A Unified Framework for System-level Design: *Modeling and Performance Optimization of Scalable Networking System*

Hwisung Jung and Massoud Pedram

University of Southern California
Dept. of Electrical Engineering

International Symposium on Quality Electronic Design 2007

USC

Agenda

- Introduction
- Background
- A Unified Modeling Framework
- Performance Optimization
- Experimental Results
- Conclusion

Introduction

- Realistic system modeling is an important step toward:
  - optimizing performance and energy consumption
  - realizing the target system specification early in design process
- Scalable networking system requires:
  - time-to-market
  - highly efficient design cycle
- Implications of high-functionality / performance design:
  - high power densities
  - elevated temperature
  - low circuit reliability
- A unified system modeling framework enables:
  - Realization of reliable system design
  - Improvement in accuracy and robustness of energy optimization techniques

Selected Prior Work

- F. Bause (Proc. Petri Net 1993)
  - Petri Net + Queuing Model
- N. L. Benitez (Trans. Reliability 2000)
  - Petri-Net based performance evaluation
- Q. Qiu, et al. (TCAD 2001)
  - Stochastic system modeling w/ GSPN
- A. Bogliolo, et al. (IEEE 2004)
  - Continuous-Time Markov Decision Process (CTMDP) based Model
- S. Kim, et al. (TVLSI 2006)
  - Queuing model-based SOC design

Motivation

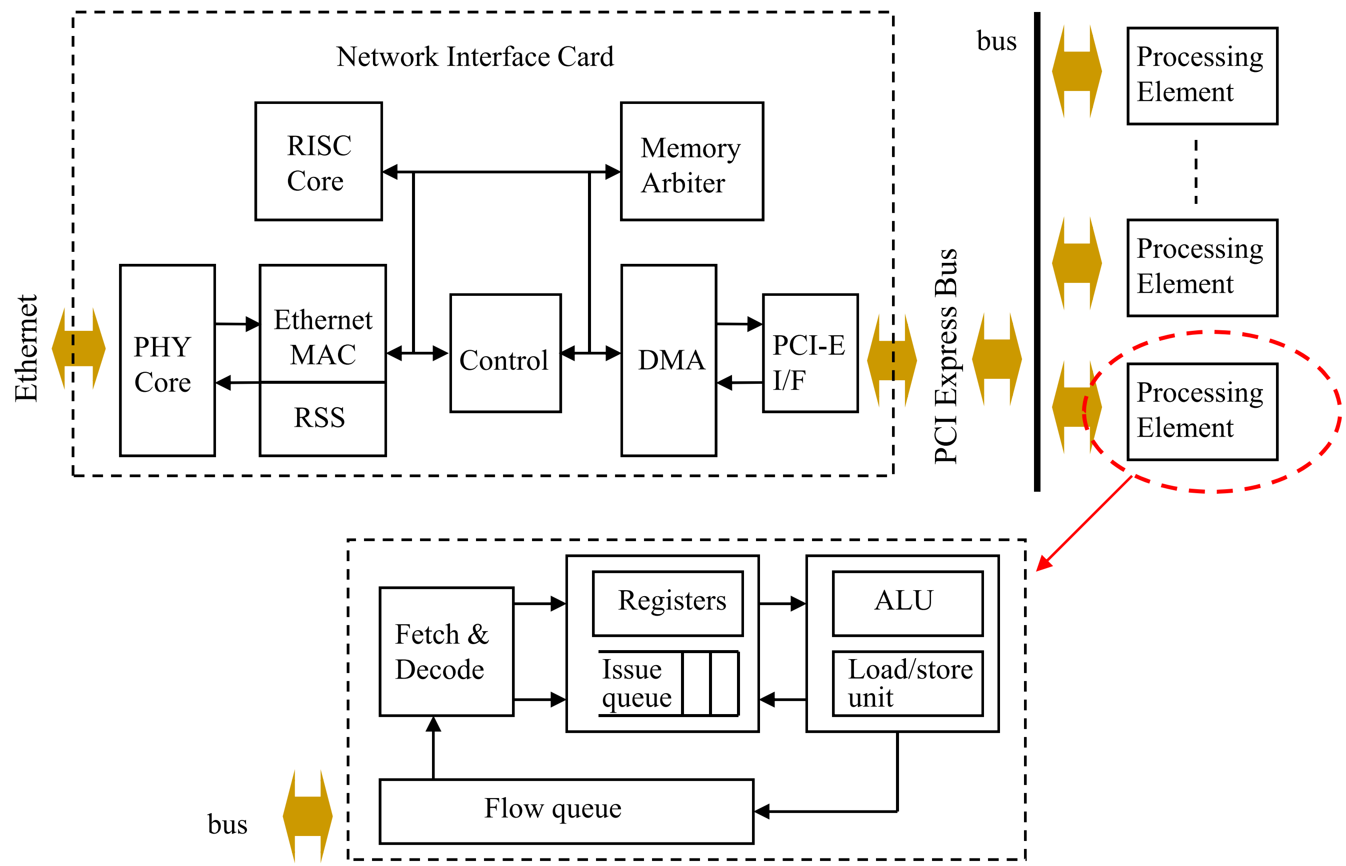
- System modeling framework must handle:
  - Concurrency, synchronization, and heterogeneity

	Pros	Cons
Queuing Network	• Models resource contention and scheduling strategies	• Not suitable for representing blocking and synchronization of processes
GSPN	• Suitable for modeling blocking and synchronization aspects • Associated w/ CTMDP	• Difficulty in representing scheduling strategies • Assumes exponential distribution for state transition

- Extended queuing Petri net (EQPN)
  - ESPN (Extended SPN: semi-Markov process) + G/M/1 queuing model

Background (1)

- Block diagram of a scalable networking system

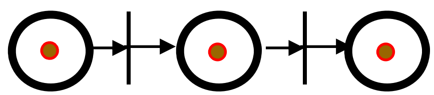


(*Refer to*: D. Bertozzi, et al., "Xpipes: A Network-on-chip Architecture for Gigascale SOC", *IEEE Magazine*, 2004)

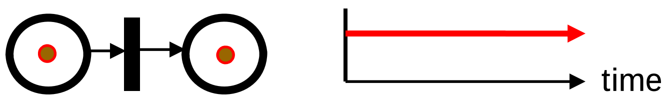
Background (2)

- ESPN (Extended Stochastic Petri Net), tuple (*P, T, E, M, F, G*)
  - *P* is a finite set of places
  - *T* is a finite set of transitions
  - *E* is a set of arcs
  - *M* is a marking
  - *F* is a set of firing rates
  - *G* is a firing function

- Immediate transition



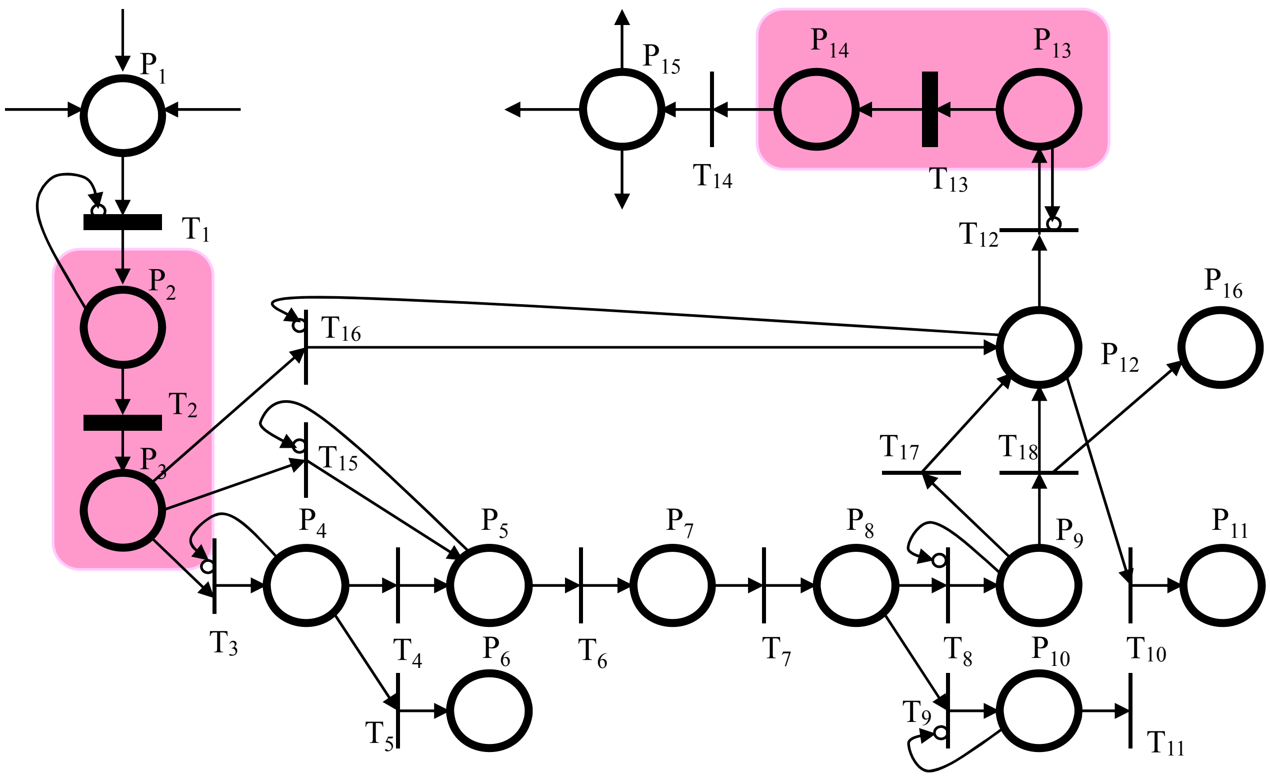
- Timed transition



- The numerical solution for ESPN is based on the Semi-Markov Process (SMP) model

A Unified Modeling Framework (1)

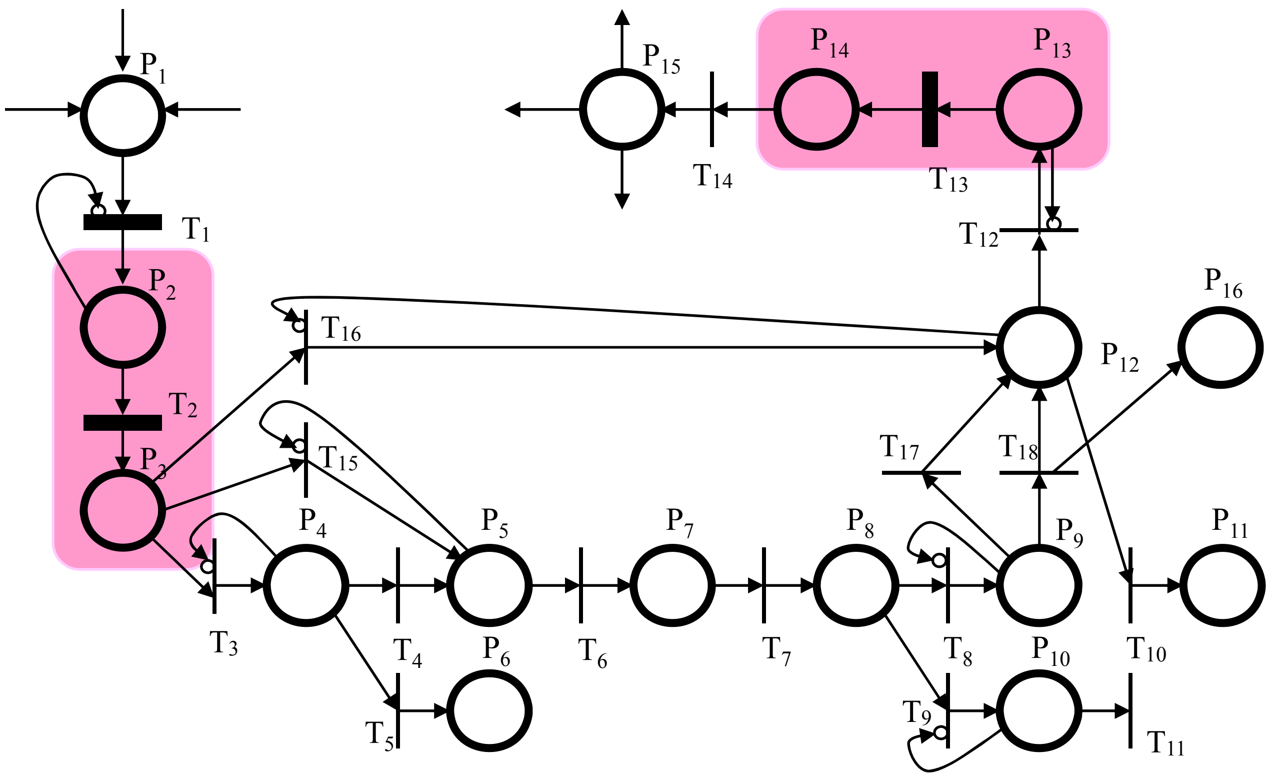
- ESPN model for a PE



Place	Description
P_1	Inbound switching
P_2	Inbound flow queue writing
P_3	Writing done
P_4	Instruction fetch
P_5	Instruction cache accessing
P_6	Instruction cache miss handling
P_7	Instruction decode
P_8	Issue queuing
P_9	Memory inst. executing
P_{10}	Integer & FP unit accessing
P_{11}	Retirement
P_{12}	Data cache accessing
P_{13}	Outbound flow queue writing
P_{14}	Writing done
P_{15}	Outbound switching
P_{16}	Data cache miss handling

A Unified Modeling Framework (2)

- ESPN model for a PE



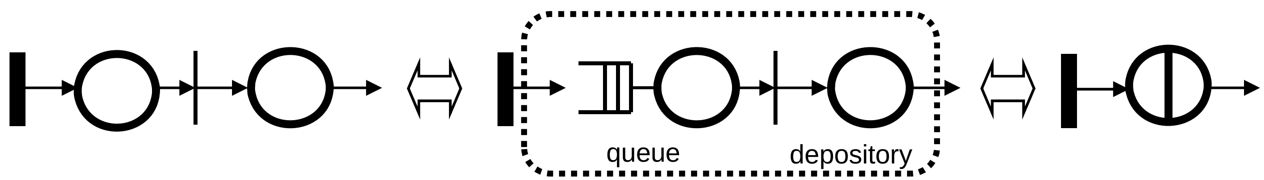
Trans	Description
T_1	Incoming switching delay
T_2	Queuing delay
T_3	Immediate transition
T_4	Immediate transition (cache hit)
T_5	Immediate transition (cache miss)
T_6	Immediate transition
T_7	Immediate transition
T_8	Memory access
T_9	Integer & FP unit access
T_{10}	Immediate transition (reg. update)
T_{11}	Immediate transition (reg. update)
T_{12}	Immediate transition
T_{13}	Queuing delay
T_{14}	Immediate transition
T_{15}	Inst. cache update
T_{16}	Data cache update
T_{17}	Immediate transition (cache hit)
T_{18}	Immediate transition (cache miss)

A Unified Modeling Framework (3)

- Queuing and scheduling mechanisms handle resource contention.
- To facilitate the queuing strategy, we extend previous models.

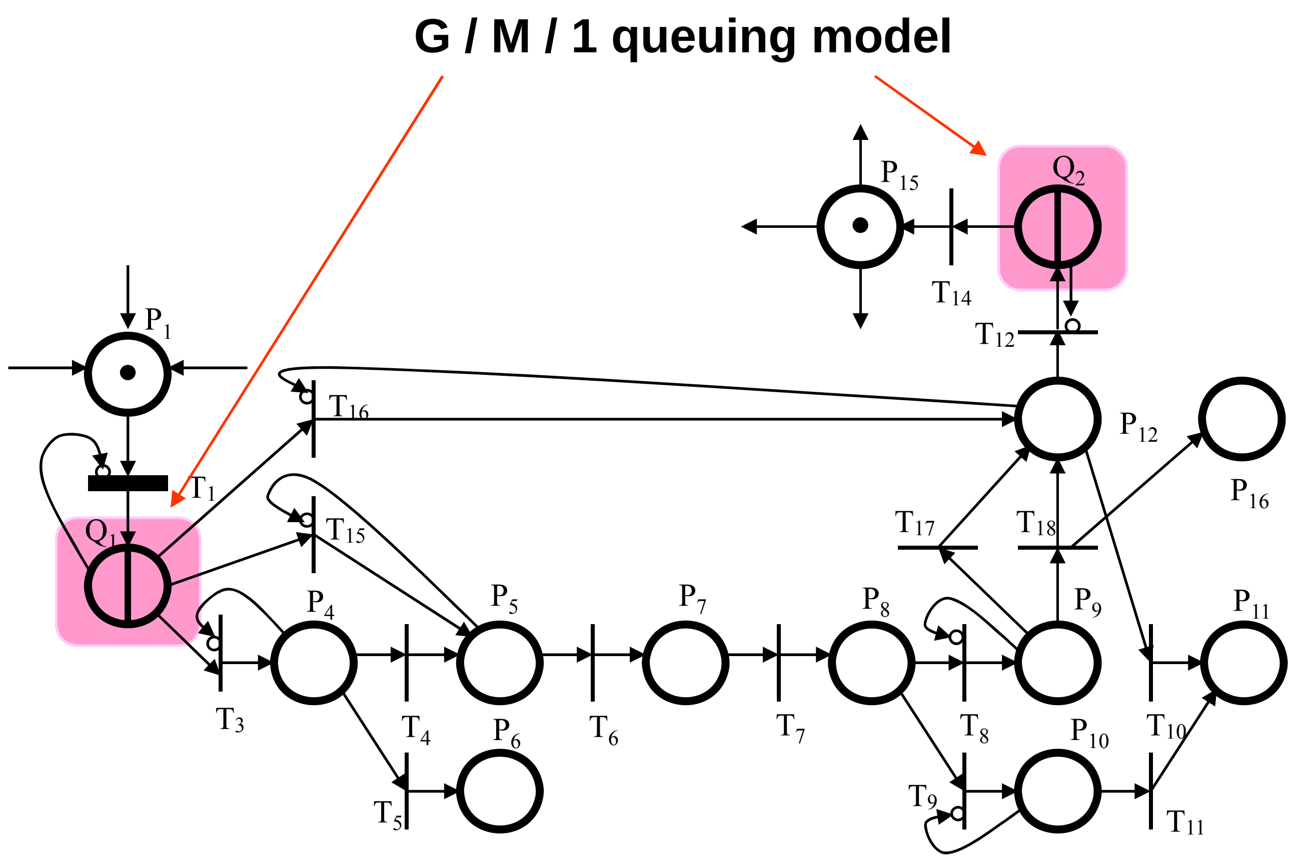
- **Definition 1:** Extended Queuing Petri Net (EQPN) is a triplet *(ESPN, PQ , W*)
  - *ESPN* is the underlying Extended Stochastic Petri Net,
  - $PQ = \{PQ_1, PQ_2\}$, where PQ_1 is the set of timed *queuing places* and PQ_2 is the set of immediate queuing places,
  - $W = \{W_1, W_2\}$, *where* W_1 is the set of timed transitions and W_2 is the set of immediate transitions.

- Queuing place consists of *a queue* and *a depository*.



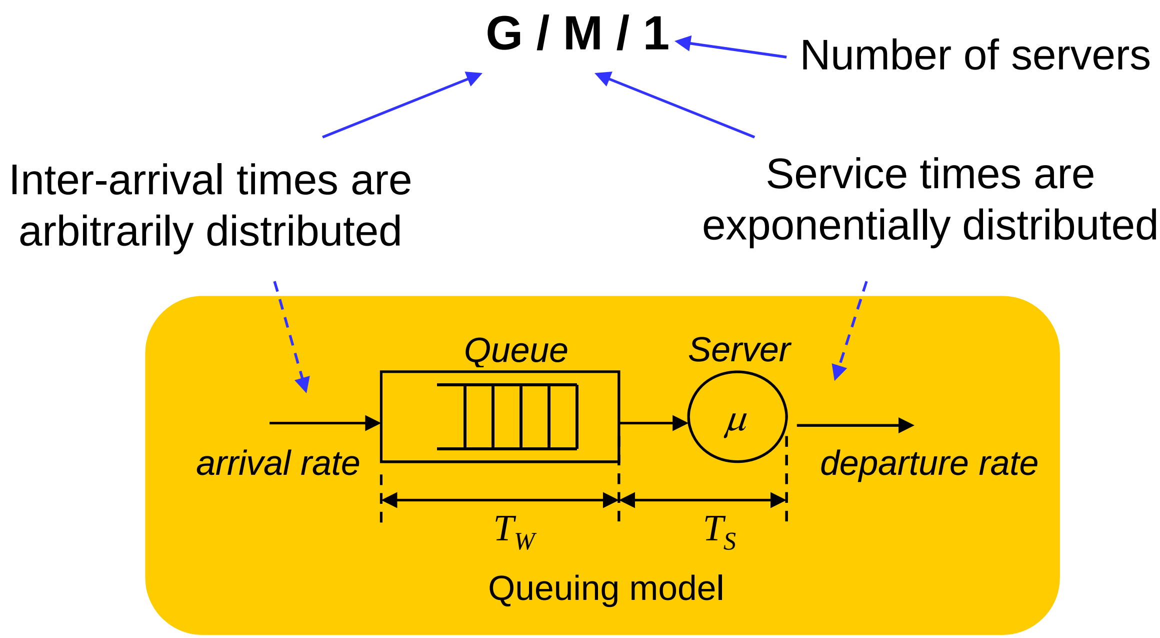
A Unified Modeling Framework (4)

- EQPN model for a PE



A Unified Modeling Framework (5)

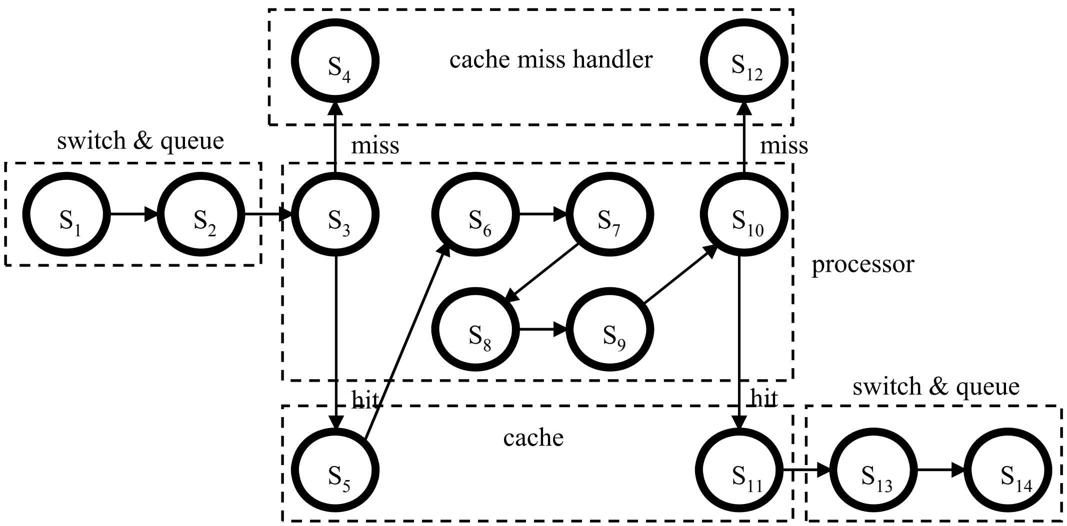
- Queuing place may be represented by the G/M/1 queuing model:



- The more commonly used M/M/1 queuing model underestimates the occurrence probability of requests with long inter-arrival times.

A Unified Modeling Framework (6)

- Reachability graph contains two types of markings:
  - *Vanishing marking* enables only immediate transitions.
  - *Tangible marking* incurs timed transitions as a function of time.
- Marking process of an EQPN is equivalent to a SMP.
- SMP model enables mathematical programming techniques for performance optimization.



State	Description
S_1	Inbound switching
S_2	Inbound flow queue writing
S_3	Instruction fetch
S_4	Instruction cache miss handling
S_5	Instruction cache access
S_6	Instruction decode
S_7	Issue queuing
S_8	Instruction executing
S_9	Integer & FP unit accessing
S_{10}	Retirement
S_{11}	Data cache accessing
S_{12}	Data cache miss handling
S_{13}	Outbound flow queue writing
S_{14}	Outbound switching

Analysis of EQPN Model

- Let *W* denote the number of waiting tasks in the PE just before a new task arrives, then we have

$$q_n = Prob\{W = n\} = (1-\gamma)\gamma^n \ , \ n = 0, 1, ..., \infty$$

where γ is the unique solution (real, $0 < \gamma < 1$) of Laplace-Stieltjes transform (LST) of the inter-arrival time distribution function.

- Let $T_{W,k}$ represent the *waiting time* in the k^{th} PE, given by

$$T_{W,k} = \gamma / \left[\mu (1-\gamma) \right]$$

- The *utilization ratio* of a PE is defined as:

$$u_k = BP_k / \left(BP_k + IP_k \right)$$

where *BP* is duration of the busy period of k^{th} PE
IP is its idle period.

- The *link utilization* (a measure of traffic workload) is defined as:

$$LU(e) = \sum_G D(e,c) / N_{clk} \qquad D(e,c) = \begin{cases} 1 & \text{if traffic passes on link } e \text{ at cycle } c \\ 0 & \text{otherwise} \end{cases}$$

where *G* is the # of clock cycle, *e* is the link path between NIC and PE, and N_{clk} is the # of clock cycles of the link given to PE.

Performance Optimization (1)

- The expected power dissipation is the summation of state-dependent power term and a transition dependent energy cost:

$$pow_{\exp}(s,a) = pow_{k}(s) + \frac{1}{\tau(s,a)} \sum_{s \in S} Prob(s' \mid s,a) ene(s,s')$$

- *K* denotes the set of PE
- *ene*(*s, s'*) is the energy required to transit from state *s* to *s'*
- τ(*s, a*) is the expected duration of the time that the PE spent in the state *s* if action *a* is chosen.

- Let a sequence of states $s^0, s^1, \ldots, s^k$ denote a processing path δ by which the PE moves from s^0 to s^k.

- For a given policy π, the average total power dissipation can be given over the set of processing paths:

$$actpow^{\pi}_{avg}(\delta) = EXP[\sum_{i=0}^{k} \alpha^{t_i} pow_{\exp}(s^i, a^i)]$$ (α: discount factor, $0 < \alpha < 1$)

Performance Optimization (2)

- To find optimal state-action sets, we must solve the following optimization problem:

$$\min \sum_{s} \sum_{a} actpow^{\pi}_{avg}(\delta)\varphi(s,a)$$

$$\text{s.t.} \quad \sum_{a} \varphi(s,a) - \sum_{s'} \sum_{a} \varphi(s',a) Prob(s' \mid s,a) = 0$$

$$\sum_{s} \sum_{a} \varphi(s,a)\tau(s,a) = 1$$

$$\sum_{k \in \delta} (T_{W,k} + T_{S,k}) \le T_d \qquad \forall \delta \in paths$$

$$BP_k \big/ (BP_k + IP_k) \ge u_k \qquad \forall k \in K$$

- $T_{W,k} = \sum_{i=1}^{n} i \cdot q_{i,k}, \quad T_{S,k} = 1 \big/ \mu_k$
- $BP_k = \sum_{i=1}^{n} q_{i,k}, \quad IP_k = q_{0,k}$
- $0 \le q_{i,k} \le 1 \quad i = 0, \dots, n$
- $\varphi(s,a) \ge 0 \quad all\ s \in S, a \in A$
- $\varphi(s, a)$ is the frequency that the system is in *s* and *a* is issued.

- The average energy dissipation of the PE may be calculated as:

$$ene_{avg} = actpow^{\pi}_{avg}(\delta) \cdot \sum_{l \in L} \sum_{k \in K} Texe_{l.k} + \sum_{k \in K} slpow_k \cdot (T_d - \sum_{l \in L} Texe_{l.k})$$

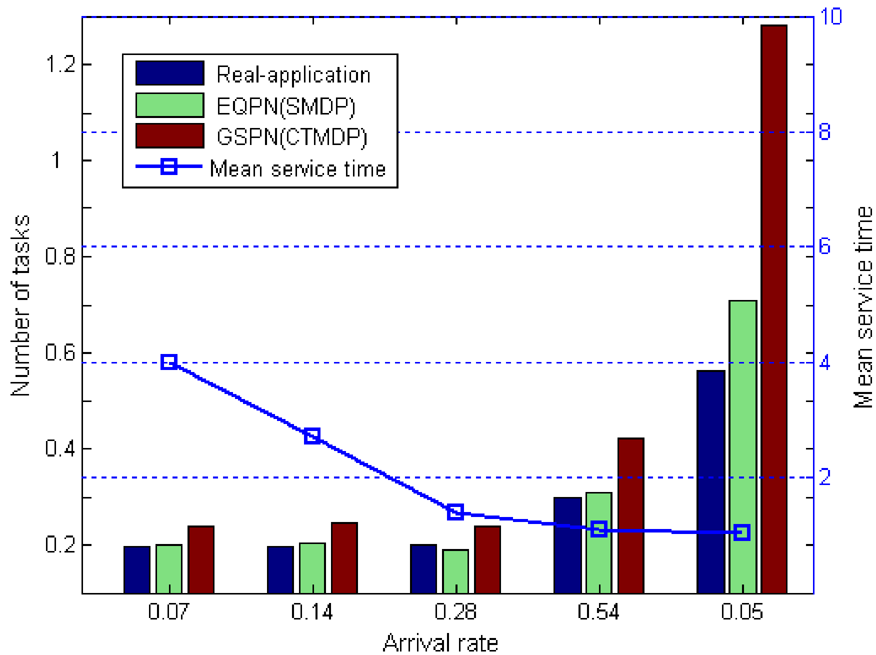
- *L* denotes the set of tasks
- $Texe_{l.k}$ is the execution time of task *l* on k^{th} PE
- T_d is the user-specified total time
- $slpow_k$ is the sleep power on *kth* PE

Experimental Results (1)

- Performance characteristics of the NIC
  - Maximum 1000Base-T full duplex bandwidth for each packet size is achieved.
  - The IP packet size is varied; The inter-packet gap is kept at 0.0096us.

Packet size (bytes)	Service rate (pkt/sec)	Inter-arrival time (sec)	Arrival rate (pkt/sec)	Service time (sec)
1518	84819	12.2E-6	81699	11.7E-6
1024	124936	8.28E-6	120656	8.00E-6
512	245100	4.19E-6	238549	4.08E-6
256	317400	2.14E-6	466417	3.15E-6
128	325200	1.12E-6	892857	3.07E-6
64	338000	0.60E-6	1644736	2.95E-6

Performance characteristics



Experimental Results (2)

- Consider UltraSPARC-II*i* model as the PE
  - Consumes 17.6W (active) at 1.7V, 650MHz, and 20mW (sleep)
  - PE has DVFS set (1.7V/650MHz, 1.6V/325MHz, and 1.5V/108MHz)
  - PE accepts both high and low priority data, where a high-priority data move ahead of all the low-priority data.

	Original model			SMP-based Optimization				
Arrival Rate of High-Priority Threads	Waiting Time at High-Priority Queue	Waiting Time at Low-Priority Queue	Energy	Frequency that the system is in state *s* and action *a* (high-priority case) [$\varphi(s, a_1)$ $\varphi(s, a_2)$ $\varphi(s, a_3)$]	Power for High-Priority Threads	Power for Low-Priority Threads	Energy	Energy Savings
0.02	0.53	1.11	64.1	0.61 0.31 0.10	13.1	6.6	62.1	3.3%
0.04	0.56	1.22	66.5	0.59 0.28 0.10	12.9	6.0	65.3	1.8%
0.06	0.60	1.35	69.6	0.55 0.27 0.09	11.9	5.6	64.8	6.9%
0.08	0.63	1.50	72.7	0.53 0.26 0.06	11.4	4.8	65.1	10.5%
0.10	0.67	1.67	76.4	0.49 0.25 0.08	10.8	4.1	64.9	15.1%

SMP-based energy optimization (normalized)

Experimental Results (3)

- Set the performance constraints on T_d and u_k
  - E.g., $T_d = 9$ and u_k varies on arrival rate
  - Consider different task arrival rates

	Original model				Proposed approach			
Arrival rate of data	Response Time T_R	Busy + Idle Period	Util. of PE	Energy	Response Time T_R	Util. of PE	Energy	Energy Savings
0.7	1.02	2.08	0.49	17.9	2.04	0.98	15.9	11.4%
0.6	0.89	2.12	0.42	15.7	1.78	0.83	13.9	11.5%
0.5	0.79	2.21	0.35	13.9	1.58	0.71	12.3	11.4%
0.4	0.74	2.64	0.28	13.0	1.48	0.56	11.6	11.4%
0.3	0.71	3.39	0.21	12.5	1.42	0.42	11.2	11.4%
0.2	0.70	5.00	0.14	12.4	1.40	0.28	10.9	11.3%
0.1	0.70	9.99	0.07	12.5	1.40	0.14	11.1	11.2%

Energy optimization under performance constraints (normalized)

Conclusion

- A unified modeling framework, EQPN, improves the modeling accuracy of the system.
- By modeling the system with EQPN, the model parameters become more realistic.
- Performance optimization problem based on a corresponding SMP was formulated and solved.
- Simulation results demonstrate system-wide energy savings up to 11.5% under performance constraints.