02917 Advanced Topics in Embedded Systems

Brief Introduction to Duration Calculus

Michael R. Hansen

DTU Informatics
Department of Informatics and Mathematical Modelling
Plan for today:

- A motivating example
  wireless sensor networks
- Brief introduction to Duration Calculus
- Overview of fundamental (un)decidability results
- A basic decidability results
  – with non-elementary complexity
- Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Plan for today:

- A motivating example wireless sensor networks
- Brief introduction to Duration Calculus
  - Overview of fundamental (un)decidability results
  - A basic decidability results
    - with non-elementary complexity
  - Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Plan for today:

- A motivating example wireless sensor networks
- Brief introduction to Duration Calculus
- Overview of fundamental (un)decidability results
  - A basic decidability results
    - with non-elementary complexity
  - Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Plan for today:

- A motivating example: wireless sensor networks
- Brief introduction to Duration Calculus
- Overview of fundamental (un)decidability results
- A basic decidability results
  - with non-elementary complexity
- Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Plan for today:

- A motivating example: wireless sensor networks
- Brief introduction to Duration Calculus
- Overview of fundamental (un)decidability results
- A basic decidability results
  - with non-elementary complexity
- Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Plan for today:

- A motivating example wireless sensor networks
- Brief introduction to Duration Calculus
- Overview of fundamental (un)decidability results
- A basic decidability results
  - with non-elementary complexity
- Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Plan for today:

- A motivating example
  wireless sensor networks
- Brief introduction to Duration Calculus
- Overview of fundamental (un)decidability results
- A basic decidability results
  – with non-elementary complexity
- Towards efficient model checking for Duration Calculus based on approximations
- A decision procedure for Presburger Arithmetic

At 1 pm: IMM summer party.
Nodes with solar panels

A node of a wireless sensor network has a solar panel:

[Diagram showing stored energy over time]
Energy consumption depends on usage

A node has a platform consisting of several components:
WSN-Model using parallel automata

A wireless sensor network can be modelled by parallel automata:

\[
\begin{align*}
\text{WSN} & = \|_{i=1}^{n} (\text{Node}_i \parallel \text{Environment}_i) \\
\text{Node}_i & = \text{SolarPanel}_i \parallel \text{Application}_i \\
\text{Environment}_i & = \text{Sun}_i \\
\text{Application}_i & = \|_{j=1}^{m_i} \text{Program}_j \parallel \text{Platform}_i \\
\text{Platform}_i & = \text{Processor}_i \parallel \text{Sensor}_i \parallel \text{Memory}_i \parallel \text{Radio}_i \\
\vdots & \\
\end{align*}
\]
Requirements can be modelled by Duration Calculus

There should be sufficient energy during the lifetime:

$$\square_p ( \ell \leq K \Rightarrow E_0 + \sum_i c_i \int \text{sun}_i - \sum_j k_j \int \text{program}_j > 0 )$$

- Succinct formulation
- Tool support
WSN-Requirements expressed in Duration Calculus

Requirements can be modelled by Duration Calculus

There should be sufficient energy during the lifetime:

\[ \square_p ( \ell \leq K \Rightarrow E_0 + \sum_i c_i \int_{\text{sun}_i} - \sum_j k_j \int_{\text{program}_j} > 0 ) \]

- Succinct formulation
- Tool support
WSN-Requirements expressed in Duration Calculus

Requirements can be modelled by Duration Calculus

There should be sufficient energy during the lifetime:

$$\square_p ( \ell \leq K \Rightarrow E_0 + \sum_i c_i \int \text{sun}_i - \sum_j k_j \int \text{program}_j > 0 )$$

- Succinct formulation 😊
- Tool support
WSN-Requirements expressed in Duration Calculus

Requirements can be modelled by Duration Calculus

There should be sufficient energy during the lifetime:

\[ \square_p ( \ell \leq K \Rightarrow E_0 + \sum_i c_i \int \text{sun}_i - \sum_j k_j \int \text{program}_j > 0 ) \]

• Succinct formulation 😊
• Tool support 😞
Background

- Provable Correct Systems (ProCoS, ESPRIT BRA 3104)
  Bjørner Langmaack Hoare Olderog

- Project case study: Gas Burner
  Sørensen Ravn Rischel

- Intervals properties
  Timed Automata, Real-time Logic, Metric Temporal Logic, Explicit Clock Temporal, ... Alur, Dill, Jahanian, Mok, Koymans, Harel, Lichtenstein, Pnueli, ...

- Duration of states
  Duration Calculus
  — an Interval Temporal Logic
  Zhou Hoare Ravn 91
  Halpern Moszkowski Manna

- Logical Calculi, Applications, Mechanical Support

- Duration Calculus: A formal approach to real-time systems
  Zhou Chaochen and Michael R. Hansen
  Springer 2004

Current focus: Tool support with applications
Background

- Provable Correct Systems (ProCoS, ESPRIT BRA 3104)  
  Bjørner Langmaack Hoare Olderog
- Project case study: Gas Burner  
  Sørensen Ravn Rischel

- Intervals properties  
- Duration of states  
  Duration Calculus  
  — an Interval Temporal Logic  
  Zhou Hoare Ravn 91  
  Halpern Moszkowski Manna

- Logical Calculi, Applications, Mechanical Support
- Duration Calculus: A formal approach to real-time systems  
  Zhou Chaochen and Michael R. Hansen  
  Springer 2004

  Current focus: Tool support with applications
Background

- **Provable Correct Systems (ProCoS, ESPRIT BRA 3104)**
  Bjørner Langmaack Hoare Olderog

- **Project case study: Gas Burner**
  Sørensen Ravn Rischel
  - **Intervals properties**
  - **Duration of states**
  Duration Calculus — an Interval Temporal Logic
  Zhou Hoare Ravn 91
  Halpern Moszkowski Manna

- **Logical Calculi, Applications, Mechanical Support**

- **Duration Calculus: A formal approach to real-time systems**
  Zhou Chaochen and Michael R. Hansen
  Springer 2004

Current focus: Tool support with applications
Background

• Provable Correct Systems (ProCoS, ESPRIT BRA 3104)  
  Bjørner Langmaack Hoare Olderog

• Project case study: Gas Burner  
  Sørensen Ravn Rischel

  • Intervals properties  

  • Duration of states  
    Duration Calculus  
    — an Interval Temporal Logic  
    Zhou Hoare Ravn 91  
    Halpern Moszkowski Manna

• Logical Calculi, Applications, Mechanical Support

  • Duration Calculus: A formal approach to real-time systems  
    Zhou Chaochen and Michael R. Hansen  
    Springer 2004

  Current focus: Tool support with applications
Background

- Provable Correct Systems (ProCoS, ESPRIT BRA 3104)  
  Bjørner Langmaack Hoare Olderog

- Project case study: Gas Burner  
  Sørensen Ravn Rischel

  - Intervals properties

  - Duration of states
    Duration Calculus  
    — an Interval Temporal Logic
    Zhou Hoare Ravn 91  
    Halpern Moszkowski Manna

- Logical Calculi, Applications, Mechanical Support

- Duration Calculus: A formal approach to real-time systems
  Zhou Chaochen and Michael R. Hansen
  Springer 2004

Current focus: Tool support with applications
Background

- Provable Correct Systems (ProCoS, ESPRIT BRA 3104)  
  Bjørner Langmaack Hoare Olderog
- Project case study: Gas Burner  
  Sørensen Ravn Rischel
  - Intervals properties
  - Duration of states
    Duration Calculus  
    — an Interval Temporal Logic  
    Zhou Hoare Ravn 91  
    Halpern Moszkowski Manna
- Logical Calculi, Applications, Mechanical Support
- Duration Calculus: A formal approach to real-time systems  
  Zhou Chaochen and Michael R. Hansen  
  Springer 2004

  Current focus: Tool support with applications
A ProCoS Case Study: Gas Burner System

State variables modelling Gas and Flame:

\[ G, F : \text{Time} \rightarrow \{0, 1\} \]

State expression modelling that gas is Leaking

\[ L \equiv G \land \neg F \]

Requirement

- Gas must at most be leaking 1/20 of the elapsed time

\[ (e - b) \geq 60 \text{s} \Rightarrow 20 \int_{b}^{e} L(t) dt \leq (e - b) \]
A ProCoS Case Study: Gas Burner System

State variables modelling Gas and Flame:

\[ G, F : \mathbb{Time} \rightarrow \{0, 1\} \]

State expression modelling that gas is Leaking

\[ L \equiv G \land \neg F \]

Requirement

- Gas must at most be leaking 1/20 of the elapsed time

\[ (e - b) \geq 60 \text{ s} \Rightarrow 20 \int_b^e L(t) dt \leq (e - b) \]
A ProCoS Case Study: Gas Burner System

State variables modelling Gas and Flame:

\[ G, F : \mathbb{Time} \rightarrow \{0, 1\} \]

State expression modelling that gas is Leaking

\[ L \equiv G \land \neg F \]

Requirement

- Gas must at most be leaking 1/20 of the elapsed time

\[(e - b) \geq 60 \text{ s} \Rightarrow 20 \int_{b}^{e} L(t) dt \leq (e - b)\]
Gas Burner example: Design decisions

-Leaks are detectable and stoppable within 1s:
  \[ \forall c, d : b \leq c < d \leq e. (L[c, d] \Rightarrow (d - c) \leq 1\text{s}) \]

  where
  \[ P[c, d] \equiv \int_c^d P(t) = (d - c) > 0 \]

  which reads “\( P \) holds throughout \([c, d]\)”

-At least 30s between leaks:
  \[ \forall c, d, r, s : b \leq c < r < s < d \leq e. \]
  \[ (L[c, r] \land \neg L[r, s] \land L[s, d]) \Rightarrow (s - r) \geq 30\text{s} \]

Proof obligation: Conjunction of design decisions implies the requirements.
Gas Burner example: Design decisions

-Leaks are detectable and stoppable within 1s:

\[ \forall c, d : b \leq c < d \leq e. \left( L[c, d] \Rightarrow (d - c) \leq 1 \text{s} \right) \]

where

\[ P[c, d] \equiv \int_c^d P(t) = (d - c) > 0 \]

which reads “\( P \) holds throughout \([c, d]\)”

-At least 30s between leaks:

\[ \forall c, d, r, s : b \leq c < r < s < d \leq e. \left( L[c, r] \land \neg L[r, s] \land L[s, d] \right) \Rightarrow (s - r) \geq 30 \text{s} \]

Proof obligation: Conjunction of design decisions implies the requirements.
Gas Burner example: Design decisions

- Leaks are detectable and stoppable within 1s:

\[ \forall c, d : b \leq c < d \leq e. (L[c, d] \Rightarrow (d - c) \leq 1 \text{s}) \]

where

\[ P[c, d] \equiv \int_c^d P(t) = (d - c) > 0 \]

which reads “P holds throughout \([c, d]\)”

- At least 30s between leaks:

\[ \forall c, d, r, s : b \leq c < r < s < d \leq e. \\
(L[c, r] \land \neg L[r, s] \land L[s, d]) \Rightarrow (s - r) \geq 30 \text{s} \]

Proof obligation: Conjunction of design decisions implies the requirements.
Terms: $\theta ::= x \mid v \mid \theta_1 + \theta_n \mid \ldots$

Temporal Variable

$v : \mathbb{Intv} \to \mathbb{R}$

Formulas: $\phi ::= \theta_1 = \theta_n \mid \neg \phi \mid \phi \lor \psi \mid \phi \sim \psi \mid (\exists x)\phi \mid \ldots$

chop

$\phi : \mathbb{Intv} \to \{\text{tt, ff}\}$

Chop:

for some $m : b \leq m \leq e$

In DC: $\mathbb{Intv} = \{ [a, b] \mid a, b \in \mathbb{R} \land a \leq b \}$
Interval Logic - Halpern Moszkowski Manna 83

Terms: \( \theta ::= x | v | \theta_1 + \theta_n | \ldots \)  

Temporal Variable  
\( v : \text{Intv} \rightarrow \mathbb{R} \)

Formulas: \( \phi ::= \theta_1 = \theta_n | \neg \phi | \phi \lor \psi | \phi \land \psi | (\exists x)\phi | \ldots \)  
chop  
\( \phi : \text{Intv} \rightarrow \{\text{tt,ff}\} \)

Chop:  
\[ \phi \land \psi \]

for some \( m : b \leq m \leq e \)

In DC: \( \text{Intv} = \{ [a, b] | a, b \in \mathbb{R} \land a \leq b \} \)
Interval Logic - Halpern Moszkowski Manna 83

Terms: \( \theta ::= x | v | \theta_1 + \theta_n | \ldots \)  

Temporal Variable

\( v : \text{Intv} \rightarrow \mathbb{R} \)

Formulas: \( \phi ::= \theta_1 = \theta_n | \neg \phi | \phi \lor \psi | \phi \sim \psi | (\exists x) \phi | \ldots \)  

chop

\( \phi : \text{Intv} \rightarrow \{tt, ff\} \)

Chop:

\[ \begin{align*}
\phi \sim \psi \\
\phi \\
\psi
\end{align*} \]

for some \( m : b \leq m \leq e \)

In DC: \( \text{Intv} = \{ [a, b] | a, b \in \mathbb{R} \land a \leq b \} \)
Extends Interval Temporal Logic as follows:

- **State variables** \( P : \text{Time} \rightarrow \{0, 1\} \)  
  - Finite Variability

- **State expressions** \( S ::= 0 \; | \; 1 \; | \; P \; | \; \neg S \; | \; S_1 \lor S_2 \)  
  \[ S : \text{Time} \rightarrow \{0, 1\} \]  
  - pointwise defined

- **Durations** \( \int S : \text{Intv} \rightarrow \mathbb{R} \) defined on \([b, e]\) by  
  \[ \int_b^e S(t)dt \]

- Temporal variables with a structure
Duration Calculus - Zhou Hoare Ravn 91

Extends Interval Temporal Logic as follows:

- **State variables** \( P : \mathbb{T}ime \to \{0, 1\} \)  \hspace{1cm} Finite Variability

- **State expressions** \( S ::= 0 \mid 1 \mid P \mid \neg S \mid S_1 \lor S_2 \)

\[
S : \mathbb{T}ime \to \{0, 1\} \quad \text{pointwise defined}
\]

- **Durations** \( \int S : \mathbb{I}ntv \to \mathbb{R} \) defined on \([b, e]\) by

\[
\int_b^e S(t)dt
\]

- Temporal variables with a structure
Duration Calculus - Zhou Hoare Ravn 91

Extends Interval Temporal Logic as follows:

- **State variables** \( P : \mathbb{T} \text{ime} \rightarrow \{0, 1\} \)  
  Finite Variability

- **State expressions** \( S ::= 0 \mid 1 \mid P \mid \neg S \mid S_1 \lor S_2 \)
  \( S : \mathbb{T} \text{ime} \rightarrow \{0, 1\} \) pointwise defined

- **Durations** \( \int S : \mathbb{I} \text{ntv} \rightarrow \mathbb{R} \) defined on \([b, e]\) by
  \[
  \int_b^e S(t) \, dt
  \]
  Temporal variables with a structure
Example: Gas Burner

Requirement

\[ \ell \geq 60 \Rightarrow 20\int L \leq \ell \]

Design decisions

\[ D_1 \equiv \Box([L] \Rightarrow \ell \leq 1) \]
\[ D_2 \equiv \Box(([L] \cap [\neg L] \cap [L]) \Rightarrow \ell \geq 30) \]

where \( \ell \) denotes the length of the interval, and

\[ \Diamond \phi \equiv \text{true} \land \phi \land \text{true} \]
\[ \Box \phi \equiv \neg \Diamond \neg \phi \]
\[ [P] \equiv \int P = \ell \land \ell > 0 \]

“for some sub-interval: \( \phi \)”

“for all sub-intervals: \( \phi \)”

“\( P \) holds throughout a non-point interval”

succinct formulation — no interval endpoints
Example: Gas Burner

Requirement

\[ \ell \geq 60 \Rightarrow 20\int \leq \ell \]

Design decisions

\[ D_1 \equiv \square([L] \Rightarrow \ell \leq 1) \]
\[ D_2 \equiv \square(([L] \cap [\neg L] \cap [L]) \Rightarrow \ell \geq 30) \]

where \( \ell \) denotes the length of the interval, and

\[ \Diamond \phi \equiv \text{true} \quad \text{\textasciitilde} \phi \quad \text{\textasciitilde} \text{true} \quad \text{“for some sub-interval: } \phi \text{”} \]
\[ \Box \phi \equiv \neg \Diamond \neg \phi \quad \text{“for all sub-intervals: } \phi \text{”} \]

\[ [P] \equiv \int P = \ell \land \ell > 0 \quad \text{“}P \text{ holds throughout a non-point interval”} \]

succinct formulation — no interval endpoints
Example: Gas Burner

Requirement

\[ \ell \geq 60 \implies 20 \int L \leq \ell \]

Design decisions

\[ D_1 \equiv \Box([L] \implies \ell \leq 1) \]
\[ D_2 \equiv \Box((\neg[L] \wedge \neg[L] \wedge L) \implies \ell \geq 30) \]

where \( \ell \) denotes the length of the interval, and

\[ \Diamond \phi \equiv \text{true } \neg \phi \neg \text{true} \quad \text{“for some sub-interval: } \phi \text{”} \]
\[ \Box \phi \equiv \neg \Diamond \neg \phi \quad \text{“for all sub-intervals: } \phi \text{”} \]
\[ [P] \equiv \int P = \ell \land \ell > 0 \quad \text{“} P \text{ holds throughout a non-point interval”} \]

succinct formulation — no interval endpoints