

Towards a Formal Definition of Timing Predictability

Daniel Grund

Saarland University, Saarbrücken, Germany

RePP Workshop, Oct 2009



1 Motivation

2 Predictability

- Disambiguation
- Definitions

3 Component Model

- Individual Components
- Composition

4 Summary & Questions

1 Motivation

2 Predictability

- Disambiguation
- Definitions

3 Component Model

- Individual Components
- Composition

4 Summary & Questions

- Complexity of embedded systems increases
 - They become more and more “unpredictable”
 - With current methods
 - ▶ computation of WCET bounds will become infeasible or
 - ▶ computed bounds will be useless
- ⇒ Need better analyses or more predictable systems

Also true for schedulability and response-time analysis?

- To design predictable systems one should know what predictability means, right!?
- People mean different things when saying “predictability”
- Criteria for predictability are mostly
 - ▶ intuitive
 - ▶ case-based
 - ▶ phenomenological, symptom-based

| Predictability: | better | worse |
|--------------------|----------|---------------|
| pipeline | in order | out of order |
| branch prediction | static | dynamic |
| cache organization | private | shared caches |
| replacement policy | LRU | PLRU |

- This is based on intuition and experience
- Is there an underlying principle?

- Disambiguate meanings and clarify notions
- Derive definitions/criteria that are
 - ▶ formal
 - ▶ uniformly applicable
 - ▶ reason-based

1 Motivation

2 Predictability

- Disambiguation
- Definitions

3 Component Model

- Individual Components
- Composition

4 Summary & Questions

What is Predictability?

Oxford Dictionary

predictable: adjective, able to be predicted

to predict: verb, state that (a specified event) will happen in the future

- Predictability is a property of a system
- Definition should not depend on specific analysis techniques

Proposition

The notion of predictability should capture **if, and how precise**, a property can be predicted by **any analysis**.

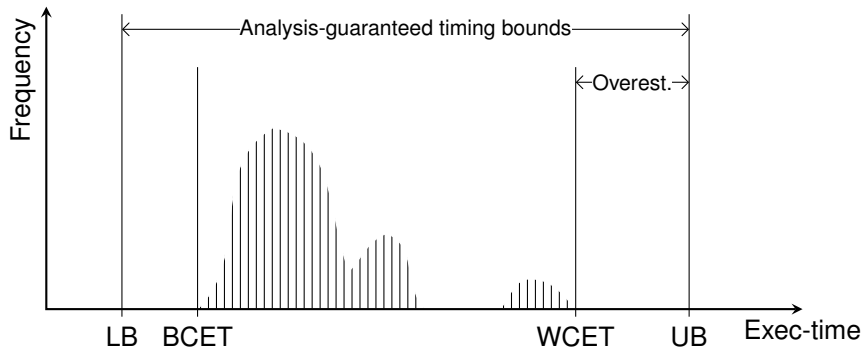
- Would like to consider analysis costs
- predictable: able to be predicted **with reasonable effort**

Alternative proposition

The notion of predictability should capture if, and how precise, a property can be predicted by any **efficient** analysis

- How to argue about all efficient analyses?
- What is efficient?
- Complicates matters even more

- In deterministic systems one can predict properties exactly
 - However, there is non-determinism
 - The amount of non-determinism determines the quality of a prediction
- ⇒ Determine and distinguish sources of non-determinism
- Next: Property “Time”



- BCET/WCET: Best- and worst-case execution time
- LB/UB: Lower and upper bound

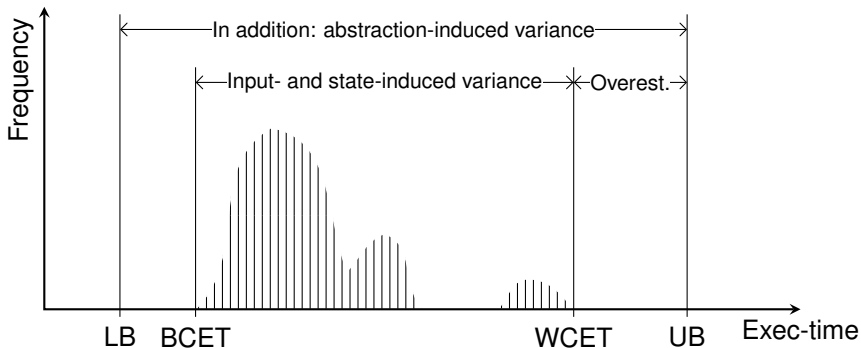
- No analysis can do better than [BCET, WCET]
- This interval limits the precision of predictions
- ⇒ Intrinsic (inherent) / **Predictability**

- An efficient analysis has to abstract and can only predict [LB, UB]
- ⇒ Extrinsic (ascribed to analysis) / **Predictability'**

Non-determinism in Execution-Times

Execution time of a program p on a given system depends on

- the system input
- the initial hardware state



- Uncertainty about state shows up at the “beginning”
- Implies set of possible initial hardware states
- Uncertainty about input shows up during execution
- Implies set of possible transitions

Proposition

| | |
|------------------------------|--|
| State-induced predictability | variance due to different initial states |
| Input-induced predictability | variance due to different inputs |

- Let Q denote the set of all **system states**
- System is input-deterministic
- Let I denote the set of all **inputs**
- e.g. streams of sensor values
- Let $T_p(q, i)$ be the **execution time** of program p , starting in hardware state $q \in Q$ with input $i \in I$.

Definition: Predictability

$$\Pr_p(Q, I) := \min_{q_1, q_2 \in Q} \min_{i_1, i_2 \in I} \frac{T_p(q_1, i_1)}{T_p(q_2, i_2)}$$

- Captures maximal variance in execution time
- Measure is within [0..1]
- 1 means perfectly predictable (= Berkeley-repeatable?)

Definition: State-induced Predictability

$$\text{SIPr}_p(Q, I) := \min_{q_1, q_2 \in Q} \min_{i \in I} \frac{T_p(q_1, i)}{T_p(q_2, i)}$$

Definition: Input-induced Predictability

$$\text{IIPr}_p(Q, I) := \min_{q \in Q} \min_{i_1, i_2 \in I} \frac{T_p(q, i_1)}{T_p(q, i_2)}$$

- Predictability comprises state- and input-induced predictability
- $\text{SIPr}_p \cdot \text{IIPr}_p \leq Pr_p$

- Definitions based on observable behavior (time)
- But we want to capture its reasons
- ⇒ What is the reason for the variance?

- Definitions depend on a program
- But predictability is system property
- ⇒ How to quantify over all programs?

1 Motivation

2 Predictability

- Disambiguation
- Definitions

3 Component Model

- Individual Components
- Composition

4 Summary & Questions

- System consists of connected components
- Each component has
 - ▶ component state
 - ▶ component input
- State of all components determines system state

What is the reason for the variance?

- Variance should appear in individual components
- Variance in components “somehow” adds up

Goal: Come up with model that is compositional wrt. predictability

- 1 Characterize variance/predictability of each component
- 2 Use compositionality to deduce system predictability

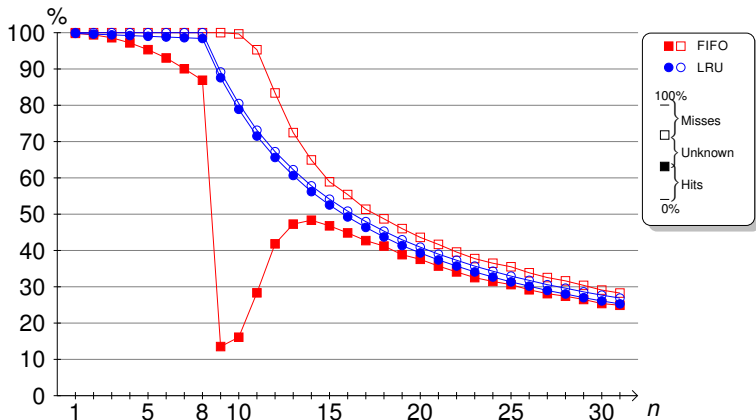
How to quantify over all programs?

- Program can be seen as part of system input
- Anyway, $\left\{ \begin{array}{l} \text{Program} \\ \text{System input} \\ \text{System state} \end{array} \right\}$ induce component input
- Over-approximate component input
- For now, consider all component inputs

- State: cached blocks and status of replacement algorithm
- Input: sequences of accesses
- Output: (*Hit*|*Miss*)*
- Characterize predictability

State-induced variance

for different classes of inputs



- Graph showed events
- Events can be associated with times
- Times **do not** simply add up
- E.g. penalties might mask each other out

- Which information do we need about components to achieve compositionality?
- That is, how to abstract?
- Certainly not by a graph like the previous one
- What sizes are appropriate to describe components?
- Input-output relation?

1 Motivation

2 Predictability

- Disambiguation
- Definitions

3 Component Model

- Individual Components
- Composition

4 Summary & Questions

- Predictability is variance due to **inherent** system properties
- Difficult to restrict to “efficient” analysis
- How do we achieve compositionality wrt. predictability?
- What information do we need about components to achieve this?
- Likely, I am biased to WCET determination:
can **you** concretize my terms into your domain?