Parametric Real-Time System Feasibility Analysis Using Parametric Timed Automata

PhD Dissertation
Yusi Ramadian

Advisor: Luigi Palopoli
Co-advisor: Alessandro Cimatti
Real-Time System Applications

• A computer based system, which produces results to inputs complying with some temporal constraints
Main Research Idea

Real Time System Design Process

Parameters

Feasibility Analysis on Parametric Real Time System

Parameter 1
Region of Feasibility

Correct and Robust Real Time System
Contribution of PhD research

- Parametric Timed Automata (PTA) definition and representation of Real Time System
- Parametric Verification of Temporal Properties (PVTP) method
- Implementation in tool Quinq
- Application in case problems:
  - periodic task system [RTSSo8],
  - heterogeneous system [ETFA10],
  - collaboration with Modular Performance Analysis Toolbox (MPA) [CASES11].
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
The Importance of CORRECT & ROBUST Real Time System

- Safety consideration
- Manufacturing consideration
- Variability in environment and run-time

→ Need for formalization of design process
**Design Process**

- **Design & Modelling**
  - Activation pattern
  - Timing properties
  - Scheduling Algorithm

- **Robustness & Parameter Tuning**
  - Assign values
  - Evaluate system robustness w.r.t to parameters
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
## Sensitivity Analysis: Example Scenario

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Deadline</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Computation Time</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>Offset</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

![Task Diagram]

---

Page 9
Sensitivity Analysis:
Example Scenario Discussion #1

• Classical scheduling theory $\Rightarrow$ system failure
  Task system is not schedulable

• Solution
  - Stronger machine (Hardware solution)
  - Tweaking offset..
Sensitivity Analysis: Region of Schedulability on Offsets

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Deadline</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Computation Time</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>Offset</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
Sensitivity Analysis: Corrected Scenario

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Deadline</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Computation Time</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>Offset</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

The diagram shows the timeline for Task 1 and Task 2 with offset times indicated.
Sensitivity Analysis: Example Scenario Discussion #2: Robustness

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Period</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Deadline</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Computation Time</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>Offset</td>
<td>5</td>
<td>?</td>
</tr>
</tbody>
</table>

System robustness = ?
Requirement conclusion #1:

We want to find out:
- the schedulability regions in the space of parameters

→ most robust design for our real-time systems
Sensitivity Analysis: Example Scenario Discussion #3:

- System model not in classical RTS
- Examples:
  - System with buffers
  - Complex activation pattern
  - Heterogeneous, distributed system
  - Flexible deadline (e.g. Firm Deadline)
Sensitivity Analysis

• Requirement conclusion #2

Sensitivity analysis for general real-time system
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in QuinQ
  – Architecture
  – Demo

• State of the art
• Conclusion
Sensitivity Analysis: Formal problem definition

- Problem 1:

System S

Task 1  Task 2  ...  Task n

P : {P1, P2, ... Pn}

Scheduling Algorithm
Sensitivity Analysis:
Formal problem definition

- Problem 2:

\[ \text{System } S \]

\[ \text{Task } 1 \quad \text{Task } 2 \quad \ldots \quad \text{Task } n \]

\[ P : \{P_1, P_2, \ldots, P_n\} \]
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
Parametric Timed Automata

- Timed automata with parameters extension
- Main differences:
  - Parameters
  - Auxiliary variables
  - General update statement
Real Time System in PTA: Activation Pattern

- Allow complex activation pattern

 Activation depending on time
  Release $\tau_1$
  Clock = $Q_1$
  Clock := Clock – $Q_2$

 Activation depending on environment
  Release $\tau_3$
  Event$_e$ = True
  Clock := 0

 Activation depending on other tasks
  Release $\tau_2$
  Clock = $Q_2$
  Clock := 0

 Avoiding restrictive offset $0$
  Wait for offset
  Clock $\leq Q_1$
  Clock := 0

 Wait for $Q_1$
  Clock $\leq Q_1$
  Clock := 0

 Wait for $Q_2$
  Clock $\leq Q_2$
  Clock := 0

 Release $\tau_2$
  Clock = $Q_2$
  Clock := 0
Real Time System in PTA: Feasibility Checker

Parameter

Idle

Busy

Check

Error

Release \( \tau_1 \)

Clock_\( x \):=0

\( r:= C_1 \)

Clock_\( x \):=0

\( r:= C_1 \)

Clock_\( x \):=0

\( r:= r + C_1 \)-Clock_\( x \)

Clock_\( x \):=D_1 \land Clock_\( x \) \leq r

Auxiliary Variables
Sensitivity Analysis via PTA

- PTA:
  - Activation Automata
  - Activation Automata
  - Activation Automata

- Feasibility region R:
Symbolic representation of PTA

- Current state variables $V$:
  Discrete vars $D$: Location, transitions as boolean
  Continuous vars $X$: Clocks and other variables as real

- Symbolic model of PTA: set of constraints on boolean and real variables

- Examples:
  $Loc_i \rightarrow x - y \leq O_1$
  $Trans_i \rightarrow (x \geq C_1) \land (x' = C_1 + x) \land (y' = y)$
Bounded Model Checking (BMC)

• Look only for counterexample made of $k$-states

• BMC($k$):

$$I(V^0) \land R(V^0, V^1) \land \ldots \land R(V^{k-1}, V^k) \land \text{Error}(V^k)$$

• Completeness of the solution is not guaranteed

• Complementing method: inductive reasoning
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
PTVP algorithm intuition:  
Search for an error trace

- Verification on reachability problem using BMC:  
  An error trace for every found counterexample
- Alternatively, error trace can be searched via non-parametric model checker
- An error trace \( \pi : \)
PTVP algorithm intuition:
An error trace $\pi$

- Along with the trace, an assignment for the parameters that validate the trace is produced.
PTVP algorithm intuition: Sensitivity analysis to an error trace

- By processing the trace, the surrounding region of parameters that make the trace true is identified
- ...And we rule this region out from the next search
PTVP algorithm intuition: Schedulability region

- Feasibility region: found by iteratively bounding the parameter space from the unschedulability regions
Parametric Verification of Temporal Properties (PVTP) Algorithm

Require: PTA describing activations and scheduling of n tasks
Ensure: Schedulability Region

1: for \( i = 1 \) to n do
2: \hspace{1em} PTA.init(ParamSchedProblemForTask(i))
3: \hspace{2em} j = 0
4: \hspace{3em} while PTA.reachable(Error) do
5: \hspace{4em} trace = PTA.get_trace()
6: \hspace{5em} Unfeasible[j] = PTA.get_parameter(trace)
7: \hspace{6em} PTA.add_constraints( negate( Unfeasible[j]) )
8: \hspace{7em} j++
9: \hspace{8em} Feasible[i] = not(big_or(0, j, Unfeasible))
10: \hspace{9em} Return big_and(0, n, Feasible)
Sensitivity Analysis

- Given Polyhedron in the space of clocks and parameters $\text{Poly}\{P, X\}$
- Obtain $\text{Poly}\{P\} \iff \exists X, \text{Poly}\{P, X\}$

→ Existential Quantifier Elimination
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
Implementation in Quinq

• Based on NuSMV3 symbolic model checker with underlying MathSAT SMT solver

• Main functionalities:
  – Input handling
  – PVTP algorithm implementation
  – Completion check
  – Output handling

• Components
  – Sensitivity add-on
  – High level periodic system analysis
  – Search optimization
  – Model checker drivers
  – Graph generator

• Blackbox components: UPPAAL, JUNT, existelim
Input Handling

Periodic System Input → NuSMV Model

NuSMV Model → JUNT

JUNT → NuSMV Model

Automatic Periodic Analysis
• Point of Considerations:
  – Integer vs Real domain
  – Array data structure
  – Clocks
  – Transition synchronization
Architecture

RTS Model

QUINQ

Input Handling

PVTP Implementation

Output handling

NuSMV3

UPPAAL

Feasibility Region
Parametric Verification of Temporal Property Implementation

- NuSMV Model
  - Error Trace Search
  - Error Trace
  - Sensitivity Analysis
  - Feasibility Region
- PVTP Implementation
PTVP Algorithm Implementation

- NuSMV Model
  - Error Trace Search
    - Parametric search
    - Non-parametric search
  - Error Trace

- NuSMV3
- UPPAAL

Page 41
Non-Parametric Search

UPPAAL Model + Parameters

Non-parametric search

Search Point Generator

Trace Translation

Trace Replication

Error Trace

UPPAAL

NuSMV3
Sensitivity Analysis: Implementation

Model + Error Trace

Sensitivity Analysis

Trace Extraction

Polyhedral region in clocks and variables space

Constraint Processing

Polyhedral region in parameters space

After Processing

Parametric Constraints
Constraint Processing

- Constraints in Clocks & Vars
  - Reorder expression
    - Case Handling
      - Equality Constraint Propagation
      - Existelim
  - Parametric Constraints
Presentation Outline

- Motivation
  - Real time system design
  - Example scenario
  - Problem Statement

- Solution
  - Parametric Timed Automata
  - Parametric Verification of Temporal Property Method

- Implementation in Quinq
  - Architecture
  - Demo

- State of the art

- Conclusion
- Previously Illustrated example: 2 periodic tasks system with 3 parameters
- \( S = \{\text{task1, task2}\} \)
- Periodic tasks: \( T_1 = D_1 = 20 \), \( T_2 = D_2 = 30 \)
- Offset: \( O_1 = 0 \)
- Parameters:
  - Computation time: \( C_1, C_2 \)
  - Offset: \( O_2 \)
Demo: result

- Offset $2 = 8$
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
## Sensitivity Analysis Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Flexible RTS</th>
<th>Expression</th>
<th>System known apriori</th>
<th>Inference Point</th>
<th>Number of Parameters</th>
<th>Feasibility Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bini</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>No limit</td>
<td>✓</td>
</tr>
<tr>
<td>MAST</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SMART</td>
<td>✓</td>
<td>*</td>
<td>✓</td>
<td>-</td>
<td>No limit*</td>
<td>✓</td>
</tr>
<tr>
<td>IMITATOR</td>
<td>✓</td>
<td>*</td>
<td>-</td>
<td>✓</td>
<td>2</td>
<td>✓</td>
</tr>
<tr>
<td>QUINQ</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>No limit*</td>
<td>✓</td>
</tr>
</tbody>
</table>
Presentation Outline

• Motivation
  – Real time system design
  – Example scenario
  – Problem Statement

• Solution
  – Parametric Timed Automata
  – Parametric Verification of Temporal Property Method

• Implementation in Quinq
  – Architecture
  – Demo

• State of the art
• Conclusion
Conclusion

- PTA representation $\rightarrow$ flexible activation pattern, general RTS presentation
- PVTP algorithm $\rightarrow$ General method to obtain feasibility region
- Implemented in Quinq with applications on some example cases
- Edge on comparison with other tools:
  - Flexible RTS representation
  - No reference point input needed
  - Whole region of schedulability result
K-Induction

• Does there exist $k$ such that the following formula is unsatisfiable

$$\text{if unsatisfiable and } \text{BMC}(k) \text{ unsatisfiable then error state } \text{unreachable}$$