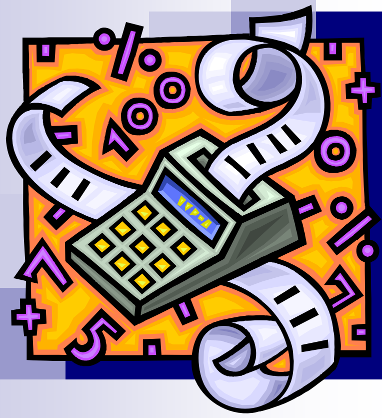
8th International Forum on Application-Specific Multi-Processor SoC (MPSoC '08)
23rd-27th June 2008, Aachen, Germany

Monday, 23rd June 2008



Analytical Models of Communication for MPSoCs

Paolo Ienne

Joint work with
Chrysostomos Nicopoulos,
Balaji Raman,
and Patrick Thiran

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Processor Architecture Laboratory
Laboratoire d'Architecture des Processeurs

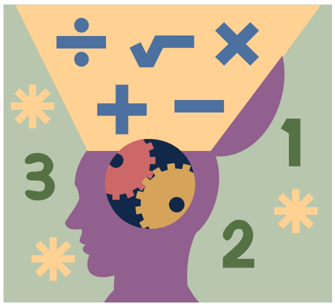
Real Men (and Women) Simulate!

- Computer Architecture is dominated by simulation
  - Analytical and statistical models fail in many cases to capture essential system behaviour of microarchitectures
- Embedded System Design is similarly simulation-centric
  - (Apparently?) strong guarantee of correctness

YET

- Simulation times are often **design bottlenecks**
- Simulation **rarely gives real insight and strong guarantees** on the dynamic behaviour of a system

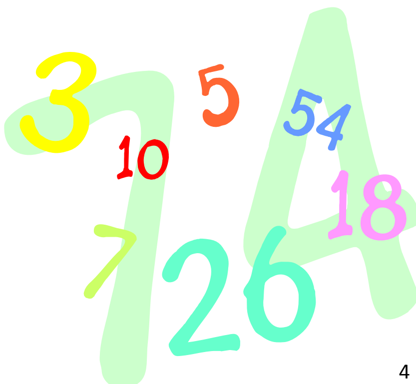
Why Not Look at Analytical Models?



- **Mathematical modeling** is an attractive alternative
  - **Fast** exploration and analysis
    - Closed form
  - **Deep insights** valuable for resource saving
    - Can prove or invalidate properties, not just verify
  - **Efficient** design exploration
    - Exploration encompassing **several variables simultaneously** is feasible and can isolate the effect of each parameter

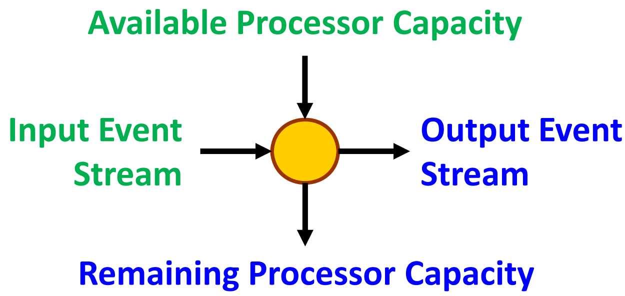
Many Analytical Models

- **Synchronous Data Flow graphs (SDF)**
  - Easily model concurrency through data flow representation
- **Stochastic Automata Networks (SAN)**
  - Efficient in modeling communicating processes
  - Based on Markovian models
- **Event Adaptation Functions (Sym TA/s)**
  - Global analysis in heterogeneous systems by coupling (interfacing) local event models
- **Real-Time Calculus (RTC)**
  - Main focus here



Real-Time Calculus

- Based on **Network Calculus**
- Mathematical model of networks based on **Min-Plus Algebra**:
  - Addition becomes computation of the Minimum
  - Multiplication becomes Addition
- Based on **deterministic queuing theory**
  - Not a statistical method; provides **worst-case bounds** (not averages)
- Does not restricts events to particular models (e.g., periodic)
- Models **flow through network elements**

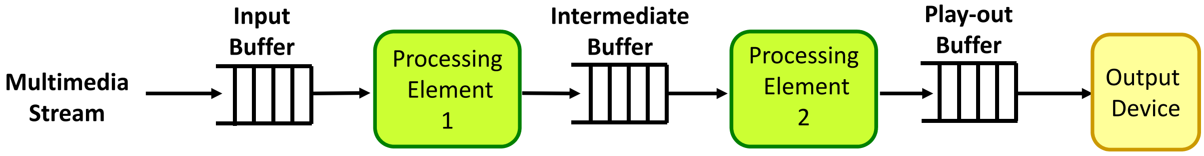


Seminal Work

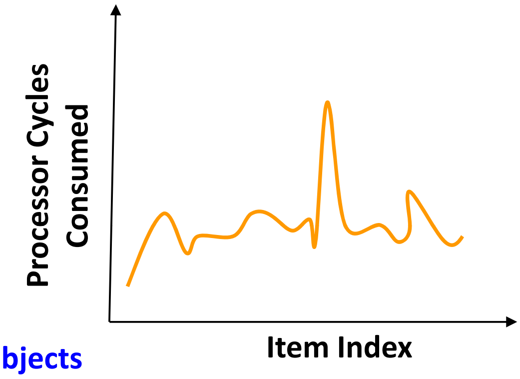
- **Cruz (UCSD), Le Boudec and Thiran (EPFL), C-S Chang (National Tsing Hua University, Taiwan)**
  - Analysis of deterministic flow systems in the Internet
  - Properties of integrated service networks, scheduling, buffer dimensioning, window flow control, etc.
- **Thiele (ETHZ), Chakraborty (NUS)**
  - *Hard real-time systems*
    - Scheduling analysis and interface-based design
  - *Network processor architectures*
    - Design space exploration
  - *Multimedia systems*
    - Buffer-sizing, processor-frequency selection, DVFS, etc.

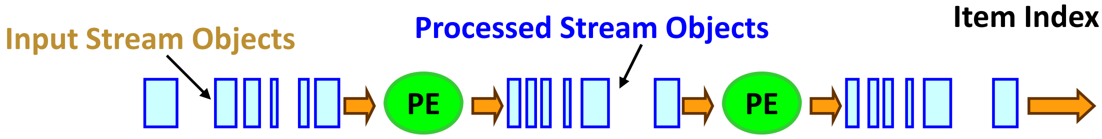
Example: MPEG-2 Decoder

- Real-Time Calculus to model a multimedia SoC
  - Input: Compressed video clip
  - Output: Decoded video clip at a pre-specified play-out rate



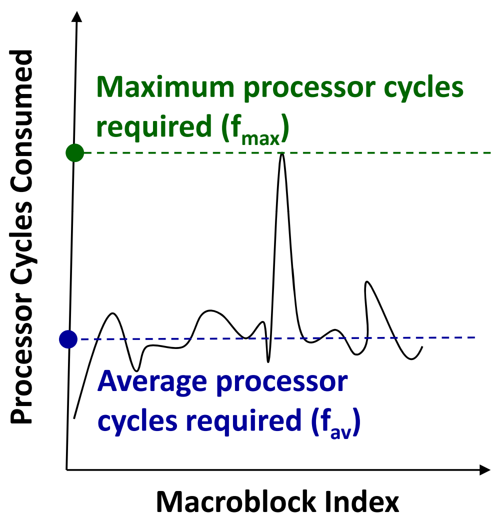
- Multimedia streams exhibit **data-dependent variability**
  - Variable arrival of items
  - Variable execution requirements
- Captured through **Variability Characterization Curves (VCC)**





Processor Frequency Requirements

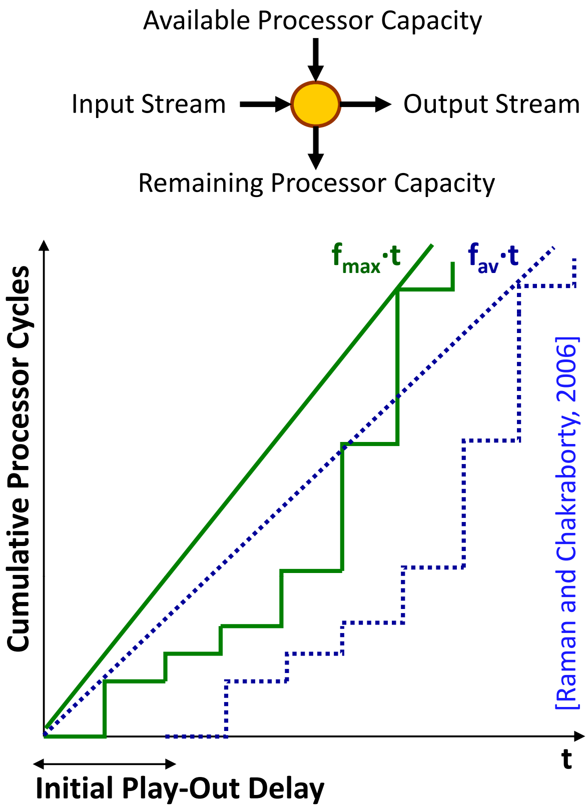
- **Data-dependent variability** is fundamentally intertwined with **two crucial parameters** in the video decoding system
- **Play-Out Rate** (e.g., 30 frames/s)
  - Requires a data item to be processed within certain time
  - Imposes real-time constraints on the application
- **Processor Frequency**
  - Minimum cycles/second required to meet play-out rate
- A naïve design would choose the worst-case frequency f_{max}
  - **Can we do better?**



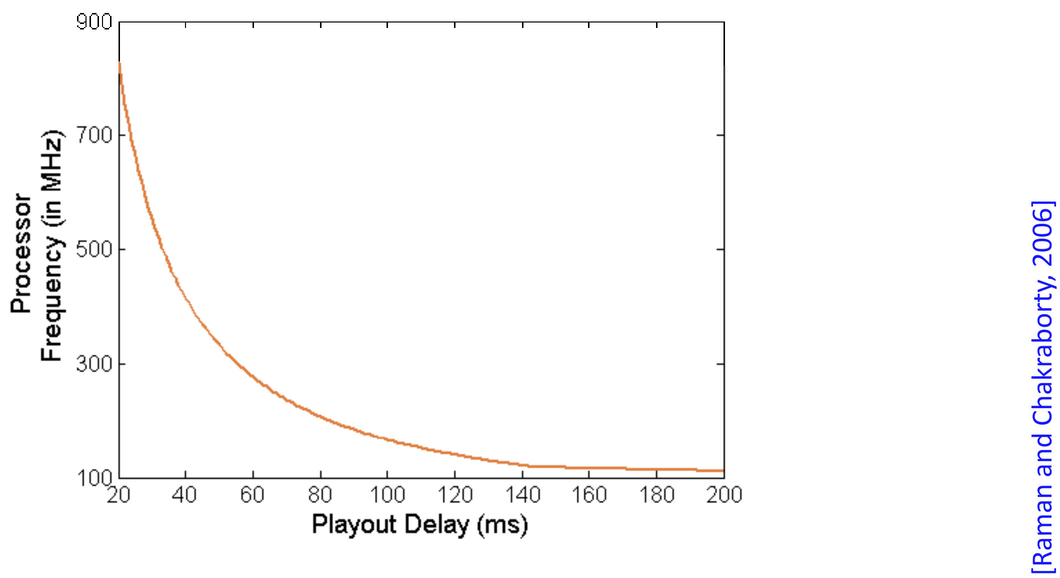
Introduce a Play-Out Delay

- **Play-Out Delay**
  - Delay after which the output device starts to display the video
  - Intuitively, a non-null play-out delay creates a small reservoir of time ready to be spent when critically needed
- If we choose a **near-zero play-out delay**
  - Processor frequency required to meet play-out rate is f_{max}
- If, however, we accept a **small initial play-out delay**
  - **Frequency requirement immediately drops to f_{av}**

> **Real-Time Calculus** enables **fast** and **accurate play-out delay estimation**, because it captures the inherent variability of the workload

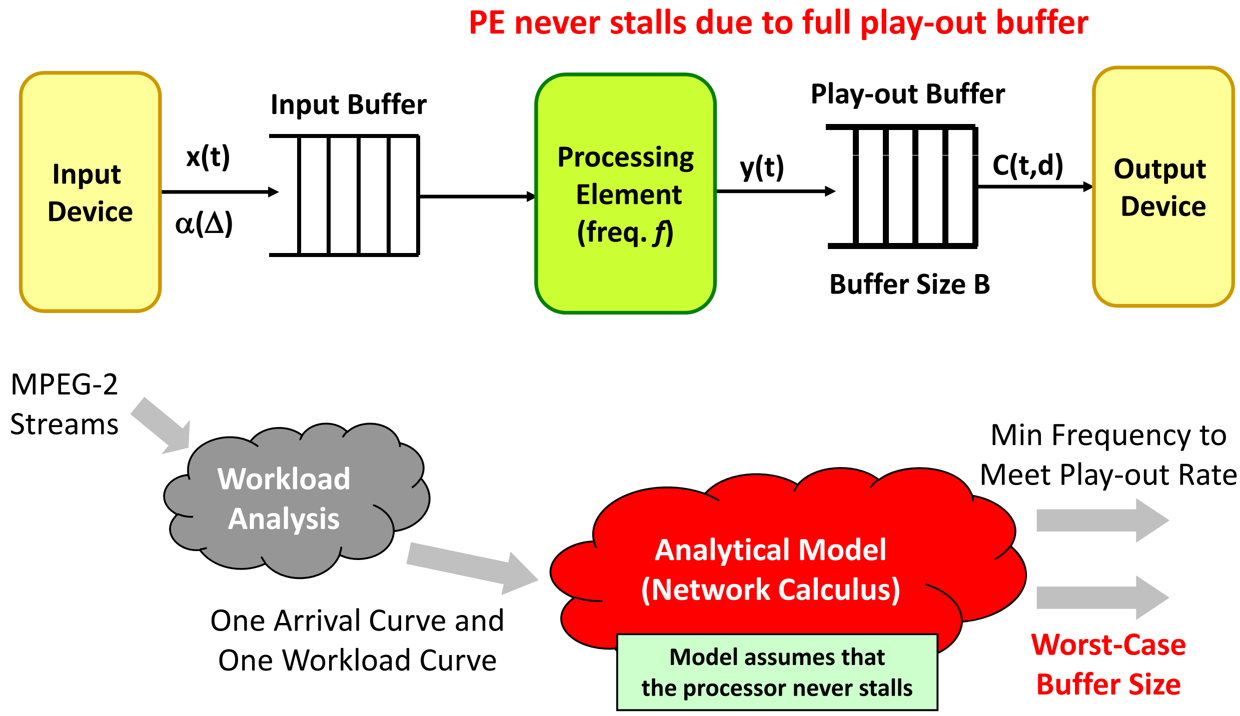


Play-Out Delay vs. Frequency



- Small **increases in play-out delay** lead to **significant reduction in the processor frequency requirements**
- Continuing to increase play-out delay yields **diminishing returns**
- **BUT...**

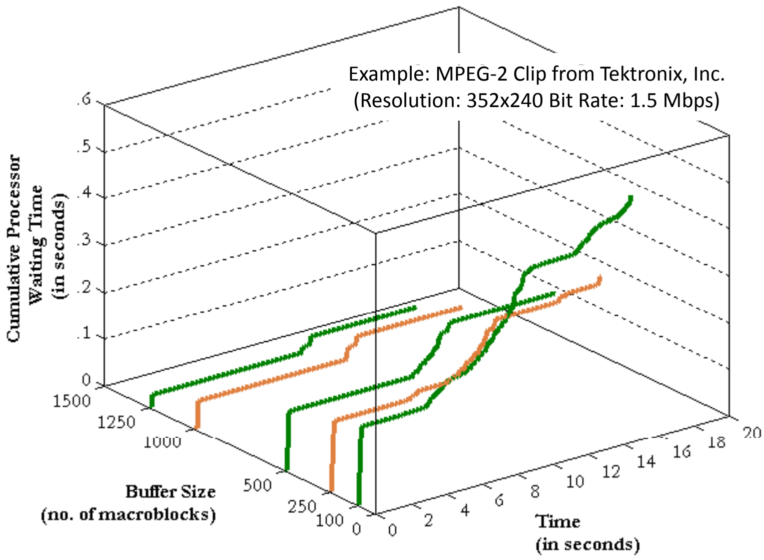
Strong Assumptions: No Processor Stall



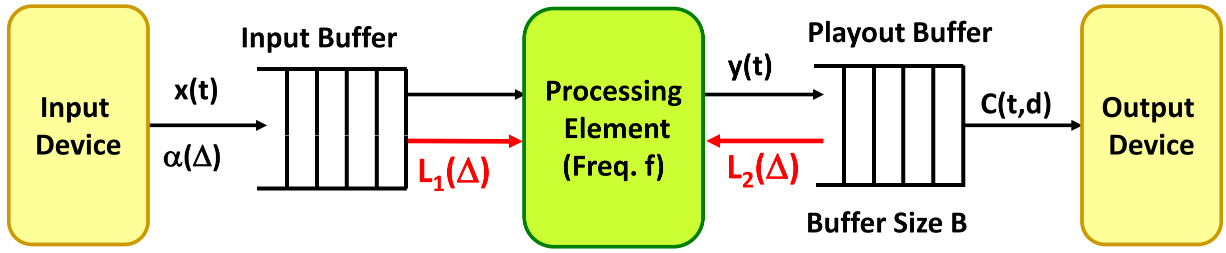
Worst-Case Buffers Are Too Large

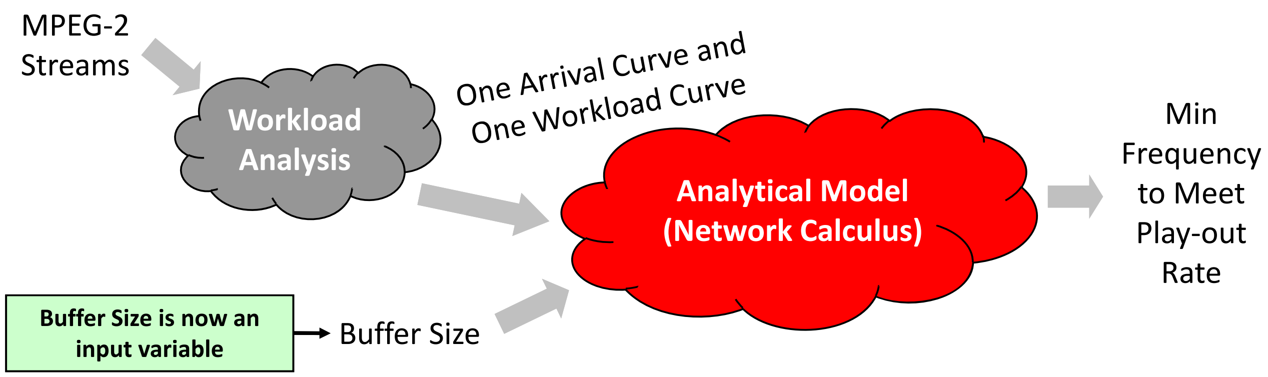
- **Decreasing the buffer size** below the worst-case size provided by the analytical model means that the processor will stall
  - Such stalls may affect the play-out rate
- One should then **increase the processor frequency**
  - **Simulation** can be used to assess play-out rate

Buffer Size [macroblocks]	Min Processor Frequency to meet Play-Out Rate [MHz]
1453 (= 1.38 MB)	114
1250	114 (play-out rate still maintained)
1000	116
500	117
250	118
100	165



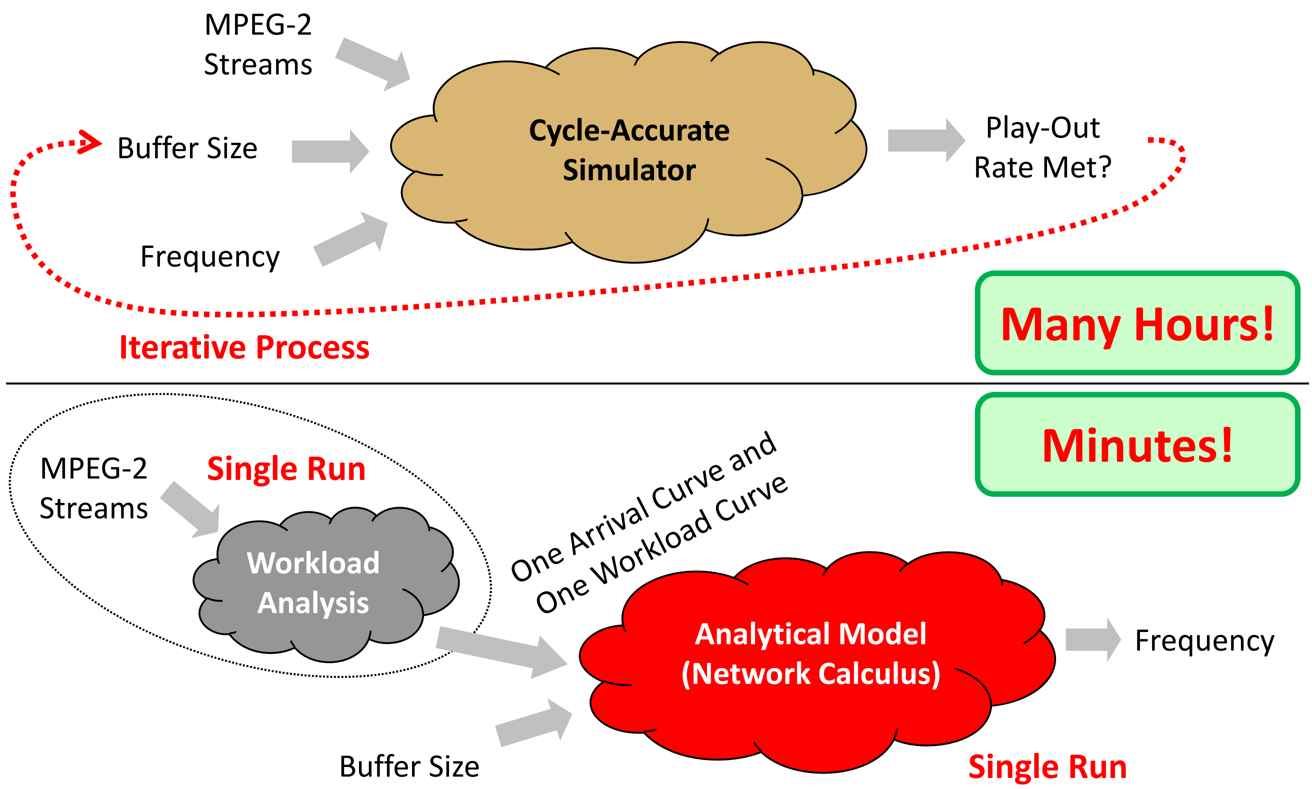
Modelling Processor Stall Time





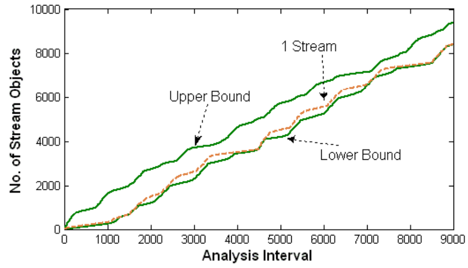
[Raman, Nicopoulos, Thiran, and Ienne, 2008]

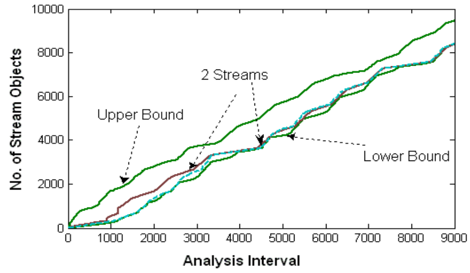
Simulation vs. Analytical Model

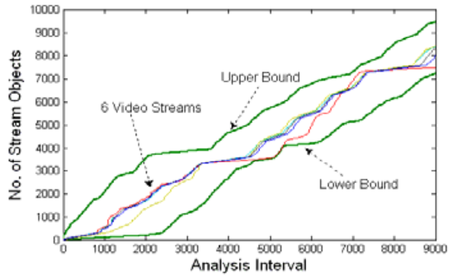


Worst-Case and Pessimism

- System behaviour is deterministic and results in **guaranteed worst-case bounds**
  - Modelling determines conservativeness of a solution (e.g., worst-case or smaller play-out buffer)
- Workload is captured **empirically** in curves that globally capture the behaviour of **all sample inputs (e.g., clips) at once**
  - They encompass a **combined behaviour** that may not be captured as accurately by each individual clip (e.g., as in simulation)







Conclusions

- Brute-force **simulation** is increasingly **unfeasible**
  - Systems and applications complexity excludes thorough exploration
- **Analytical models** are a **powerful** solution
  - Can model critical features (e.g., processor stalls)
  - Can quickly focus the solution space around good solutions
  - Simulation can validate promising solutions
- They are extremely **difficult to develop**
  - Any engineer can easily add a play-out buffer and model processor stalls
  - To develop some analytical models you need to make a PhD...