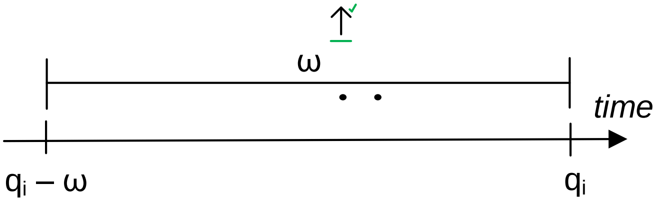


Handling Deadlines: Caching

RTEC$^{\rightarrow}$

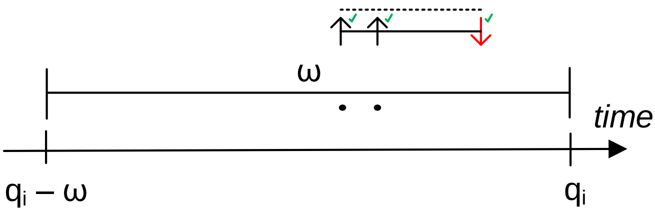


RTEC$^{\rightarrow}$-naive

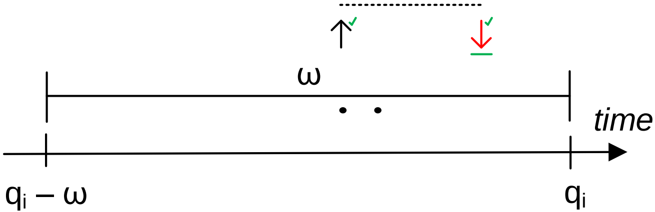


Handling Deadlines: Caching

RTEC$^{\rightarrow}$

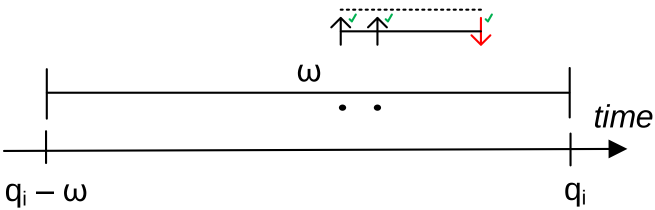


RTEC$^{\rightarrow}$-naive

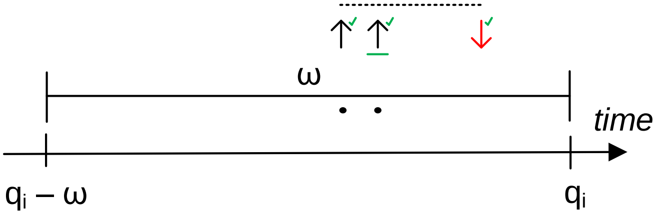


Handling Deadlines: Caching

RTEC→

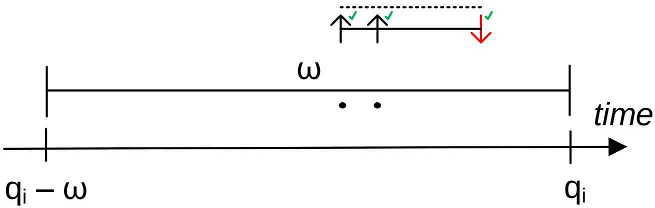


RTEC→-naive

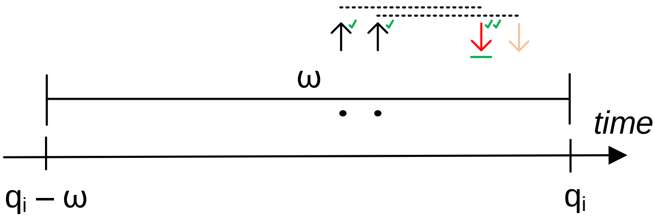


Handling Deadlines: Caching

RTEC$^{\rightarrow}$

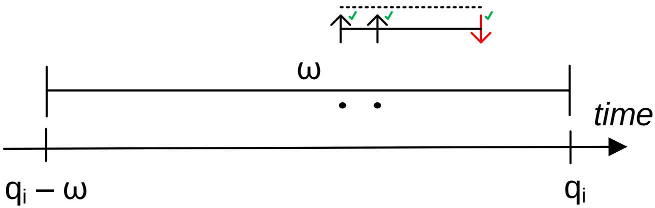


RTEC$^{\rightarrow}$-naive

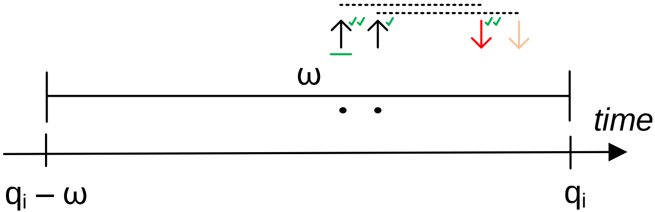


Handling Deadlines: Caching

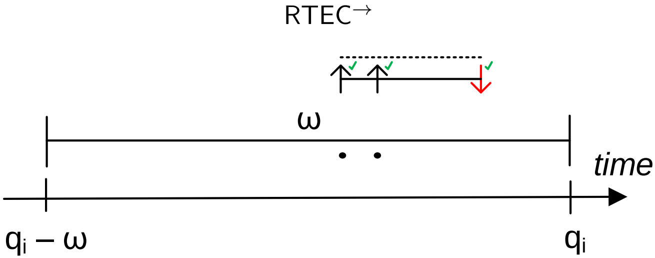
RTEC$^{\rightarrow}$

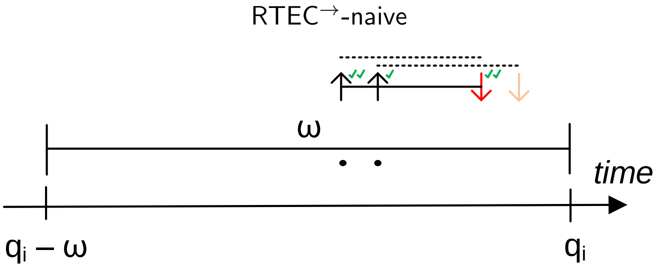


RTEC$^{\rightarrow}$-naive

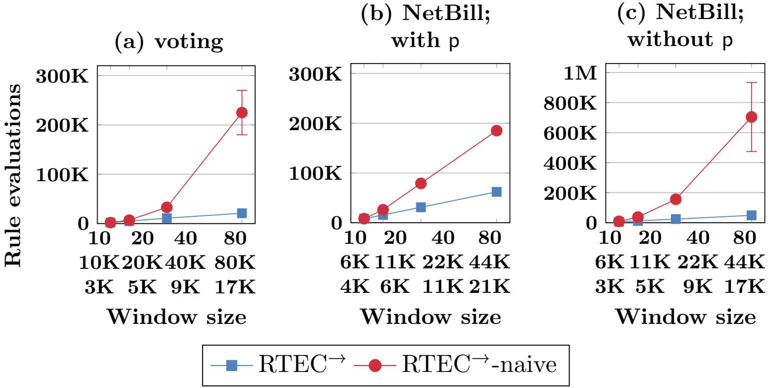


Handling Deadlines: Caching

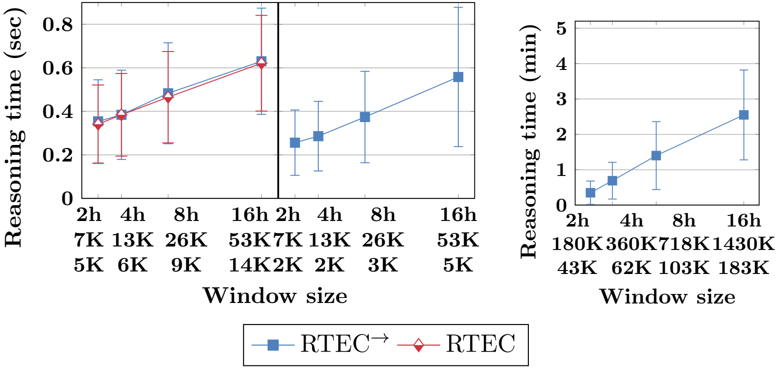




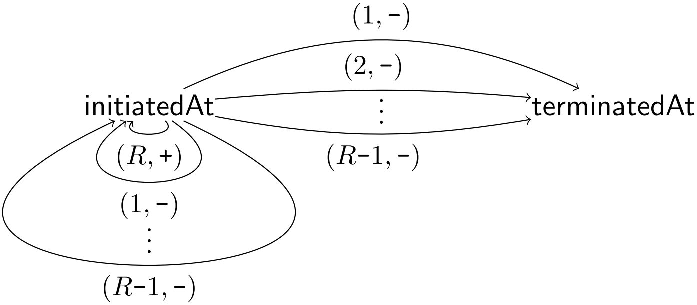
RTEC$^{\rightarrow}$: Experimental Results



RTEC$^{\rightarrow}$: Experimental Results



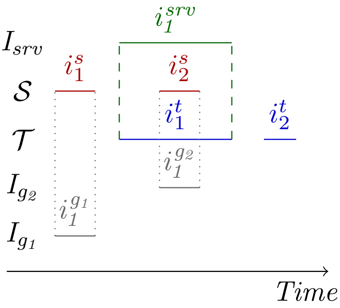
Semantics of RTEC$^{\rightarrow}$



The cycle-sum graph of an RTEC$^{\rightarrow}$ program.

RTEC$_A$: Allen Relations

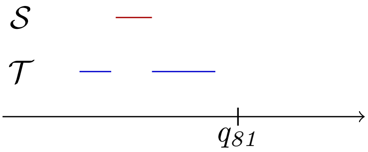
$\textbf{holdsFor}(\textit{suspiciousRendezVous}(\textit{Vessel}_1, \textit{Vessel}_2) = \text{true}, I) \leftarrow$
$\quad \textbf{holdsFor}(\textit{gap}(\textit{Vessel}_1) = \textit{farFromPorts}, I_{g_1}),$
$\quad \textbf{holdsFor}(\textit{gap}(\textit{Vessel}_2) = \textit{farFromPorts}, I_{g_2}),$
$\quad \textbf{holdsFor}(\textit{proximity}(\textit{Vessel}_1, \textit{Vessel}_2) = \text{true}, \mathcal{T}),$
$\quad \textbf{union\_all}([I_{g_1}, I_{g_2}], \mathcal{S}),$
$\quad \textbf{allen}(\text{during}, \mathcal{S}, \mathcal{T}, \text{target}, I).$



RTEC$_\mathsf{A}$: Windowing

$\textbf{holdsFor}(\textit{disappearedInArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, I) \leftarrow$
$\quad \textbf{holdsFor}(\textit{withinArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, \mathcal{S}),$
$\quad \textbf{holdsFor}(\textit{gap}(\textit{Vessel}) = \textit{farFromPorts}, \mathcal{T}),$
$\quad \textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$

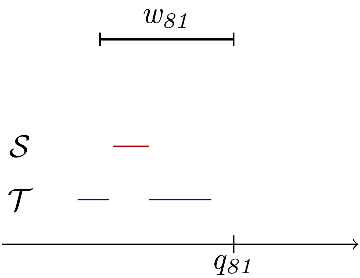
Query time: q_{81}



RTEC$_A$: Windowing

$\textbf{holdsFor}(\textit{disappearedInArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, I) \leftarrow$
  $\textbf{holdsFor}(\textit{withinArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, \mathcal{S}),$
  $\textbf{holdsFor}(\textit{gap}(\textit{Vessel}) = \textit{farFromPorts}, \mathcal{T}),$
  $\textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$

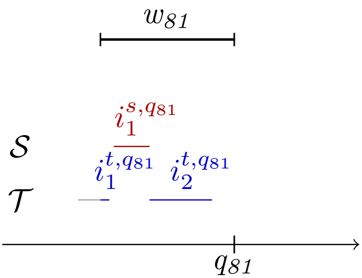
Query time: q_{81}



RTEC$_A$: Windowing

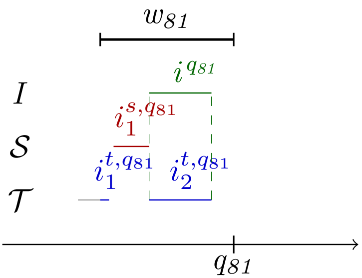
$$\textbf{holdsFor}(\mathit{disappearedInArea}(\mathit{Vessel}, \mathit{AreaType}) = \text{true}, I) \leftarrow$$
$$\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{AreaType}) = \text{true}, \mathcal{S}),$$
$$\quad \textbf{holdsFor}(\mathit{gap}(\mathit{Vessel}) = \mathit{farFromPorts}, \mathcal{T}),$$
$$\quad \textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$$

Query time: q_{81}



holdsFor(*disappearedInArea*(*Vessel*, *AreaType*) = true, *I*) ←
holdsFor(*withinArea*(*Vessel*, *AreaType*) = true, $\mathcal{S}$),
holdsFor(*gap*(*Vessel*) = *farFromPorts*, $\mathcal{T}$),
allen(meets, $\mathcal{S}$, $\mathcal{T}$, target, *I*).

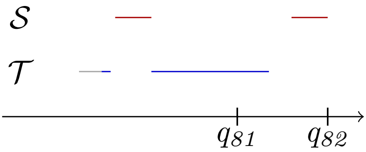
Query time: q_{81}



RTEC$_A$: Windowing

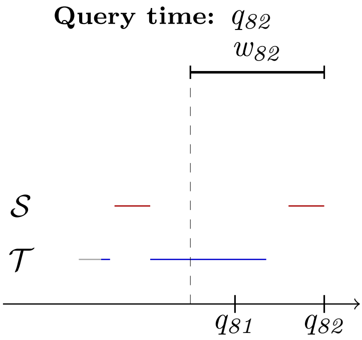
$$\textbf{holdsFor}(\mathit{disappearedInArea}(\mathit{Vessel}, \mathit{AreaType}) = \text{true}, I) \leftarrow$$
$$\textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{AreaType}) = \text{true}, \mathcal{S}),$$
$$\textbf{holdsFor}(\mathit{gap}(\mathit{Vessel}) = \mathit{farFromPorts}, \mathcal{T}),$$
$$\textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$$

Query time: q_{82}



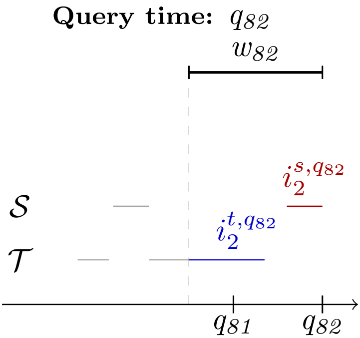
RTEC$_A$: Windowing

holdsFor(*disappearedInArea*(*Vessel*, *AreaType*) = true, *I*) ←
  holdsFor(*withinArea*(*Vessel*, *AreaType*) = true, $\mathcal{S}$),
  holdsFor(*gap*(*Vessel*) = *farFromPorts*, $\mathcal{T}$),
  allen(meets, $\mathcal{S}$, $\mathcal{T}$, target, *I*).



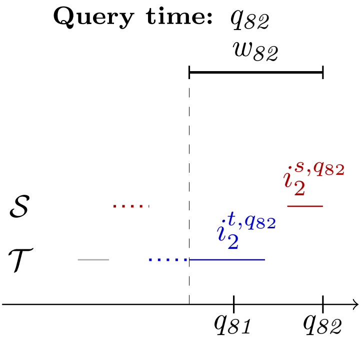
RTEC$_A$: Windowing

$\textbf{holdsFor}(\textit{disappearedInArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, I) \leftarrow$
  $\textbf{holdsFor}(\textit{withinArea}(\textit{Vessel}, \textit{AreaType}) = \text{true}, \mathcal{S}),$
  $\textbf{holdsFor}(\textit{gap}(\textit{Vessel}) = \textit{farFromPorts}, \mathcal{T}),$
  $\textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$



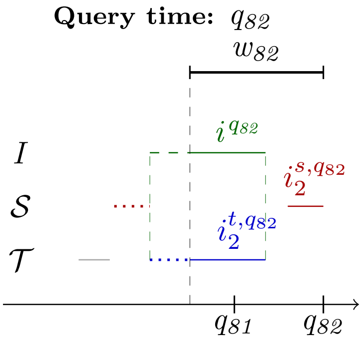
RTEC$_A$: Windowing

$\textbf{holdsFor}(\mathit{disappearedInArea}(\mathit{Vessel}, \mathit{AreaType}) = \text{true}, I) \leftarrow$
$\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{AreaType}) = \text{true}, \mathcal{S}),$
$\quad \textbf{holdsFor}(\mathit{gap}(\mathit{Vessel}) = \mathit{farFromPorts}, \mathcal{T}),$
$\quad \textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I).$



RTEC$_A$: Windowing

holdsFor(*disappearedInArea*(*Vessel*, *AreaType*) = true, *I*) ←
  holdsFor(*withinArea*(*Vessel*, *AreaType*) = true, $\mathcal{S}$),
  holdsFor(*gap*(*Vessel*) = *farFromPorts*, $\mathcal{T}$),
  allen(meets, $\mathcal{S}$, $\mathcal{T}$, target, *I*).



$RTEC_A$: Experimental Evaluation

Batch setting.

Batch size	Reasoning Time		
Input Intervals	$RTEC_A$	AEGLE	D^2IA
200	**1**	980	2K
2K	**14**	4K	6K
20K	**154**	71.5K	395K
200K	**1.8K**	MEM	>3.6M

Streaming setting.

Window size		Reasoning Time		Output Interval Pairs	
Days	Input Intervals	$RTEC_A$	D^2IA	$RTEC_A$	D^2IA
1	125	**1**	48	5K	5K
2	250	**2**	164	19K	18K
4	500	**4**	568	72K	71K
8	1K	**8**	1.7K	237K	236K
16	2K	**15**	7.8K	878K	874K

Instantaneous Recognition

$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T), \\
&\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).
\end{aligned}$$

$$\mathit{0.70} :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ \mathit{1}).$$
$$\mathit{0.46} :: \textbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ \mathit{1}).$$

Instantaneous Recognition

$$\textbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{happensAt}(walking(P_2),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$0.70 :: \textbf{happensAt}(walking(mike),\ 1).$$
$$0.46 :: \textbf{happensAt}(walking(sarah),\ 1).$$

$$P(\textbf{initiatedAt}(moving(mike, sarah) = \text{true}, 1)) =$$
$$P(\textbf{happensAt}(walking(mike), 1)) \times$$
$$P(\textbf{happensAt}(walking(sarah), 1)) \times$$
$$P(\textbf{holdsAt}(close(mike, sarah) = \text{true}, 1)) \times$$
$$P(\textbf{holdsAt}(orientation(mike, sarah) = \text{true}, 1))$$
$$= 0.7 \times 0.46 \times 1 \times 1 = 0.322$$

Instantaneous Recognition

$\mathbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \mathbf{happensAt}(\mathit{walking}(P_1),\ T),$
$\quad \mathbf{happensAt}(\mathit{walking}(P_2),\ T),$
$\quad \mathbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$
$\quad \mathbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$

$\mathbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \mathbf{happensAt}(\mathit{walking}(P_1),\ T),$
$\quad \mathbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$

$\mathit{0.70} :: \mathbf{happensAt}(\mathit{walking}(\mathit{mike}),\ \mathit{1}).$
$\mathit{0.46} :: \mathbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ \mathit{1}).$

$P(\mathbf{holdsAt}(CE = \text{true}, t)) =$
$\quad P(\mathbf{initiatedAt}(CE = \text{true}, t{-}\mathit{1}) \vee$
$\quad\quad (\mathbf{holdsAt}(CE = \text{true}, t{-}\mathit{1}) \wedge$
$\quad\quad \neg\ \mathbf{terminatedAt}(CE = \text{true}, t{-}\mathit{1})))$

Instantaneous Recognition

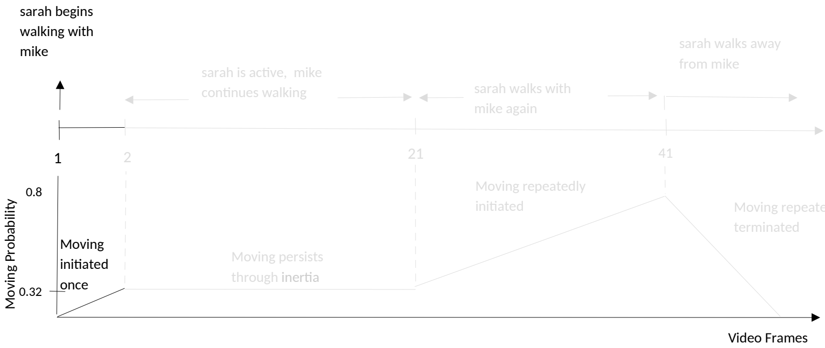
$\textbf{initiatedAt}(\textit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \textbf{happensAt}(\textit{walking}(P_1),\ T),$
$\quad \textbf{happensAt}(\textit{walking}(P_2),\ T),$
$\quad \textbf{holdsAt}(\textit{close}(P_1, P_2) = \text{true},\ T),$
$\quad \textbf{holdsAt}(\textit{orientation}(P_1, P_2) = \text{true},\ T).$

$\textbf{terminatedAt}(\textit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \textbf{happensAt}(\textit{walking}(P_1),\ T),$
$\quad \textbf{holdsAt}(\textit{close}(P_1, P_2) = \text{false},\ T).$

$0.70 :: \textbf{happensAt}(\textit{walking}(\textit{mike}),\ 1).$
$0.46 :: \textbf{happensAt}(\textit{walking}(\textit{sarah}),\ 1).$

$P(\textbf{holdsAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 2)) =$
$P(\textbf{initiatedAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 1) \vee$
$\quad (\textbf{holdsAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 1) \wedge$
$\quad \neg\textbf{terminatedAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 1)))$
$= 0.322 + 0 \times 1 - 0.322 \times 0 \times 1 = 0.322$

Instantaneous Recognition



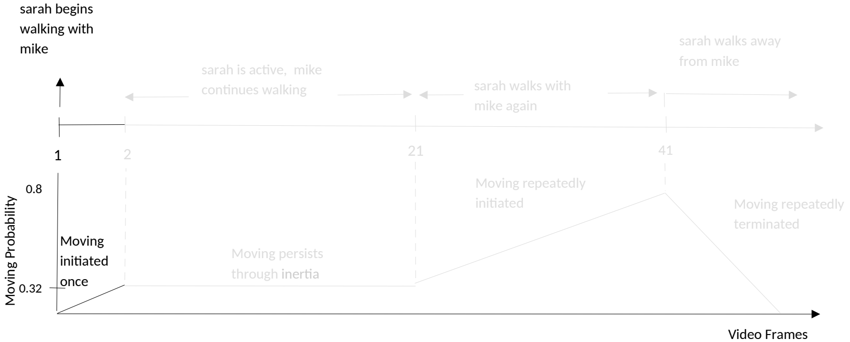
$$\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{happensAt}(\mathit{walking}(P_2),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$$

$$0.70 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 1).$$
$$0.46 :: \textbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ 1).$$

$$P(\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 2)) =$$
$$P(\textbf{initiatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 1) \vee$$
$$(\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 1) \wedge$$
$$\neg\textbf{terminatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 1)))$$
$$= 0.322 + 0 \times 1 - 0.322 \times 0 \times 1 = 0.322$$

Instantaneous Recognition

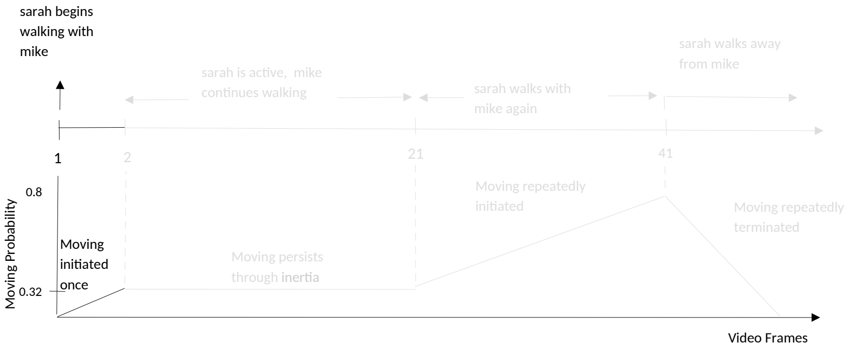


$$\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{happensAt}(\mathit{walking}(P_2),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$$

$$0.73 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 2).$$
$$0.55 :: \textbf{happensAt}(\mathit{active}(\mathit{sarah}),\ 2).\ \cdots$$

Instantaneous Recognition



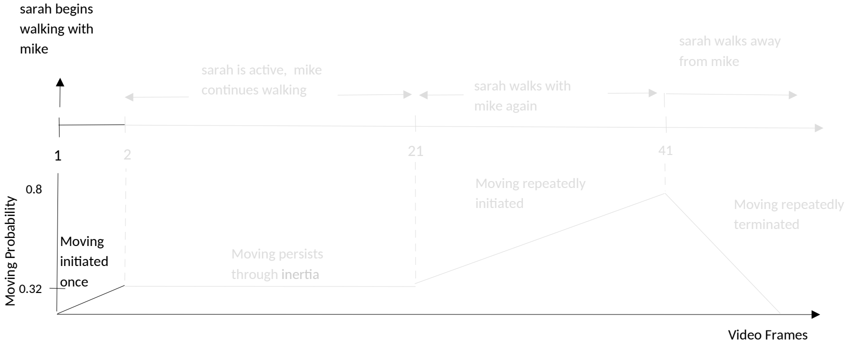
$$\textbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{happensAt}(walking(P_2),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$0.73 :: \textbf{happensAt}(walking(mike),\ 2).$$
$$0.55 :: \textbf{happensAt}(active(sarah),\ 2).\ \cdots$$

$$P(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 3)) =$$
$$P(\textbf{initiatedAt}(moving(mike, sarah) = \text{true}, 2) \lor$$
$$(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 2) \land$$
$$\neg\textbf{terminatedAt}(moving(mike, sarah) = \text{true}, 2)))$$
$$= 0 + 0.322 \times 1 - 0 \times 0.322 \times 1 = 0.322$$

Instantaneous Recognition

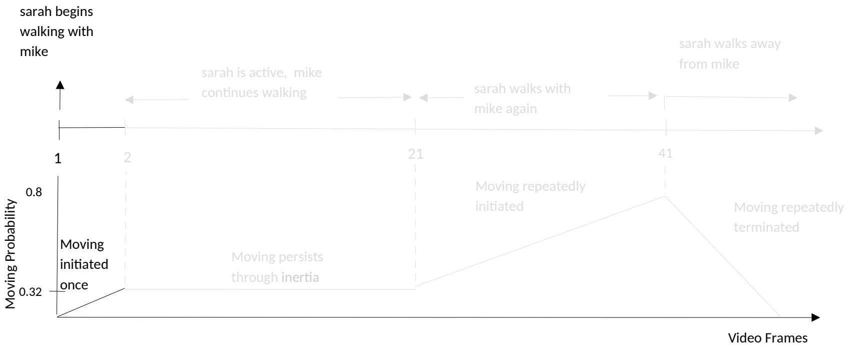


$$\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{happensAt}(\mathit{walking}(P_2),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$$

$$\mathit{0.45} :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ \mathit{20}).$$
$$\mathit{0.14} :: \textbf{happensAt}(\mathit{active}(\mathit{sarah}),\ \mathit{20}).$$

Instantaneous Recognition



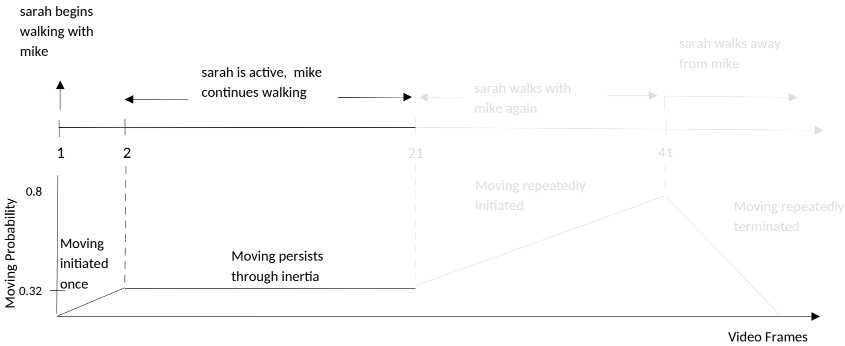
$\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),$
$\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T),$
$\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$
$\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$

$\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),$
$\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$

$0.45 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 20).$
$0.14 :: \textbf{happensAt}(\mathit{active}(\mathit{sarah}),\ 20).$

$$P(\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 21)) =$$
$$P(\textbf{initiatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 20) \lor$$
$$(\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 20) \land$$
$$\neg\textbf{terminatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 20)))$$
$$= 0 + 0.322 \times 1 - 0 \times 0.322 \times 1 = 0.322$$

Instantaneous Recognition



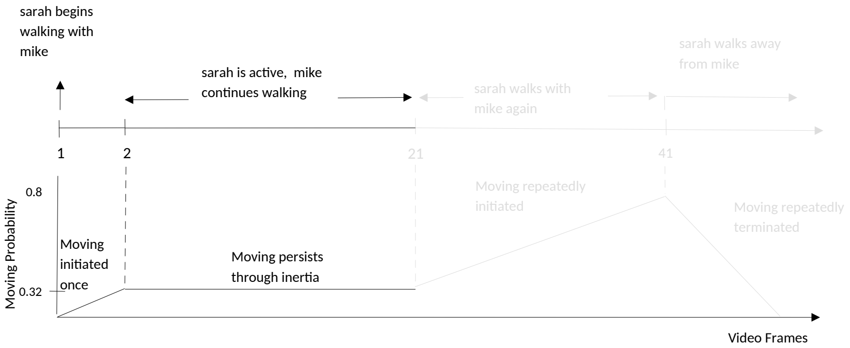
$$\begin{aligned}&\textbf{initiatedAt}(\textit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow\\ &\quad \textbf{happensAt}(\textit{walking}(P_1),\ T),\\ &\quad \textbf{happensAt}(\textit{walking}(P_2),\ T),\\ &\quad \textbf{holdsAt}(\textit{close}(P_1, P_2) = \text{true},\ T),\\ &\quad \textbf{holdsAt}(\textit{orientation}(P_1, P_2) = \text{true},\ T).\end{aligned}$$

$$\begin{aligned}&\textbf{terminatedAt}(\textit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow\\ &\quad \textbf{happensAt}(\textit{walking}(P_1),\ T),\\ &\quad \textbf{holdsAt}(\textit{close}(P_1, P_2) = \text{false},\ T).\end{aligned}$$

$$0.45 :: \textbf{happensAt}(\textit{walking}(\textit{mike}),\ 20).$$
$$0.14 :: \textbf{happensAt}(\textit{active}(\textit{sarah}),\ 20).$$

$$\begin{aligned}&P(\textbf{holdsAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 21)) =\\ &P(\textbf{initiatedAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 20) \vee\\ &\quad (\textbf{holdsAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 20) \wedge\\ &\quad \neg\textbf{terminatedAt}(\textit{moving}(\textit{mike}, \textit{sarah}) = \text{true}, 20)))\\ &= 0 + 0.322 \times 1 - 0 \times 0.322 \times 1 = 0.322\end{aligned}$$

Instantaneous Recognition

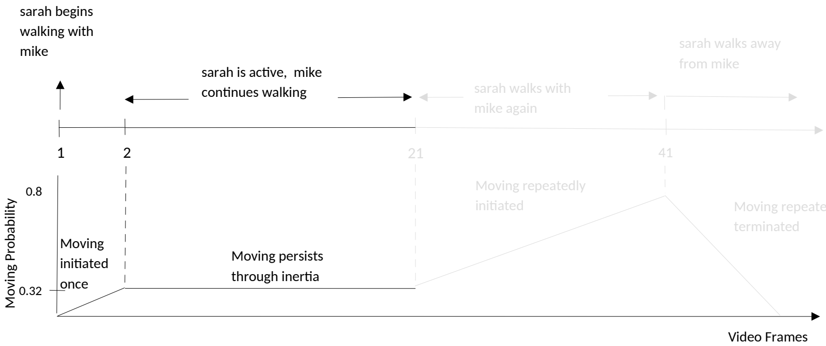


$$
\begin{aligned}
&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \mathrm{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \mathrm{true},\ T), \\
&\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \mathrm{true},\ T).
\end{aligned}
$$

$$
\begin{aligned}
&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \mathrm{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \mathrm{false},\ T).
\end{aligned}
$$

$$
\begin{aligned}
&0.39 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 21). \\
&0.28 :: \textbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ 21).\ \cdots
\end{aligned}
$$

Instantaneous Recognition



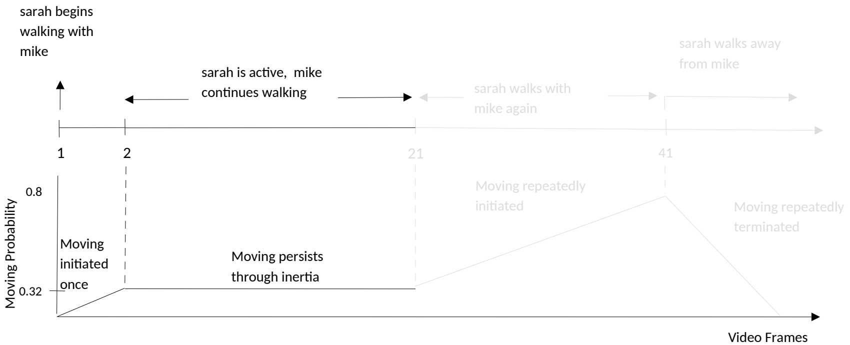
$$\mathbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\mathbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\mathbf{happensAt}(\mathit{walking}(P_2),\ T),$$
$$\mathbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$$
$$\mathbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$$

$$\mathbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\mathbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\mathbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$$

$$0.39 :: \mathbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 21).$$
$$0.28 :: \mathbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ 21).\ \cdots$$

$$P(\mathbf{initiatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 21)) =$$
$$P(\mathbf{happensAt}(\mathit{walking}(\mathit{mike}), 21)) \times$$
$$P(\mathbf{happensAt}(\mathit{walking}(\mathit{sarah}), 21)) \times$$
$$P(\mathbf{holdsAt}(\mathit{close}(\mathit{mike}, \mathit{sarah}) = \text{true}, 21)) \times$$
$$P(\mathbf{holdsAt}(\mathit{orientation}(\mathit{mike}, \mathit{sarah}) = \text{true}, 21))$$
$$= 0.39 \times 0.28 \times 1 \times 1 = 0.11$$

Instantaneous Recognition



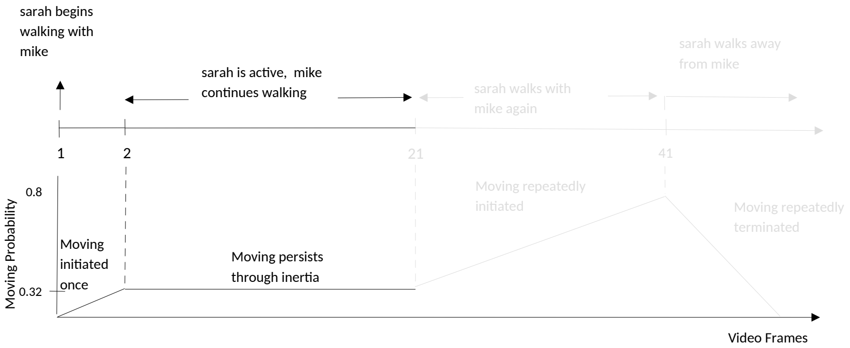
$$\textbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{happensAt}(walking(P_2),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$0.39 :: \textbf{happensAt}(walking(mike),\ 21).$$
$$0.28 :: \textbf{happensAt}(walking(sarah),\ 21).\ \cdots$$

$$P(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 22)) =$$
$$P(\textbf{initiatedAt}(moving(mike, sarah) = \text{true}, 21) \vee$$
$$(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 21) \wedge$$
$$\neg\textbf{terminatedAt}(moving(mike, sarah) = \text{true}, 21)))$$
$$= 0.11 + 0.322 \times 1 - 0.11 \times 0.322 \times 1 = 0.39$$

Instantaneous Recognition

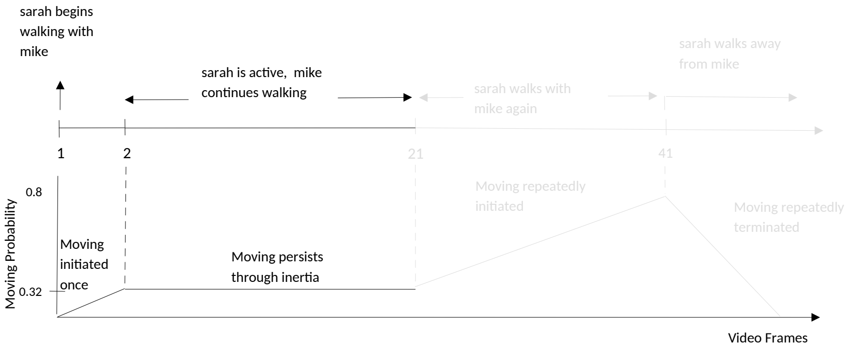


$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T), \\
&\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).
\end{aligned}$$

$$\begin{aligned}
&\mathit{0.28} :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ \mathit{40}). \\
&\mathit{0.18} :: \textbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ \mathit{40}).
\end{aligned}$$

Instantaneous Recognition



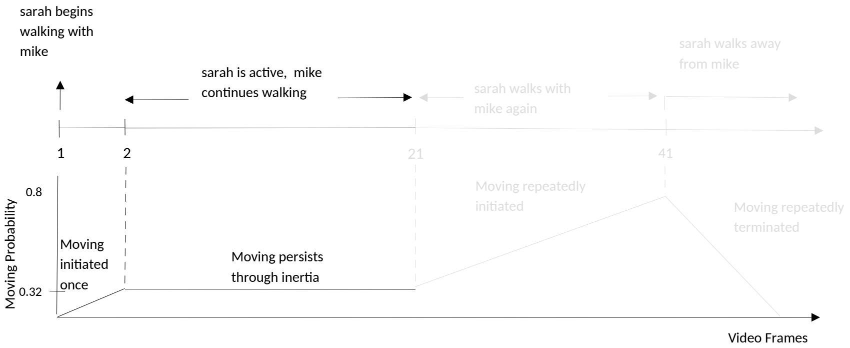
$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T), \\
&\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).
\end{aligned}$$

$$0.28 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 40).$$
$$0.18 :: \textbf{happensAt}(\mathit{walking}(\mathit{sarah}),\ 40).$$

$$\begin{aligned}
&P(\textbf{initiatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 40)) = \\
&\quad P(\textbf{happensAt}(\mathit{walking}(\mathit{mike}), 40)) \times \\
&\quad P(\textbf{happensAt}(\mathit{walking}(\mathit{sarah}), 40)) \times \\
&\quad P(\textbf{holdsAt}(\mathit{close}(\mathit{mike}, \mathit{sarah}) = \text{true}, 40)) \times \\
&\quad P(\textbf{holdsAt}(\mathit{orientation}(\mathit{mike}, \mathit{sarah}) = \text{true}, 40)) \\
&\quad = 0.28 \times 0.18 \times 1 \times 1 = 0.05
\end{aligned}$$

Instantaneous Recognition



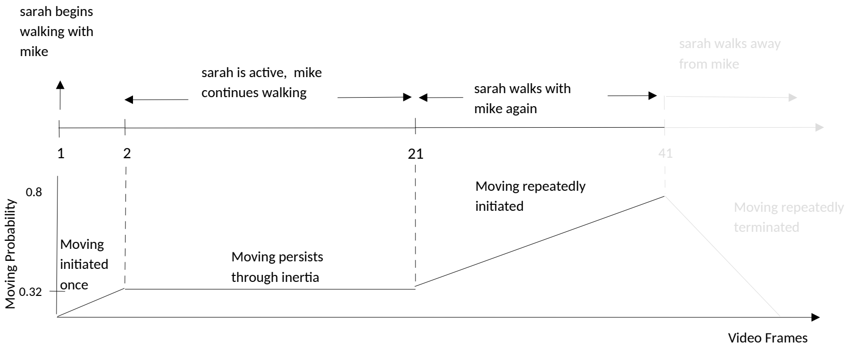
$$\textbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\quad \textbf{happensAt}(walking(P_1),\ T),$$
$$\quad \textbf{happensAt}(walking(P_2),\ T),$$
$$\quad \textbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\quad \textbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\quad \textbf{happensAt}(walking(P_1),\ T),$$
$$\quad \textbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$0.28 :: \textbf{happensAt}(walking(mike),\ 40).$$
$$0.18 :: \textbf{happensAt}(walking(sarah),\ 40).$$

$$P(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 41)) =$$
$$P(\textbf{initiatedAt}(moving(mike, sarah) = \text{true}, 40) \vee$$
$$\quad (\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 40) \wedge$$
$$\quad \neg\textbf{terminatedAt}(moving(mike, sarah) = \text{true}, 40)))$$
$$= 0.05 + 0.79 \times 1 - 0.05 \times 0.79 \times 1 = 0.80$$

Instantaneous Recognition



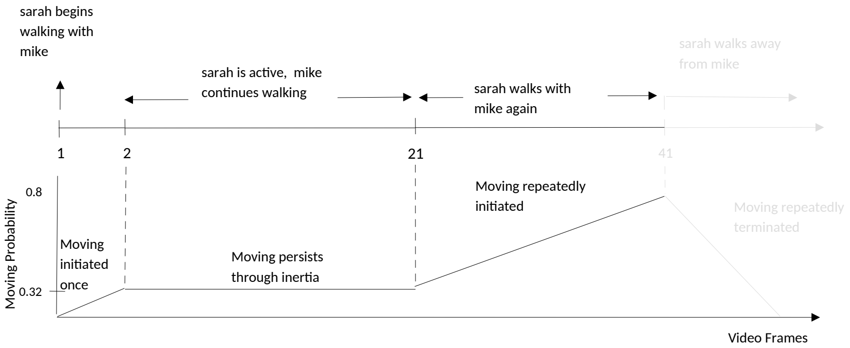
$$\textbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{happensAt}(walking(P_2),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(walking(P_1),\ T),$$
$$\textbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$0.28 :: \textbf{happensAt}(walking(mike),\ 40).$$
$$0.18 :: \textbf{happensAt}(walking(sarah),\ 40).$$

$$P(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 41)) =$$
$$P(\textbf{initiatedAt}(moving(mike, sarah) = \text{true}, 40) \vee$$
$$(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 40) \wedge$$
$$\neg\textbf{terminatedAt}(moving(mike, sarah) = \text{true}, 40)))$$
$$= 0.05 + 0.79 \times 1 - 0.05 \times 0.79 \times 1 = 0.80$$

Instantaneous Recognition

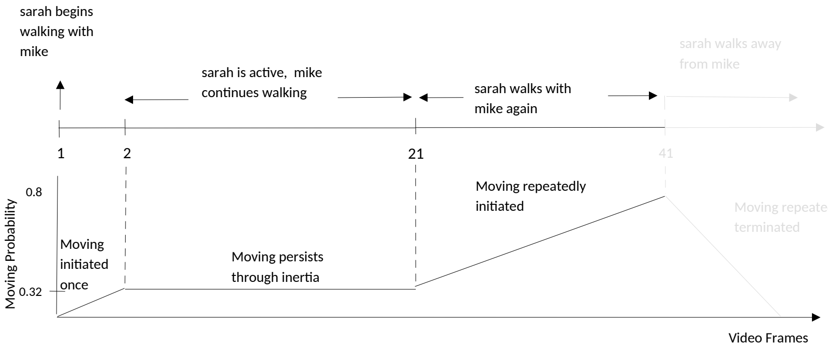


$$\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{happensAt}(\mathit{walking}(P_2),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$$
$$\textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$$

$$\mathit{0.18} :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ \mathit{41}).$$
$$\mathit{0.79} :: \textbf{happensAt}(\mathit{inactive}(\mathit{sarah}),\ \mathit{41}).\ \cdots$$

Instantaneous Recognition



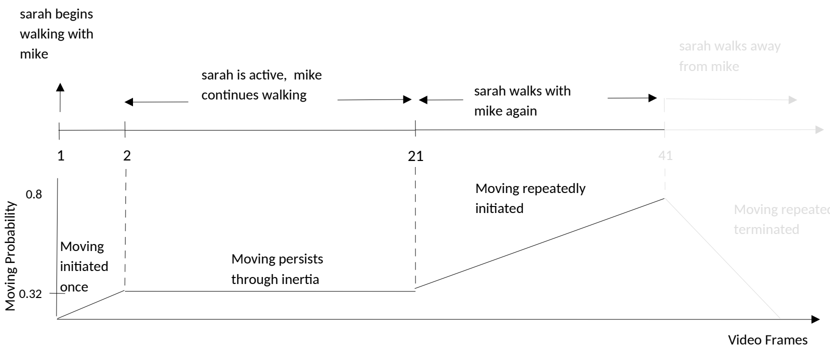
$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T), \\
&\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T), \\
&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).
\end{aligned}$$

$$\begin{aligned}
&0.18 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 41). \\
&0.79 :: \textbf{happensAt}(\mathit{inactive}(\mathit{sarah}),\ 41).\ \cdots
\end{aligned}$$

$$\begin{aligned}
&P(\textbf{terminatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 41)) = \\
&\quad P(\textbf{happensAt}(\mathit{walking}(\mathit{mike}), 41)) \times \\
&\quad P(\textbf{holdsAt}(\mathit{close}(\mathit{mike}, \mathit{sarah}) = \text{false}, 41)) \\
&\quad = 0.18 \times 1 = 0.18
\end{aligned}$$

Instantaneous Recognition



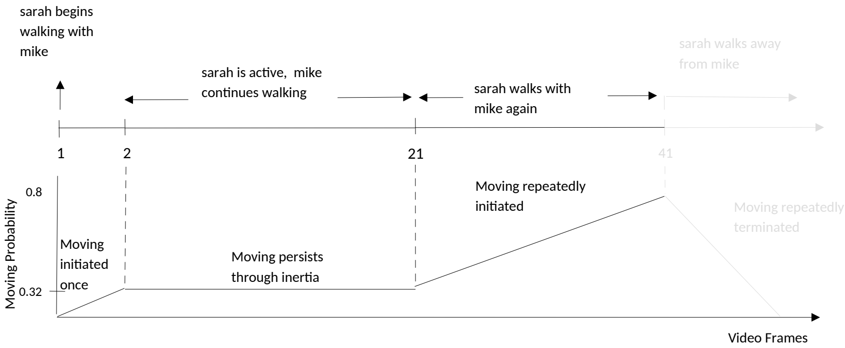
$$\textbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\quad \textbf{happensAt}(walking(P_1),\ T),$$
$$\quad \textbf{happensAt}(walking(P_2),\ T),$$
$$\quad \textbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\quad \textbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\quad \textbf{happensAt}(walking(P_1),\ T),$$
$$\quad \textbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$0.18 :: \textbf{happensAt}(walking(mike),\ 41).$$
$$0.79 :: \textbf{happensAt}(inactive(sarah),\ 41).\ \cdots$$

$$P(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 42)) =$$
$$P(\textbf{initiatedAt}(moving(mike, sarah) = \text{true}, 41) \vee$$
$$(\textbf{holdsAt}(moving(mike, sarah) = \text{true}, 41) \wedge$$
$$\neg\textbf{terminatedAt}(moving(mike, sarah) = \text{true}, 41)))$$
$$= 0 + 0.8 \times (1 - 0.18) - 0 \times 0.8 \times (1 - 0.18) = 0.66$$

Instantaneous Recognition

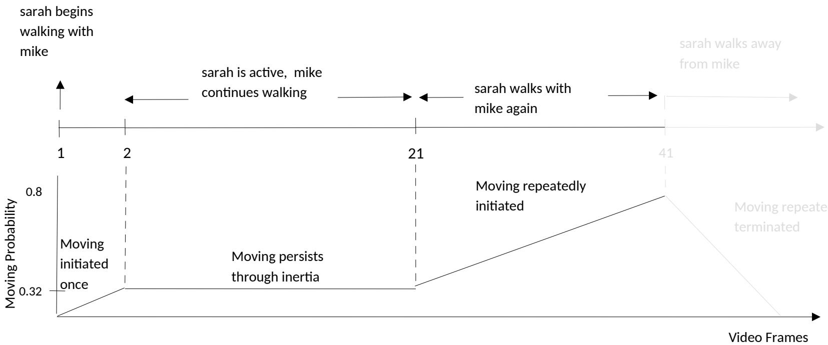


$$\mathbf{initiatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\mathbf{happensAt}(walking(P_1),\ T),$$
$$\mathbf{happensAt}(walking(P_2),\ T),$$
$$\mathbf{holdsAt}(close(P_1, P_2) = \text{true},\ T),$$
$$\mathbf{holdsAt}(orientation(P_1, P_2) = \text{true},\ T).$$

$$\mathbf{terminatedAt}(moving(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\mathbf{happensAt}(walking(P_1),\ T),$$
$$\mathbf{holdsAt}(close(P_1, P_2) = \text{false},\ T).$$

$$1.00 :: \mathbf{happensAt}(walking(mike),\ 49).$$
$$0.96 :: \mathbf{happensAt}(inactive(sarah),\ 49).$$

Instantaneous Recognition



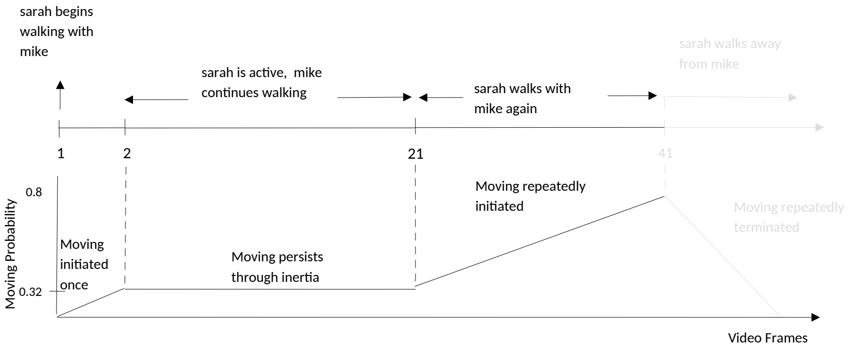
$$\begin{aligned}&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow\\&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),\\&\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T),\\&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),\\&\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).\end{aligned}$$

$$\begin{aligned}&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow\\&\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),\\&\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).\end{aligned}$$

$$\mathit{1.00} :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ \mathit{49}).$$
$$\mathit{0.96} :: \textbf{happensAt}(\mathit{inactive}(\mathit{sarah}),\ \mathit{49}).$$

$$\begin{aligned}&P(\textbf{terminatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, \mathit{49})) =\\&\quad P(\textbf{happensAt}(\mathit{walking}(\mathit{mike}), \mathit{49})) \times\\&\quad P(\textbf{holdsAt}(\mathit{close}(\mathit{mike}, \mathit{sarah}) = \text{false}, \mathit{49}))\\&\quad = \mathit{1} \times \mathit{1} = \mathit{1}\end{aligned}$$

Instantaneous Recognition



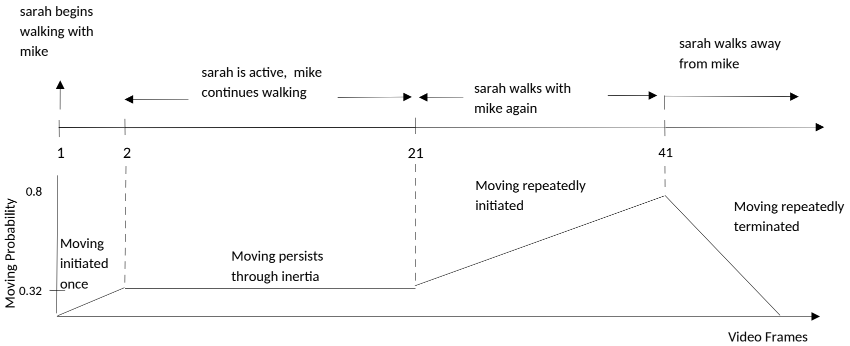
$$\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T),$$
$$\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$$
$$\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).$$

$$\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$$
$$\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),$$
$$\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$$

$$1.00 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 49).$$
$$0.96 :: \textbf{happensAt}(\mathit{inactive}(\mathit{sarah}),\ 49).$$

$$P(\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 50)) =$$
$$P(\textbf{initiatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 49) \vee$$
$$\quad (\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 49) \wedge$$
$$\quad \neg \textbf{terminatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 49)))$$
$$= 0 + 0.07 \times 0 - 0 \times 0.07 \times 0 = 0$$

Instantaneous Recognition



$$\begin{aligned}&\textbf{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow\\ &\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),\\ &\quad \textbf{happensAt}(\mathit{walking}(P_2),\ T),\\ &\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),\\ &\quad \textbf{holdsAt}(\mathit{orientation}(P_1, P_2) = \text{true},\ T).\end{aligned}$$

$$\begin{aligned}&\textbf{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow\\ &\quad \textbf{happensAt}(\mathit{walking}(P_1),\ T),\\ &\quad \textbf{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).\end{aligned}$$

$$\begin{aligned}&1.00 :: \textbf{happensAt}(\mathit{walking}(\mathit{mike}),\ 49).\\ &0.96 :: \textbf{happensAt}(\mathit{inactive}(\mathit{sarah}),\ 49).\end{aligned}$$

$$\begin{aligned}&P(\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 50)) =\\ &P(\textbf{initiatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 49) \vee\\ &\quad (\textbf{holdsAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 49) \wedge\\ &\quad \neg\textbf{terminatedAt}(\mathit{moving}(\mathit{mike}, \mathit{sarah}) = \text{true}, 49)))\\ &= 0 + 0.07 \times 0 - 0 \times 0.07 \times 0 = 0\end{aligned}$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	*0*	*0.5*	*0.7*	*0.9*	*0.4*	*0.1*	*0*	*0*	*0.5*	*1*

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	*0*	*0.5*	*0.7*	*0.9*	*0.4*	*0.1*	*0*	*0*	*0.5*	*1*
L	*−0.5*	*0*	*0.2*	*0.4*	*−0.1*	*−0.4*	*−0.5*	*−0.5*	*0*	*0.5*

$$L[i] = In[i] - \mathcal{T}$$

Interval-based Recognition

Time	1	2	3	4	5	6	7	8	9	10
In	*0*	*0.5*	*0.7*	*0.9*	*0.4*	*0.1*	*0*	*0*	*0.5*	*1*
L	-0.5	*0*	*0.2*	*0.4*	-0.1	-0.4	-0.5	-0.5	*0*	*0.5*

$$\sum_{i=s}^{e} L[i] \geq 0 \Leftrightarrow P([s,e]) \geq \mathcal{T}$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9

$$prefix[i] = \sum_{j=1}^{i} L[j]$$

Interval-based Recognition

Time	1	2	3	4	5	6	7	8	9	10
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	−0.5	0	0.2	0.4	−0.1	−0.4	−0.5	−0.5	0	0.5
prefix	−0.5	−0.5	−0.3	0.1	0	−0.4	−0.9	−1.4	−1.4	−0.9
dp										

$$dp[i] = \max_{i \leq j \leq n} (prefix[j])$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp										-0.9

$$dp[10] = \max_{10 \leq j \leq 10} (prefix[j])$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp									-0.9	-0.9

$$dp[9] = \max_{9 \leq j \leq 10} (prefix[j])$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp								-0.9	-0.9	-0.9

$$dp[8] = \max_{8 \leq j \leq 10} (prefix[j])$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp							-0.9	-0.9	-0.9	-0.9

$$dp[7] = \max_{7 \leq j \leq 10} (prefix[j])$$

Interval-based Recognition

Time	1	2	3	4	5	6	7	8	9	10
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp						-0.4	-0.9	-0.9	-0.9	-0.9

$$dp[6] = \max_{6 \leq j \leq 10} (prefix[j])$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	−0.5	0	0.2	0.4	−0.1	−0.4	−0.5	−0.5	0	0.5
prefix	−0.5	−0.5	−0.3	0.1	0	−0.4	−0.9	−1.4	−1.4	−0.9
dp	0.1	0.1	0.1	0.1	0	−0.4	−0.9	−0.9	−0.9	−0.9

$$dp[i] = \max_{i \leq j \leq 10}(prefix[j])$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	−0.5	0	0.2	0.4	−0.1	−0.4	−0.5	−0.5	0	0.5
prefix	−0.5	−0.5	−0.3	0.1	0	−0.4	−0.9	−1.4	−1.4	−0.9
dp	0.1	0.1	0.1	0.1	0	−0.4	−0.9	−0.9	−0.9	−0.9

$$dprange[s, e] = dp[e] - prefix[s-1] \textbf{ if } s > 1$$

$$= dp[e] \textbf{ if } s = 1$$

Interval-based Recognition

Time	1	2	3	4	5	6	7	8	9	10
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	−0.5	0	0.2	0.4	−0.1	−0.4	−0.5	−0.5	0	0.5
prefix	−0.5	−0.5	−0.3	0.1	0	−0.4	−0.9	−1.4	−1.4	−0.9
dp	0.1	0.1	0.1	0.1	0	−0.4	−0.9	−0.9	−0.9	−0.9

$$dprange[s, e] = dp[e] - prefix[s-1] \textbf{ if } s > 1$$

$$= dp[e] \textbf{ if } s = 1$$

$$dprange[s, e] \geq 0 \Rightarrow \exists e^* : e^* \geq e,\ P([s, e^*]) \geq \mathcal{T}$$

Interval-based Recognition

$\Uparrow\Downarrow$

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

Interval-based Recognition

Time	⇑⇓ 1	2	3	4	5	6	7	8	9	10
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	−0.5	0	0.2	0.4	−0.1	−0.4	−0.5	−0.5	0	0.5
prefix	−0.5	−0.5	−0.3	0.1	0	−0.4	−0.9	−1.4	−1.4	−0.9
dp	0.1	0.1	0.1	0.1	0	−0.4	−0.9	−0.9	−0.9	−0.9

$$dprange[1,1] = dp[1] = 0.1 \geq 0$$

Interval-based Recognition

	⇑	⇓								
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

Interval-based Recognition

	⇑	⇓								
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[1, 2] = dp[2] = 0.1 \geq 0$$

Interval-based Recognition

	$\Uparrow$		$\Downarrow$							
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[1,3] = dp[3] = 0.1 \geq 0$$

Interval-based Recognition

	$\Uparrow$			$\Downarrow$						
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[1, 4] = dp[4] = 0.1 \geq 0$$

Interval-based Recognition

	⇑				⇓					
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[1, 5] = dp[5] = 0 \geq 0$$

Interval-based Recognition

	⇑					⇓				
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[1,6] = dp[6] = -0.4 < 0$$

Interval-based Recognition

	⇑					⇓				
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[1,6] = dp[6] = -0.4 < 0$$

Interval-based Recognition

		$\Uparrow$				$\Downarrow$				
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[2,6] = dp[6] - prefix[1] = 0.1 \geq 0$$

Interval-based Recognition

		$\Uparrow$					$\Downarrow$			
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[2,7] = dp[7] - prefix[1] = -0.4 < 0$$

Interval-based Recognition

		$\Uparrow$					$\Downarrow$			
Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

$$dprange[2,7] = dp[7] - prefix[1] = -0.4 < 0$$

Interval-based Recognition

Time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
In	0	0.5	0.7	0.9	0.4	0.1	0	0	0.5	1
L	-0.5	0	0.2	0.4	-0.1	-0.4	-0.5	-0.5	0	0.5
prefix	-0.5	-0.5	-0.3	0.1	0	-0.4	-0.9	-1.4	-1.4	-0.9
dp	0.1	0.1	0.1	0.1	0	-0.4	-0.9	-0.9	-0.9	-0.9

Interval-based Recognition[1]

Interval Computation Correctness

An interval is computed iff it is a probabilistic maximal interval.

[1] Artikis et al, A Probabilistic Interval-based Event Calculus for Activity Recognition. Annals of Mathematics and Artificial Intelligence, 2021.

Interval-based Recognition[1]

Interval Computation Correctness

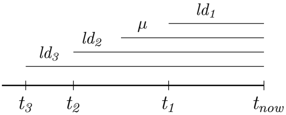
An interval is computed iff it is a probabilistic maximal interval.

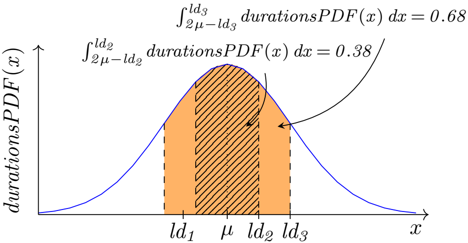
Complexity

The computation of probabilistic maximal intervals is linear to the dataset size.

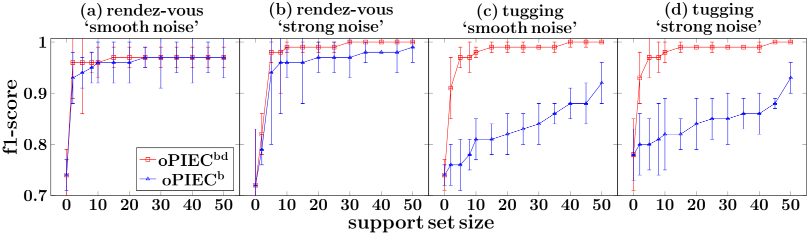
[1] Artikis et al, A Probabilistic Interval-based Event Calculus for Activity Recognition. Annals of Mathematics and Artificial Intelligence, 2021.

Deletion Probabilities of Support Set Elements





oPIEC: Experimental Results



oPIEC: Experimental Results

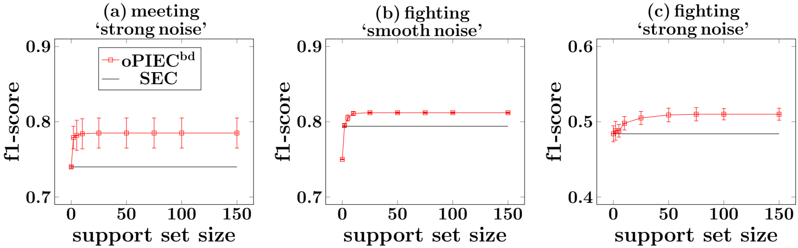
(a) Total run-times of oPIEC^{bd}, oPIEC^{b} and PIEC in seconds when processing data streams of increasing size.

total stream size (number of time-points)	**1K**	**2K**	**4K**	**8K**
PIEC	*1.92 ± 0.32*	*7.53 ± 1.24*	*29.76 ± 4.9*	*134.63 ± 22*
oPIEC^{b}	**0.09 ± 0.02**	**0.19 ± 0.05**	**0.38 ± 0.1**	**0.7 ± 0.2**
oPIEC^{bd}	**0.09 ± 0.02**	**0.19 ± 0.05**	*0.39 ± 0.1*	*0.72±0.23*

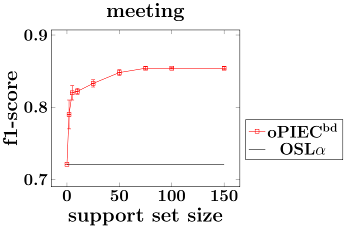
(b) Total run-times of oPIEC^{bd} and oPIEC^{b} in seconds as the support set size increases.

support set size (number of elements)	**50**	**100**	**200**	**400**
oPIEC^{b}	**0.7 ± 0.21**	**1.27 ± 0.5**	**2.4 ± 1.1**	**4.79 ± 2.41**
oPIEC^{bd}	**0.7 ± 0.23**	**1.27 ± 0.53**	**2.4 ± 1.1**	**4.79 ± 2.41**

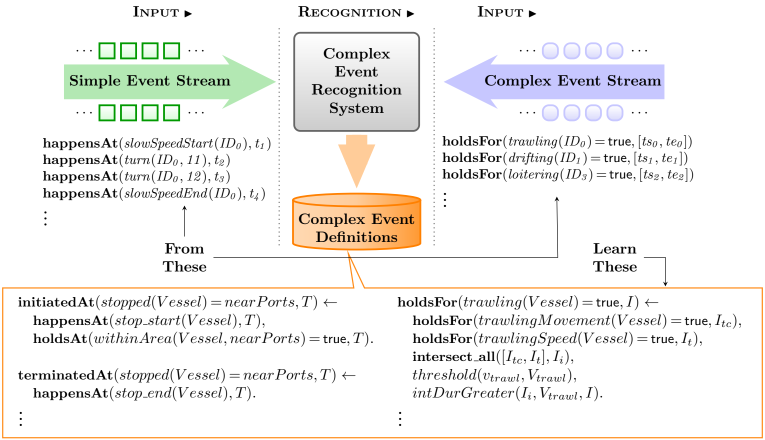
oPIEC: Experimental Results



oPIEC: Experimental Results



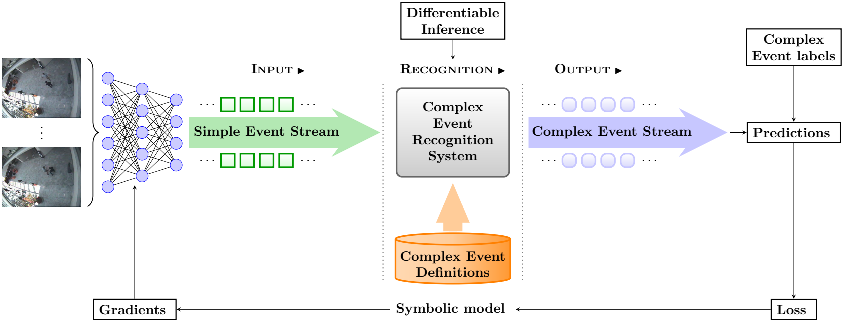
Machine Learning for Complex Event Recognition



Katzouris et al, Online Learning Probabilistic Event Calculus Theories in Answer Set Programming. Theory and Practice of Logic Programming, 2023.

Michelioudakis et al, Online semi-supervised learning of composite event rules by combining structure and mass-based predicate similarity. Machine Learning, 2024.

Neuro-Symbolic Complex Event Recognition



Marra et al, From statistical relational to neurosymbolic artificial intelligence: A survey. Artificial Intelligence, 2024.