Horizon

Performance Prediction of Expanded Parallel Discrete Event Simulations

Georg Kunz, Simon Tenbusch, James Gross, Klaus Wehrle

Why Performance Prediction of Parallel Simulations?

- **Design Questions of Parallel Simulations**
  - Model Structure
    - Identify and eliminate bottlenecks
  - Simulation Framework Performance
    - Compare different event scheduling strategies

- **Developing Efficient Parallel Simulations is Complex**
  - Additional design goal: Performance
  - Developers need insight into runtime behavior

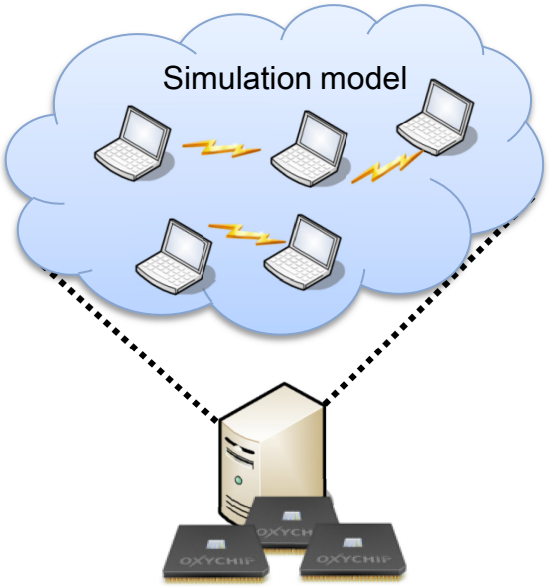
⇨ Performance Prediction and Analysis Tool

Background

Introducing Horizon

Horizon

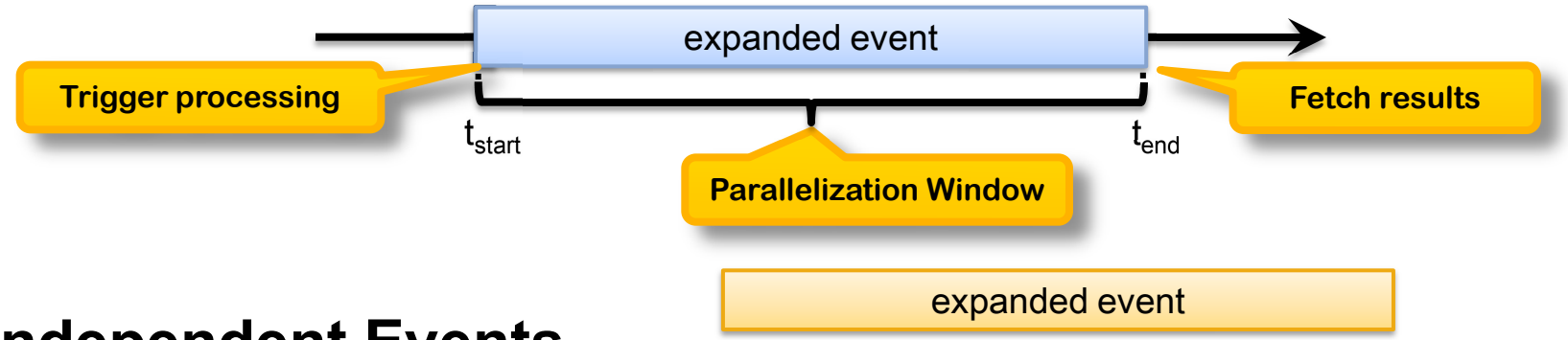
- **Horizon**
  - Focus on multi-processor systems
    - Desktop: 4-8 cores, servers: 24 cores
    - “Desktop Cluster”
    - ⇨ Cheap, powerful commodity hardware
  - Centralized architecture
  - Conservative synchronization
    - Determine independent events



- **Expanded Events**
  - Modeling paradigm
  - Per event lookahead
  - ⇨ Identify independent events

Expanded Events

- **Expanded Events**
  - Model processes that span period of time
  - Augment discrete events with durations
  - ⇨ Discrete events span period of **simulated** time



- **Independent Events**
  - Events starting between t_{start} and t_{end}
  - Do not depend on results generated by overlapping events
  - Modeling paradigm

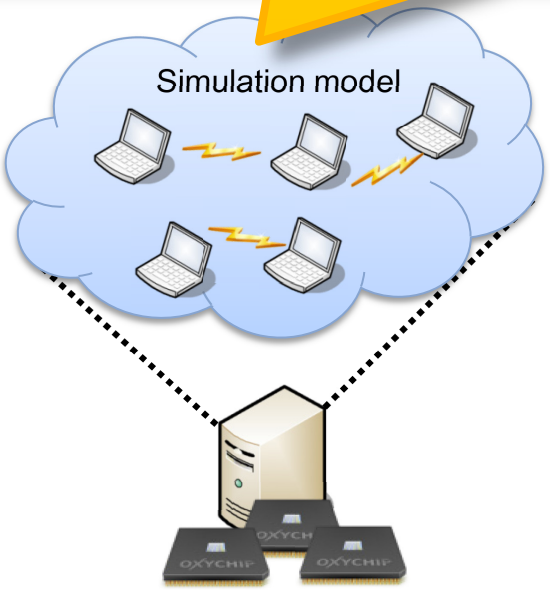
Approach

Performance Prediction Methodology

Performance Prediction Methodology

- **Goal**
  - **Minimal** simulation runtime
  - Compute **Optimal** Event Schedule
- **Constraints**
  - Available number of processing resources
  - Event inter-dependencies
  - ⇨ NP-complete Scheduling Problem

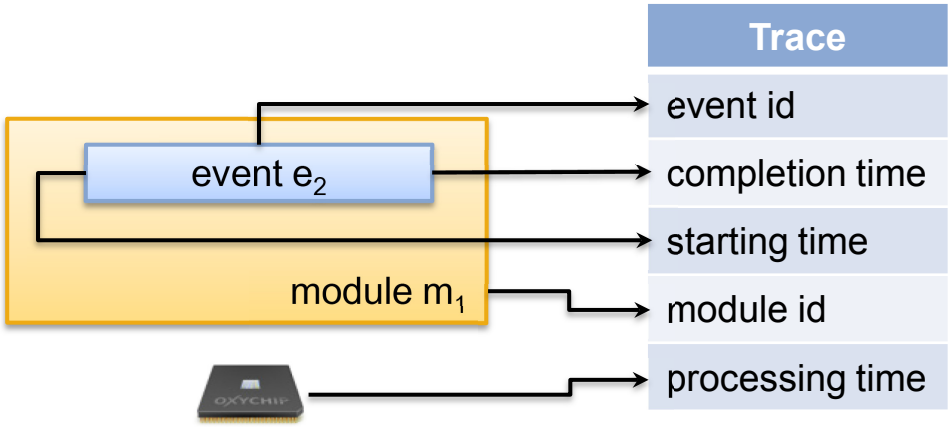
Design Questions
- Identify and eliminate bottlenecks
- Compare different event scheduling strategies



⇨ **Linear Optimization Problem**

Performance Prediction Methodology

1. Trace simulation



2. Solve linear program

Objective function:

$$\text{minimize } R$$

$\forall e \in E$:

$$\sum_{c \in C} x_{e,c} = 1 \quad (1)$$

$$y_e + p(e) \leq R \quad (2)$$

$\forall c \in C, \forall d, e \in E$ with $d < e$ and $d \parallel e$ and $m(d) \neq m(e)$:

$$y_e - y_d + (1 - z_{d,e}) \cdot M \geq (x_{d,c} + x_{e,c} - 1) \cdot p(d) \quad (3)$$

$$y_d - y_e + z_{d,e} \cdot M \geq (x_{d,c} + x_{e,c} - 1) \cdot p(e) \quad (4)$$

$\forall c \in C, \forall d, e \in E$ with $d < e$ and $d \parallel e$ and $m(d) = m(e)$:

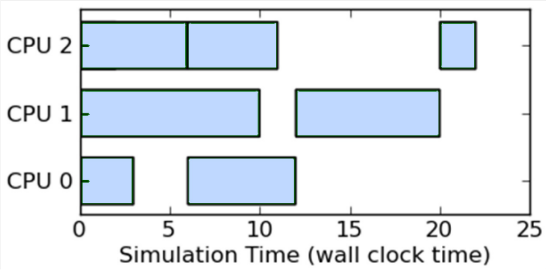
$$y_e - y_d + (1 - z_{d,e}) \cdot M \geq p(d) \quad (5)$$

$$y_d - y_e + z_{d,e} \cdot M \geq p(e) \quad (6)$$

$\forall d, e \in E$ with $f(d) < s(e)$:

$$y_d + p(d) \leq y_e \quad (7)$$

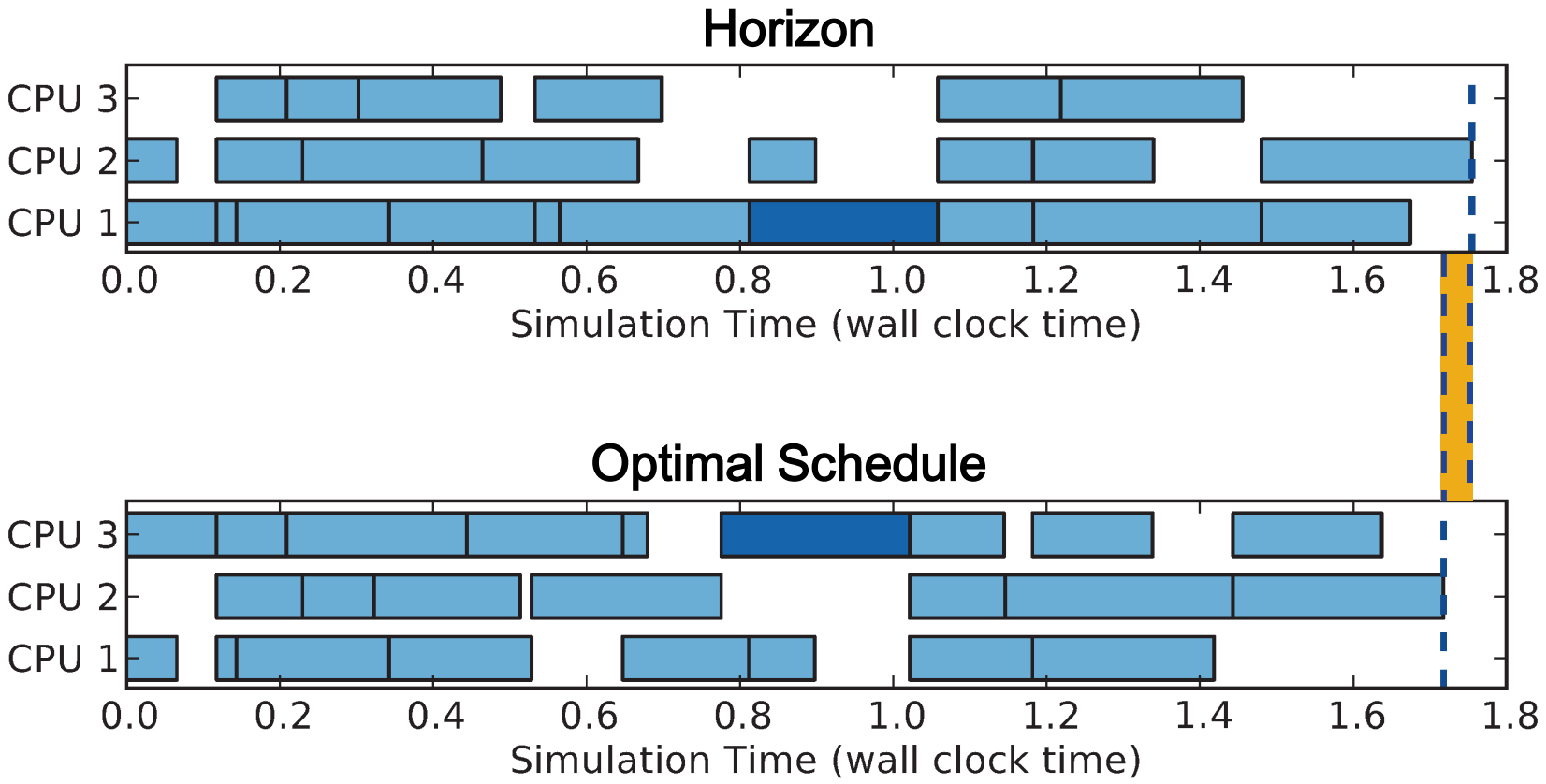
3. Analyze output



Output
CPU id
execution begin

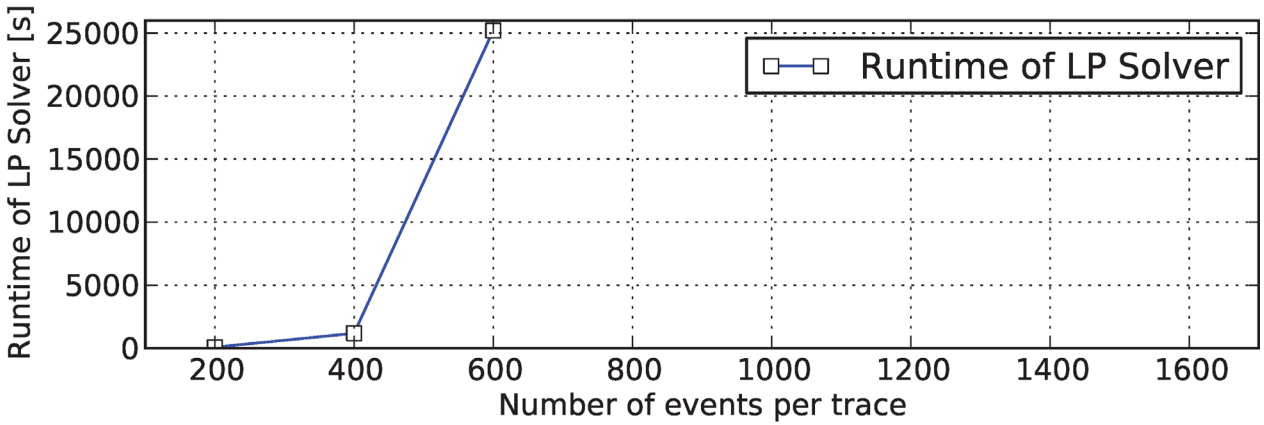
Performance Analysis

- **Analyze Output**
  - Identify and eliminate bottlenecks



Scalability Issues

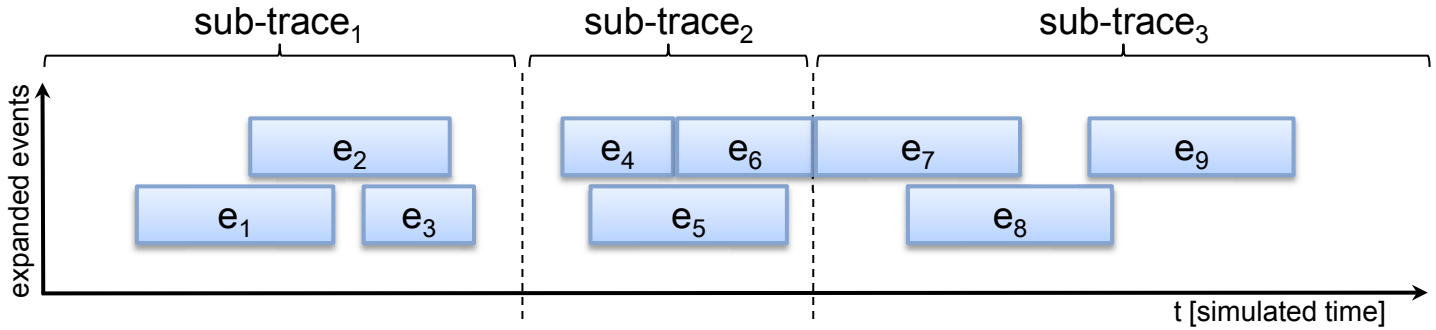
- **NP-complete Scheduling Problem**
  - Complexity limits length of trace



- **Mitigate Scalability Issue**
  - Relaxations
    - Trade accuracy for scalability
  - Trace Splitting
    - Reduces complexity **while** maintaining accuracy

Trace Splitting: Divide and Conquer

- **Observation**
  - Event sequence contains gaps
  - Act as implicit synchronization points



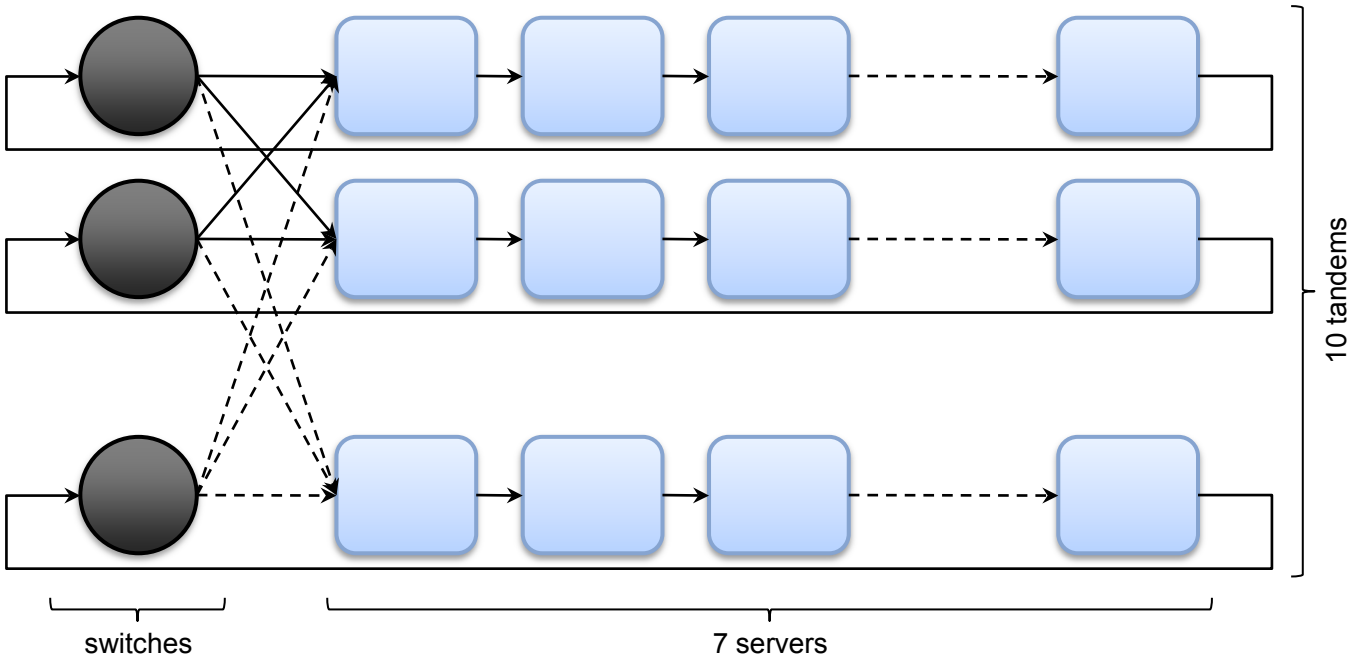
- **Solution**
  - Split trace at gaps
  - Solve scheduling problem for each sub-trace
  - Combine solutions of sub-traces
    - ⇨ Valid optimal schedule for full trace

Evaluation

How does it perform?

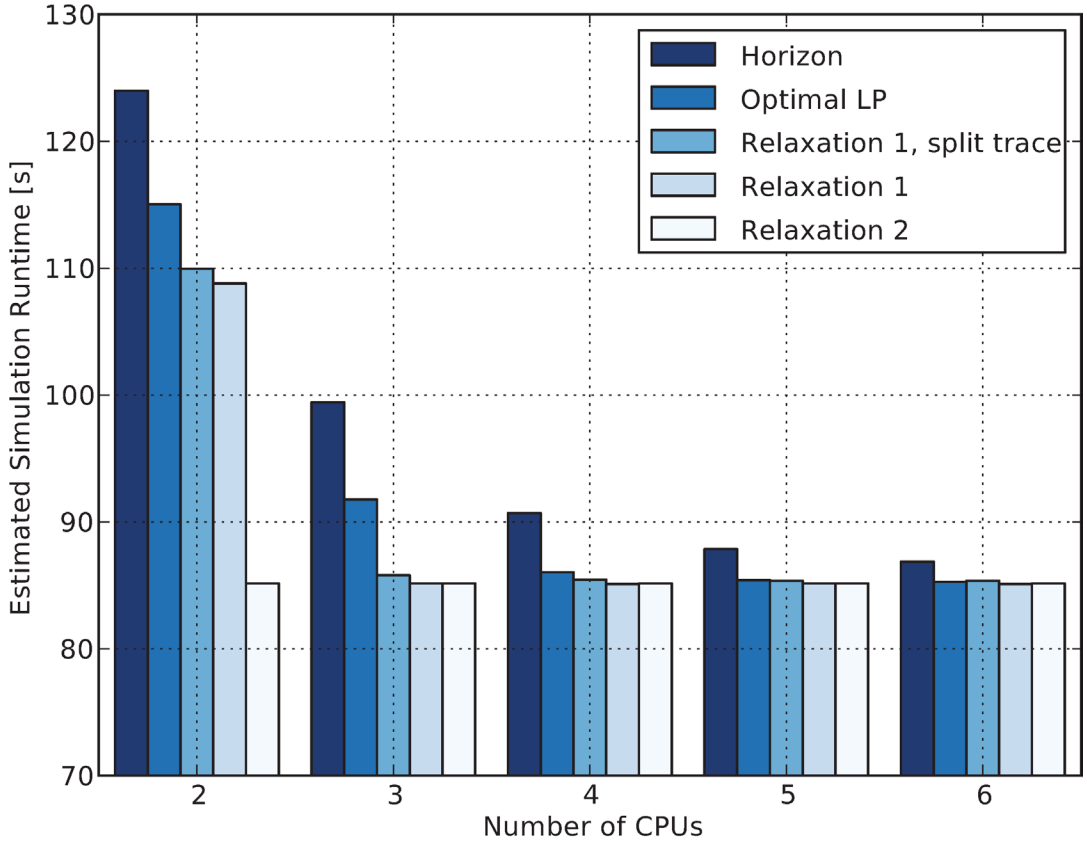
Evaluation Model

- **Closed Queueing Network**
  - Exponentially distributed service times
  - Static link delays
  - Uniformly distributed processing times



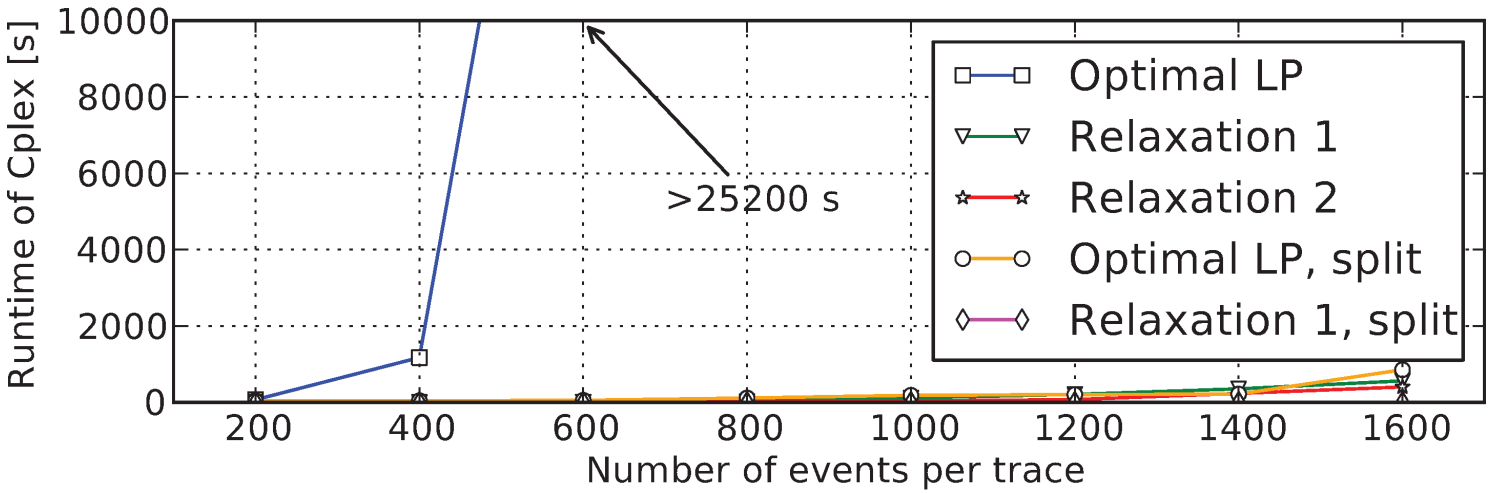
Prediction Accuracy

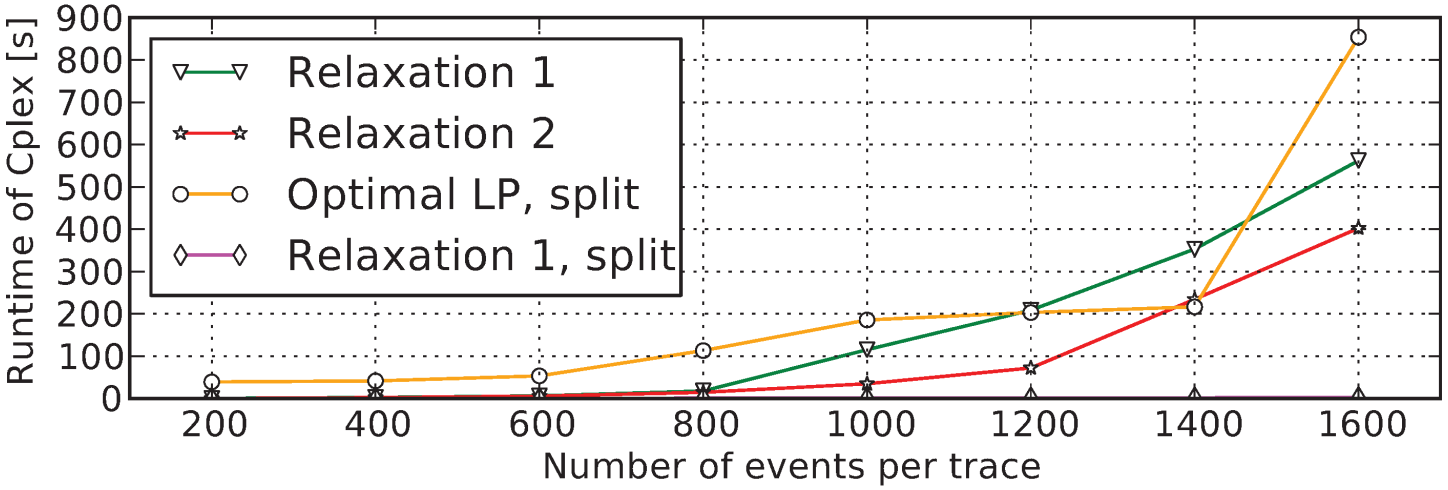
- **Comparison of Prediction Results**
  - LP implementation using Zimpl
  - LP solver CPLEX



Scalability

- **Runtime of LP Solver**





Conclusions

The take away message

Conclusion

- **Performance Prediction Methodology**
  - Provides insight into execution behavior
  - Supports development and optimization process
  - Compute optimal event schedule
- **Mitigate Scalability Issues**
  - Relaxations
  - Trace Splitting Scheme
- **Demo Announcement**
  - Horizon
  - Interference Coordination in LTE Networks (Donald Parruca)

Thank You

Questions?