

Stream Reasoning with Cycles

Periklis Mantenoglou^{2,1} Manolis Pitsikalis⁴
Alexander Artikis^{3,1}

¹NCSR Demokritos, Greece

²National and Kapodistrian University of Athens, Greece

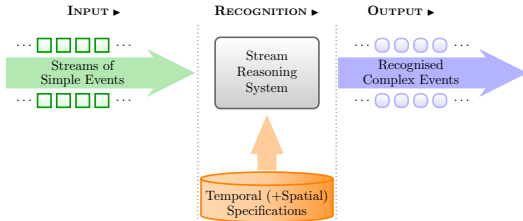
³University of Piraeus, Greece

⁴University of Liverpool, UK

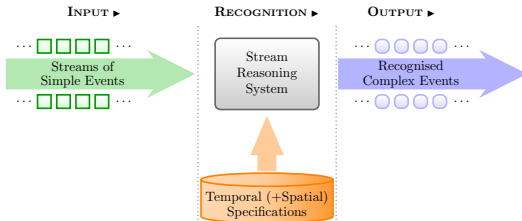
<http://cer.iit.demokritos.gr/>



Stream Reasoning



Stream Reasoning



[https://cer.iit.demokritos.gr \(maritime\)](https://cer.iit.demokritos.gr (maritime))

Event Calculus

- A **logic programming language** for representing and reasoning about events and their effects.
- Key components:
 - **event** (typically instantaneous).
 - **fluent**: a property that may have different values at different points in time.

Event Calculus

- A **logic programming language** for representing and reasoning about events and their effects.
- Key components:
 - **event** (typically instantaneous).
 - **fluent**: a property that may have different values at different points in time.
- Built-in representation of **inertia**:
 - $F = V$ holds at a particular time-point if $F = V$ has been *initiated* by an event at some earlier time-point, and not *terminated* by another event in the meantime.

Run-Time 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
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

Fluent-Value Pair Specification

Definition:

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

...

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

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

...

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

where

conditions:
 $0-K$ **happensAt**(E_k, T),
 $0-M$ **holdsAt**($F_m = V_m, T$),
 $0-N$ atemporal-constraint _{n}

Fluent-Value Pair Computation

Definition:

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

...

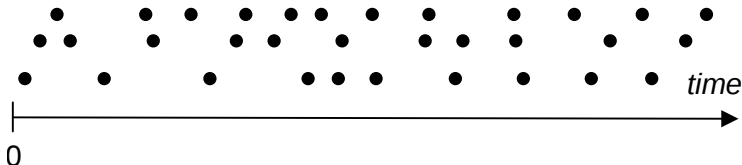
initiatedAt($F = V$, T) $\leftarrow$
 happensAt(E_{In_i} , T),
 [conditions]

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

...

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

Reasoning:



Fluent-Value Pair Computation

Definition:

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

...

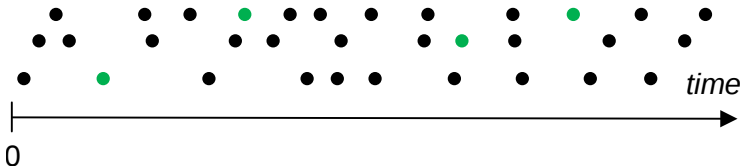
initiatedAt($F = V$, T) $\leftarrow$
 happensAt(E_{In_i} , T),
 [conditions]

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

...

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

Reasoning:



Fluent-Value Pair Computation

Definition:

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

• • •

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

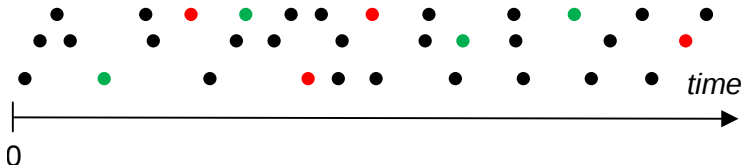
[conditions]

...

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

$$[\text{conditions}]$$

Reasoning:



Fluent-Value Pair Computation

Definition:

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

...

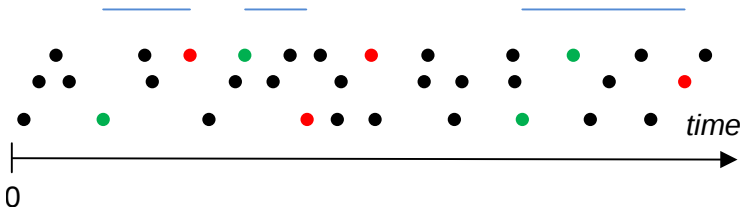
initiatedAt($F = V$, T) $\leftarrow$
happensAt(E_{In_i} , T),
[conditions]

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

...

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

Reasoning: **holdsFor**($F = V$, I)



Cyclic Dependencies in Temporal Specifications

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

Cyclic Dependencies in Temporal Specifications

initiatedAt($status(M) = proposed, T$) $\leftarrow$
 happensAt($propose(P, M), T$),
 holdsAt($status(M) = null, T$).
initiatedAt($status(M) = voting, T$) $\leftarrow$
 happensAt($second(S, M), T$),
 holdsAt($status(M) = proposed, T$).

Cyclic Dependencies in Temporal Specifications

initiatedAt(*status*(*M*) = *proposed*, *T*) $\leftarrow$
 happensAt(*propose*(*P*, *M*), *T*),
 holdsAt(*status*(*M*) = *null*, *T*).
initiatedAt(*status*(*M*) = *voting*, *T*) $\leftarrow$
 happensAt(*second*(*S*, *M*), *T*),
 holdsAt(*status*(*M*) = *proposed*, *T*).
initiatedAt(*status*(*M*) = *voted*, *T*) $\leftarrow$
 happensAt(*close_ballot*(*C*, *M*), *T*),
 holdsAt(*status*(*M*) = *voting*, *T*).

Cyclic Dependencies in Temporal Specifications

initiatedAt(*status*(*M*) = *proposed*, *T*) $\leftarrow$
 happensAt(*propose*(*P*, *M*), *T*),
 holdsAt(*status*(*M*) = *null*, *T*).

initiatedAt(*status*(*M*) = *voting*, *T*) $\leftarrow$
 happensAt(*second*(*S*, *M*), *T*),
 holdsAt(*status*(*M*) = *proposed*, *T*).

initiatedAt(*status*(*M*) = *voted*, *T*) $\leftarrow$
 happensAt(*close_ballot*(*C*, *M*), *T*),
 holdsAt(*status*(*M*) = *voting*, *T*).

initiatedAt(*status*(*M*) = *null*, *T*) $\leftarrow$
 happensAt(*declare*(*C*, *M*, *Res*), *T*),
 holdsAt(*status*(*M*) = *voted*, *T*).

Cyclic Dependencies in Temporal Specifications

initiatedAt(*status*(*M*) = *proposed*, *T*) $\leftarrow$
 happensAt(*propose*(*P*, *M*), *T*),
 holdsAt(*status*(*M*) = *null*, *T*).

initiatedAt(*status*(*M*) = *voting*, *T*) $\leftarrow$
 happensAt(*second*(*S*, *M*), *T*),
 holdsAt(*status*(*M*) = *proposed*, *T*).

initiatedAt(*status*(*M*) = *voted*, *T*) $\leftarrow$
 happensAt(*close_ballot*(*C*, *M*), *T*),
 holdsAt(*status*(*M*) = *voting*, *T*).

initiatedAt(*status*(*M*) = *null*, *T*) $\leftarrow$
 happensAt(*declare*(*C*, *M*, *Res*), *T*),
 holdsAt(*status*(*M*) = *voted*, *T*).



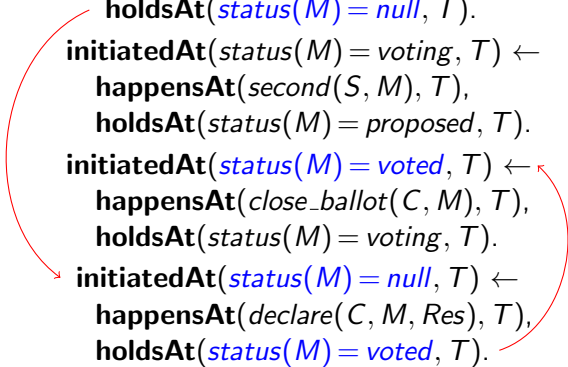
Cyclic Dependencies in Temporal Specifications

initiatedAt(*status*(*M*) = *proposed*, *T*) $\leftarrow$
 happensAt(*propose*(*P*, *M*), *T*),
 holdsAt(*status*(*M*) = *null*, *T*).

initiatedAt(*status*(*M*) = *voting*, *T*) $\leftarrow$
 happensAt(*second*(*S*, *M*), *T*),
 holdsAt(*status*(*M*) = *proposed*, *T*).

initiatedAt(*status*(*M*) = *voted*, *T*) $\leftarrow$
 happensAt(*close_ballot*(*C*, *M*), *T*),
 holdsAt(*status*(*M*) = *voting*, *T*).

initiatedAt(*status*(*M*) = *null*, *T*) $\leftarrow$
 happensAt(*declare*(*C*, *M*, *Res*), *T*),
 holdsAt(*status*(*M*) = *voted*, *T*).



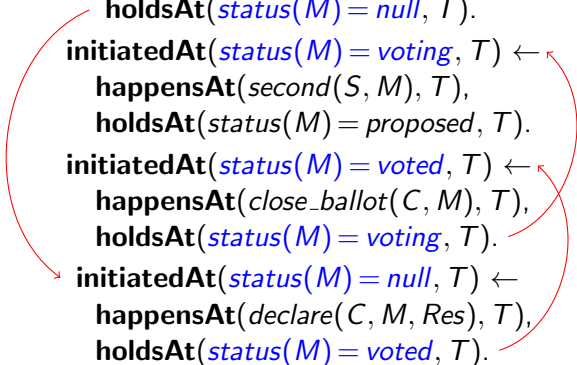
Cyclic Dependencies in Temporal Specifications

initiatedAt(*status*(*M*) = *proposed*, *T*) $\leftarrow$
 happensAt(*propose*(*P*, *M*), *T*),
 holdsAt(*status*(*M*) = *null*, *T*).

initiatedAt(*status*(*M*) = *voting*, *T*) $\leftarrow$
 happensAt(*second*(*S*, *M*), *T*),
 holdsAt(*status*(*M*) = *proposed*, *T*).

initiatedAt(*status*(*M*) = *voted*, *T*) $\leftarrow$
 happensAt(*close_ballot*(*C*, *M*), *T*),
 holdsAt(*status*(*M*) = *voting*, *T*).

initiatedAt(*status*(*M*) = *null*, *T*) $\leftarrow$
 happensAt(*declare*(*C*, *M*, *Res*), *T*),
 holdsAt(*status*(*M*) = *voted*, *T*).



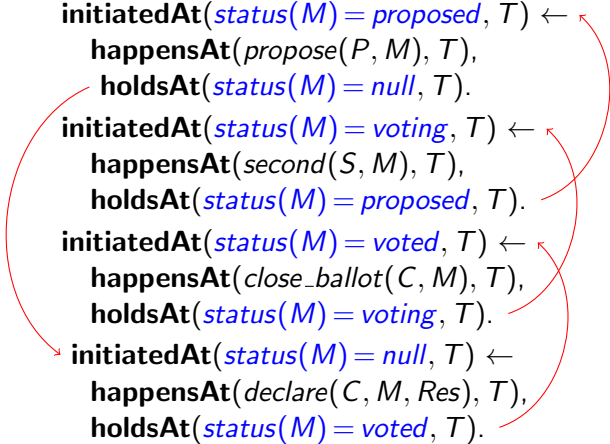
Cyclic Dependencies in Temporal Specifications

initiatedAt(*status*(*M*) = *proposed*, *T*) $\leftarrow$
 happensAt(*propose*(*P*, *M*), *T*),
 holdsAt(*status*(*M*) = *null*, *T*).

initiatedAt(*status*(*M*) = *voting*, *T*) $\leftarrow$
 happensAt(*second*(*S*, *M*), *T*),
 holdsAt(*status*(*M*) = *proposed*, *T*).

initiatedAt(*status*(*M*) = *voted*, *T*) $\leftarrow$
 happensAt(*close_ballot*(*C*, *M*), *T*),
 holdsAt(*status*(*M*) = *voting*, *T*).

initiatedAt(*status*(*M*) = *null*, *T*) $\leftarrow$
 happensAt(*declare*(*C*, *M*, *Res*), *T*),
 holdsAt(*status*(*M*) = *voted*, *T*).

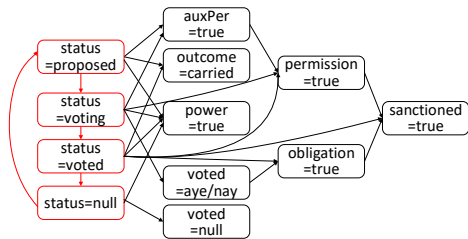


RTEC_o: Run-Time Event Calculus for Cyclic Dependencies

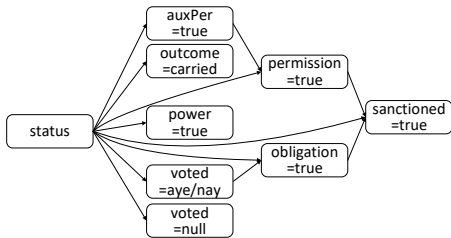
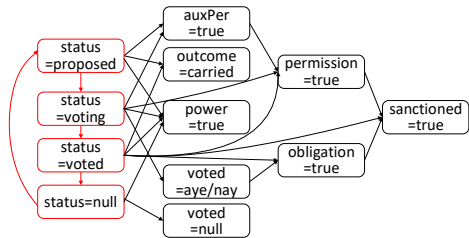
RTEC_o: an extension of RTEC for efficient reasoning over specifications with **cyclic dependencies**

- **Formal** & **open-source** computational framework
- **Locally stratified** specifications
- **Incremental caching** of intermediate computations
- **Scalable** in large, real-world data streams

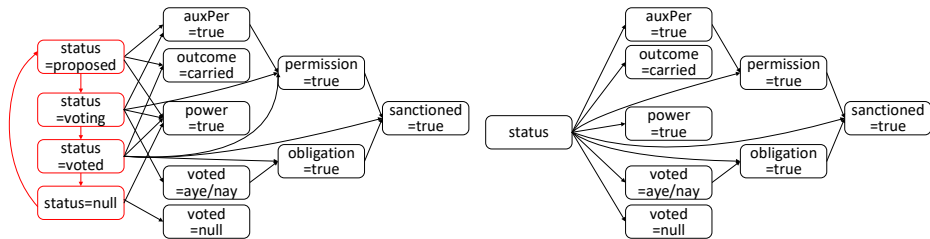
Semantics



Semantics



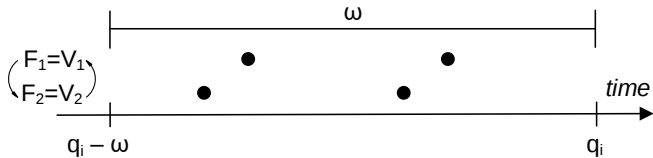
Semantics



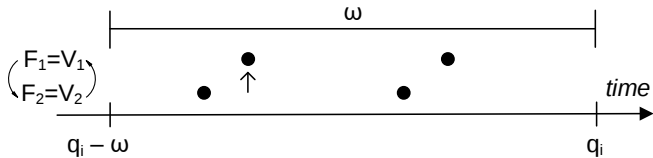
Semantics

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

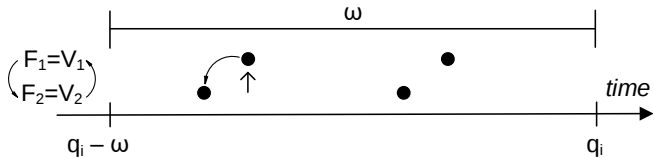
Handling Cyclic Dependencies



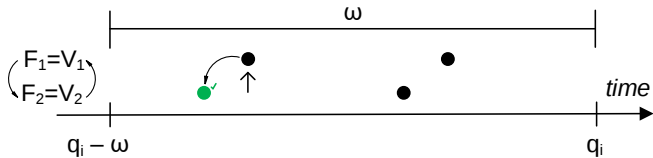
Handling Cyclic Dependencies



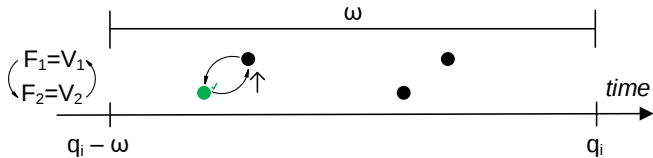
Handling Cyclic Dependencies



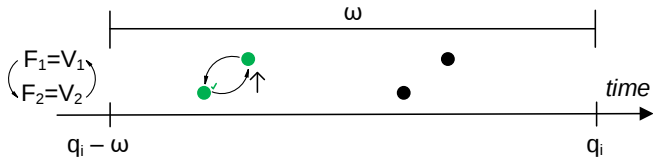
Handling Cyclic Dependencies



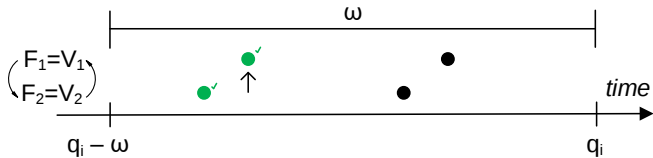
Handling Cyclic Dependencies



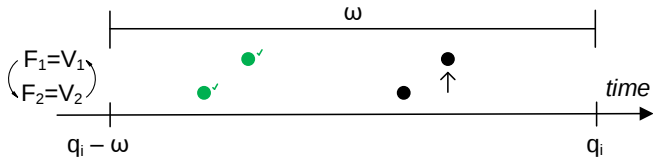
Handling Cyclic Dependencies



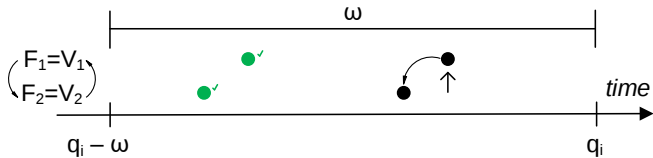
Handling Cyclic Dependencies



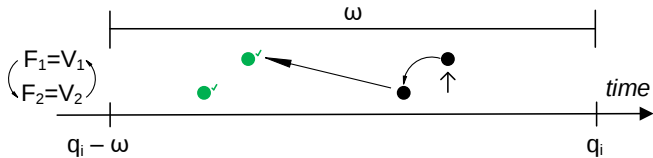
Handling Cyclic Dependencies



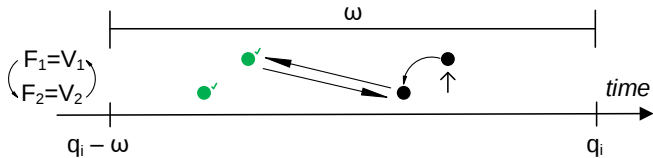
Handling Cyclic Dependencies



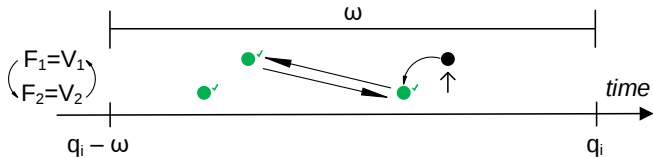
Handling Cyclic Dependencies



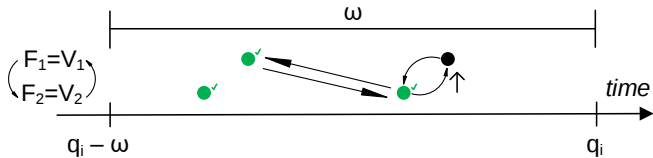
Handling Cyclic Dependencies



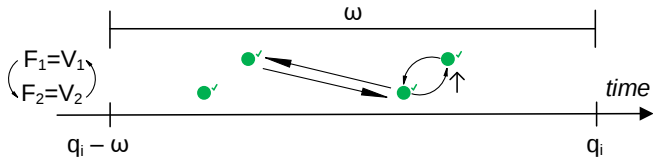
Handling Cyclic Dependencies



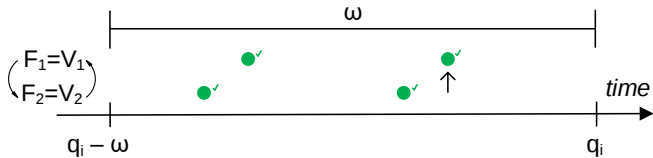
Handling Cyclic Dependencies



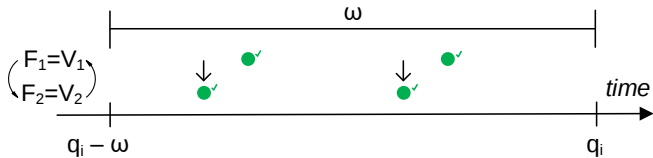
Handling Cyclic Dependencies



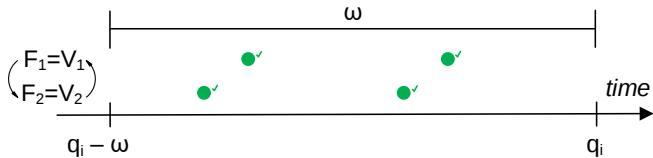
Handling Cyclic Dependencies



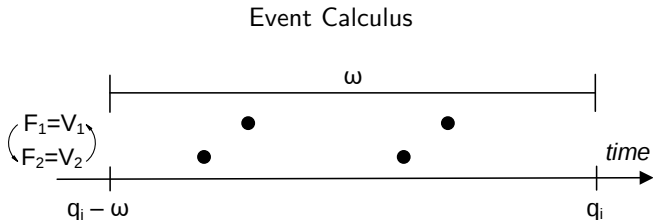
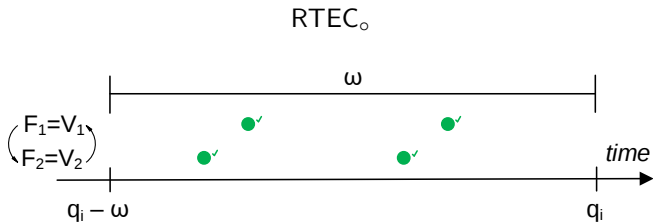
Handling Cyclic Dependencies



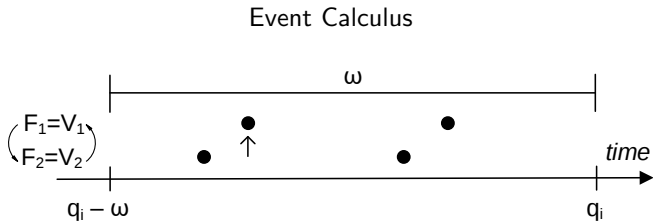
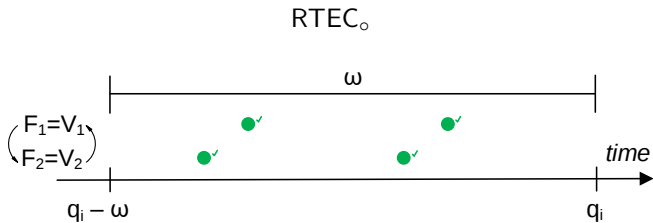
Handling Cyclic Dependencies



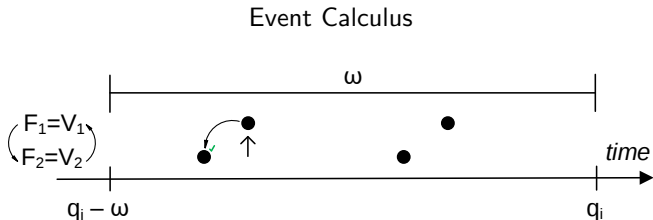
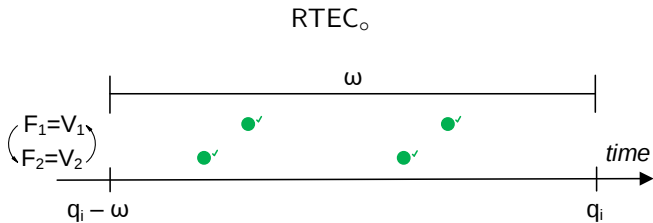
Handling Cyclic Dependencies



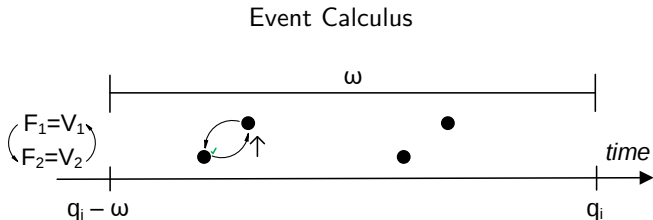
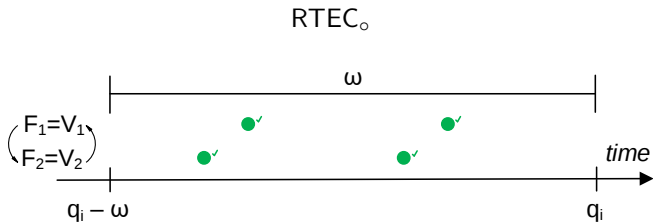
Handling Cyclic Dependencies



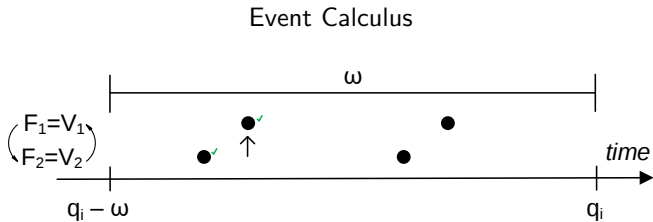
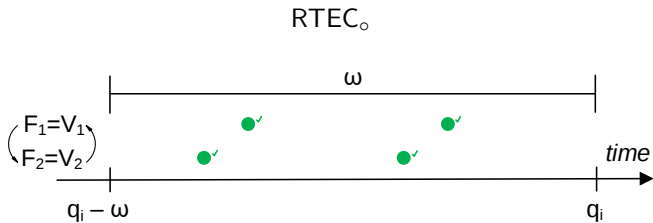
Handling Cyclic Dependencies



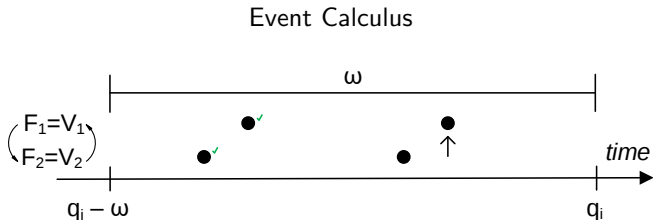
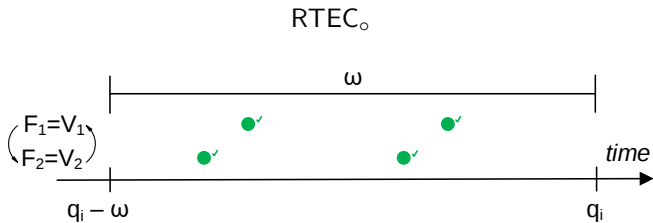
Handling Cyclic Dependencies



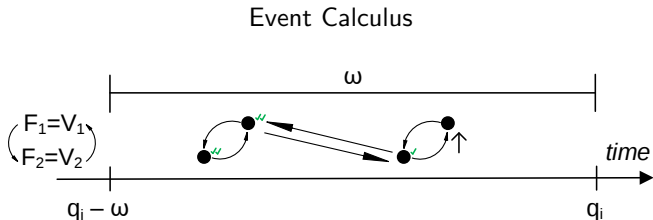
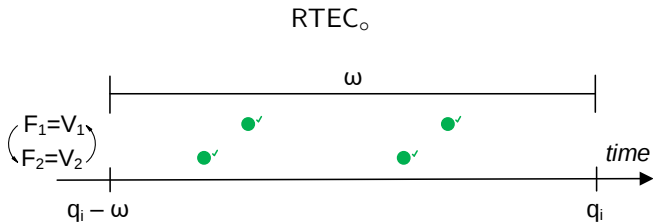
Handling Cyclic Dependencies



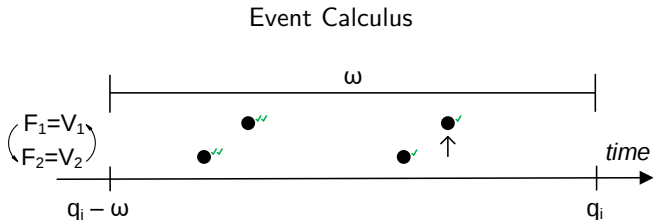
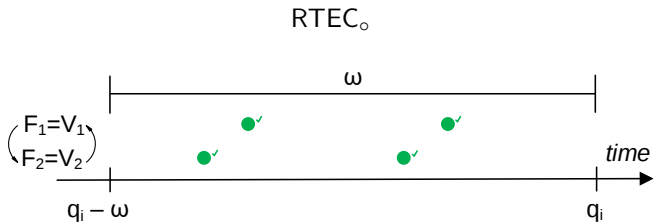
Handling Cyclic Dependencies



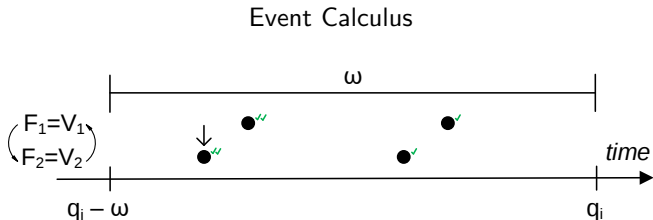
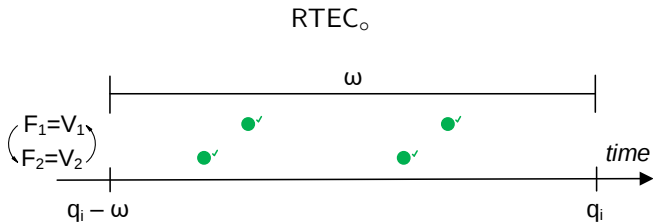
Handling Cyclic Dependencies



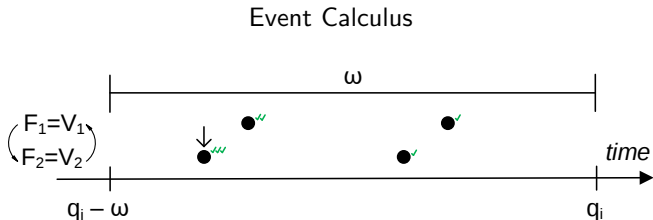
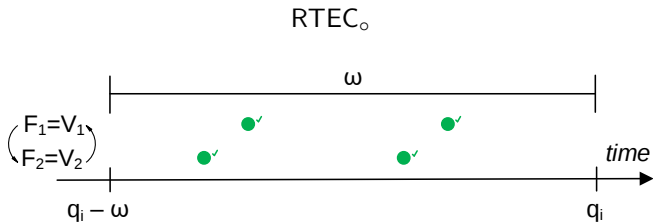
Handling Cyclic Dependencies



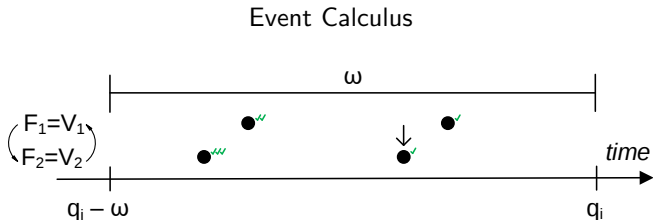
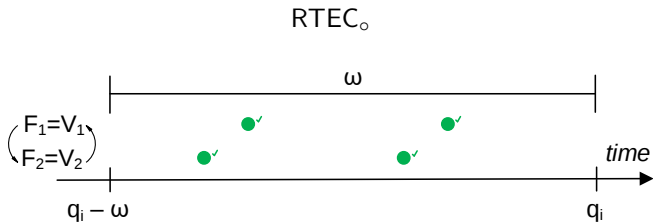
Handling Cyclic Dependencies



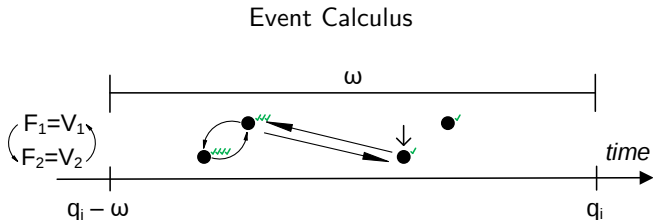
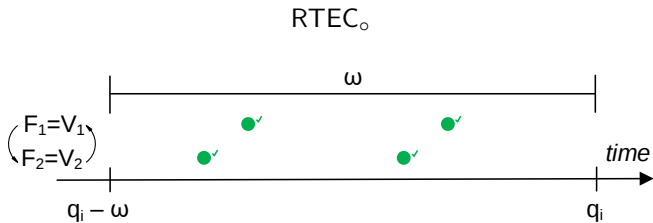
Handling Cyclic Dependencies



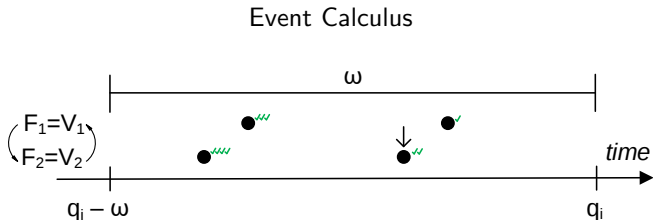
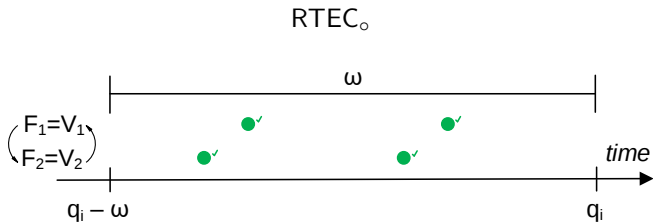
Handling Cyclic Dependencies



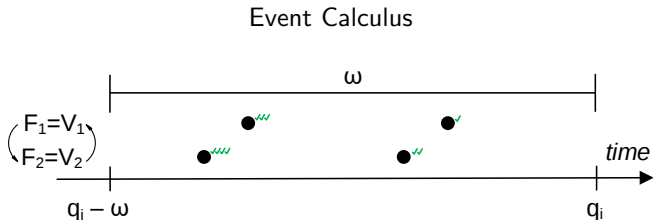
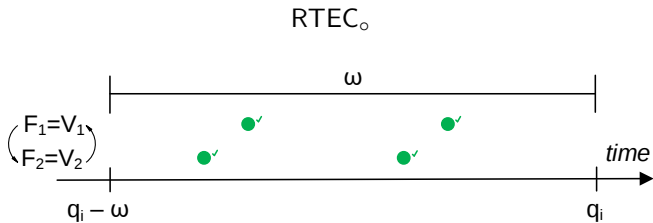
Handling Cyclic Dependencies



Handling Cyclic Dependencies

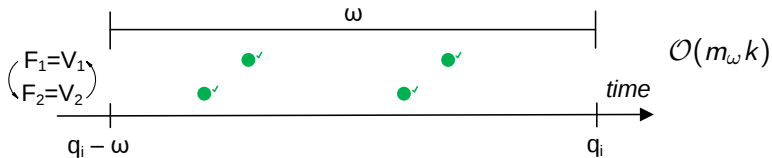


Handling Cyclic Dependencies

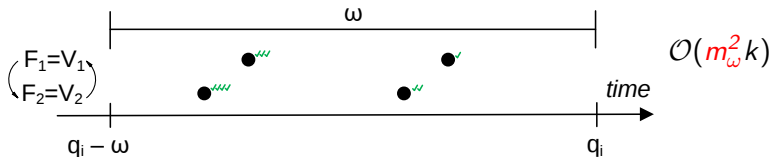


Handling Cyclic Dependencies

RTEC_o



Event Calculus



Experimental Setup

Multi-Agent Systems: Voting & NetBill

- Compute, e.g., normative positions of agents.

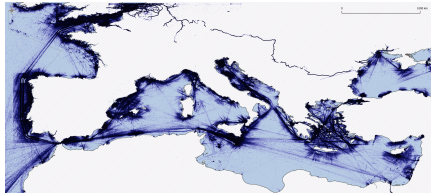
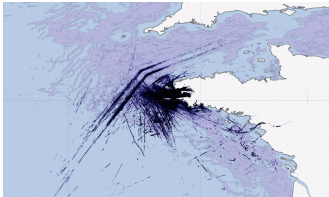
Experimental Setup

Multi-Agent Systems: Voting & NetBill

- Compute, e.g., normative positions of agents.

Maritime Situational Awareness

- Recognise dangerous, illegal and suspicious vessel activity.



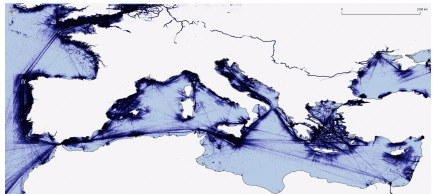
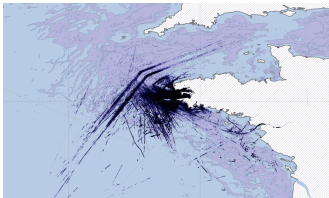
Experimental Setup

Multi-Agent Systems: Voting & NetBill

- Compute, e.g., normative positions of agents.

Maritime Situational Awareness

- Recognise dangerous, illegal and suspicious vessel activity.

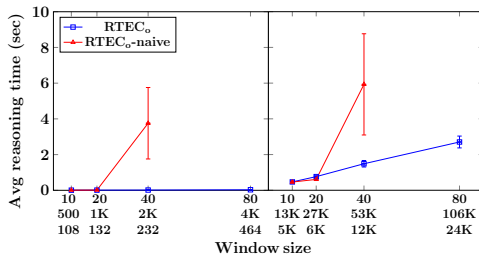


Code, Data & Temporal Specifications

<https://github.com/aartikis/RTEC>

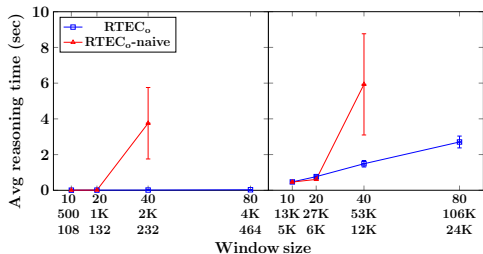
Experimental Results

Voting & NetBill

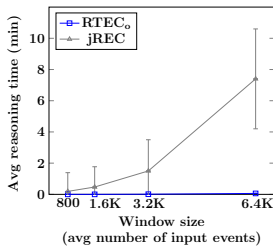


Experimental Results

Voting & NetBill

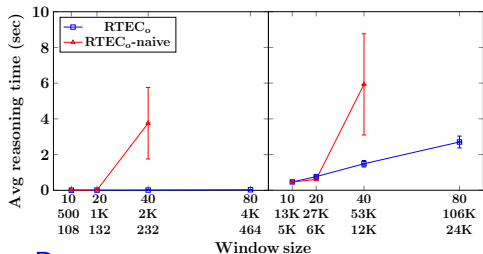


Voting: *status* fluent

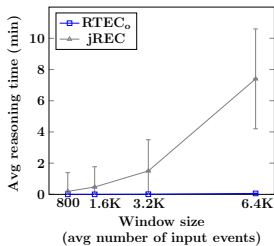


Experimental Results

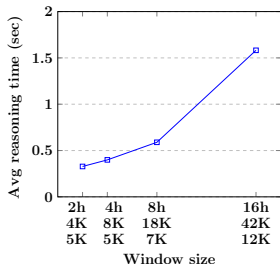
Voting & NetBill



Voting: *status* fluent

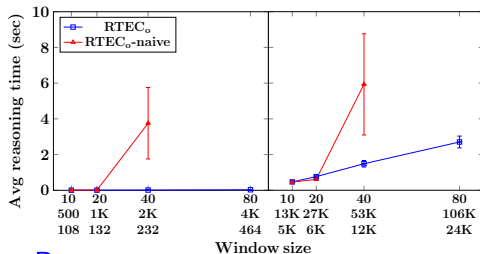


Brest

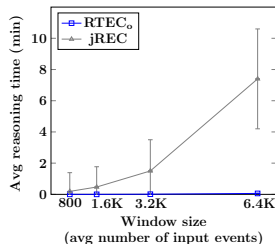


Experimental Results

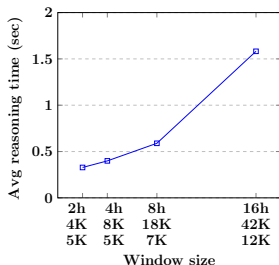
Voting & NetBill



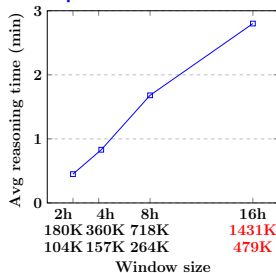
Voting: *status* fluent



Brest



European seas



Summary & Further Work

RTEC_o

- A formal, open-source stream reasoning system
- Efficient treatment of **cyclic dependencies**
- Locally stratified specifications
- **Reproducible** empirical evaluation on large data streams

Summary & Further Work

RTEC_o

- A formal, open-source stream reasoning system
- Efficient treatment of **cyclic dependencies**
- Locally stratified specifications
- **Reproducible** empirical evaluation on large data streams

Further Work

- Efficient treatment of **deadlines**
- Integrating RTEC_o in **neuro-symbolic frameworks**

Summary & Further Work

RTEC_o

- A formal, open-source stream reasoning system
- Efficient treatment of **cyclic dependencies**
- Locally stratified specifications
- **Reproducible** empirical evaluation on large data streams

Further Work

- Efficient treatment of **deadlines**
- Integrating RTEC_o in **neuro-symbolic frameworks**

Resources

<https://cer.iit.demokritos.gr/>