Scheduling of Multi-class Multi-server Queueing Systems with Abandonments

Urtzi Ayesta, Peter Jacko, and Vladimir Novak

Motivation - Abandonments

Real-world queueing systems in which customers **may abandon** if service does not start sufficiently quickly

- **Internet users** - lost of wireless signal
- **Call centers** - users' impatience
- **Health care** - irreversible deterioration of patients' health
- **Real-time systems** - data received after hard deadline
- **Inventory systems with perishable goods**

Problem description

Aim: Solving the problem of multi-class multi-server customer scheduling for a system in which we allow for abandonment.

Objective: Maximize the total discounted or time-average revenue from customers in the system

- **Revenue:** Sum of service completion rewards minus waiting costs and abandonment penalties

Main assumptions:

- *Service only from one server at a time*
- *Customers in service cannot abandon*
- *Customers in service are also charged a waiting cost*

MDP Formulation

Analyzes of the continuous-time model without arrivals

- uniformization and discretization of parameters
- time slotted into epochs $t \in \mathcal{T} := \{0, 1, 2, \dots\}$
- $K + M$ competing options, labeled by $k \in \mathcal{K}^+ := \mathcal{K} \cup \mathcal{M}$
- $\mathcal{A} := \{0, 1\}$ - *action space*

Customer k defined by

- $\mathcal{N}_k := \{0, 1\}$ - *state space*
- Expected one-period *revenue*

$$R^1_{k,0} := 0, \qquad R^1_{k,1} := -c'_k,$$
$$R^0_{k,0} := 0, \qquad R^0_{k,1} := -c'_k - d_k\theta'_k;$$

- *State-transition probability matrix*

$$\boldsymbol{P}^1_k := \begin{array}{c c} & \begin{array}{cc} 0 & 1 \end{array} \\ \begin{array}{c} 0 \\ 1 \end{array} & \begin{pmatrix} 1 & 0 \\ \mu'_k & 1-\mu'_k \end{pmatrix} \end{array}, \quad \boldsymbol{P}^0_k := \begin{array}{c c} & \begin{array}{cc} 0 & 1 \end{array} \\ \begin{array}{c} 0 \\ 1 \end{array} & \begin{pmatrix} 1 & 0 \\ \theta'_k & 1-\theta'_k \end{pmatrix} \end{array}.$$

References

- Ayesta, U., Jacko, P., & Novak, V. (2015). Scheduling of multi-class multi-server queueing systems with abandonments. **Journal of Scheduling**, 1-17.
- *Early version:* Ayesta, U., Jacko, P., & Novak, V. (2011). *2011 In IEEE Infocom:* A nearly-optimal index rule for scheduling of users with abandonment

Contact information

- Urtzi Ayesta - urtzi@laas.fr
- Peter Jacko - p.jacko@lancaster.ac.uk
- Vladimir Novak - vladimir.novak@cerge-ei.cz

Optimization Problem

$$V(\boldsymbol{X}(0)) := \max_{\boldsymbol{\pi} \in \Pi_{\boldsymbol{X},\boldsymbol{a}}} \mathbb{B}^{\boldsymbol{\pi}}_0 \left[\sum_{k \in \mathcal{K}^+} R^{a_k(\cdot)}_{k,X_k(\cdot)} \right] \tag{P}$$

$$\text{subject to } \mathbb{E}^{\boldsymbol{\pi}}_t \left[\sum_{k \in \mathcal{K}^+} a_k(t) \right] = M, \text{ for all } t \in \mathcal{T}$$

Opt. Solution for Special Cases

Single Customer at a Single Server

$$\nu^{1\mathrm{U}}_k := C_k = d_k - c_k \left(\frac{1}{\mu_k} - \frac{1}{\theta_k} \right) \tag{1}$$

Proposition 1. *C_k is the difference between the expected total revenue if serving the customer always and the expected total revenue if not serving her at all.*

Two Customers at a Single Server

$$\nu^{2\mathrm{U}}_k := \frac{C_k \theta'_k}{\theta'_k + \mu'_{3-k}}. \tag{2}$$

Solution of General Case

Larger values of K and M - analytically intractable

Whittle relaxation - using $\mathbb{B}^{\boldsymbol{\pi}}_0 [M] = M$

$$\max_{\boldsymbol{\pi} \in \Pi_{\boldsymbol{X},\boldsymbol{a}}} \mathbb{B}^{\boldsymbol{\pi}}_0 \left[\sum_{k \in \mathcal{K}^+} R^{a_k(\cdot)}_{k,X_k(\cdot)} \right] \tag{P$^{\mathrm{W}}$}$$

$$\text{subject to } \mathbb{B}^{\boldsymbol{\pi}}_0 \left[\sum_{k \in \mathcal{K}^+} W^{a_k(\cdot)}_{k,X_k(\cdot)} \right] = M.$$

Lagrange relaxation

$$\max_{\boldsymbol{\pi} \in \Pi_{\boldsymbol{X},\boldsymbol{a}}} \mathbb{B}^{\boldsymbol{\pi}}_0 \left[\sum_{k \in \mathcal{K}^+} R^{a_k(\cdot)}_{k,X_k(\cdot)} - \nu \sum_{k \in \mathcal{K}^+} W^{a_k(\cdot)}_{k,X_k(\cdot)} \right] + \nu M. \tag{P$^{\mathrm{L}}_\nu$}$$

Decomposition into Single-Option Subproblems

$$\max_{\widetilde{\pi}_k \in \Pi_{\boldsymbol{X},a_k}} \mathbb{B}^{\widetilde{\pi}_k}_0 \left[R^{a_k(\cdot)}_{k,X_k(\cdot)} - \nu W^{a_k(\cdot)}_{k,X_k(\cdot)} \right]. \tag{3}$$

Optimal Solution to Single-Option Subproblem via the Whittle Index

Let us denote for customer $k \in \mathcal{K}$, $\nu_{k,0} := 0$, and

$$\nu_{k,1} := \begin{cases} C_k \mu'_k, & \text{if } C_k \geq 0, \\ C_k \theta'_k, & \text{if } C_k < 0, \end{cases} \tag{4}$$

Theorem 1. *For problem (3), the following hold (where, in the case of equality, both actions are optimal):*

(i) *it is optimal to serve waiting customer $k \in \mathcal{K}$ if and only if $\nu \leq \nu_{k,1}$;*

(ii) *it is optimal to serve customer $k \in \mathcal{K}$ when it is already completed or abandoned if and only if $\nu \leq \nu_{k,0}$;*

(iii) *it is optimal to serve the alternative task $k \in \mathcal{M}$ if and only if $\nu \leq \nu_{k,0}$;*

Acknowledgments

Research partially supported by the French "Agence Nationale de la Recherche (ANR)" through the project ANR JCJC RACON.

Whittle Index Rule

Time-average continuous-time problem

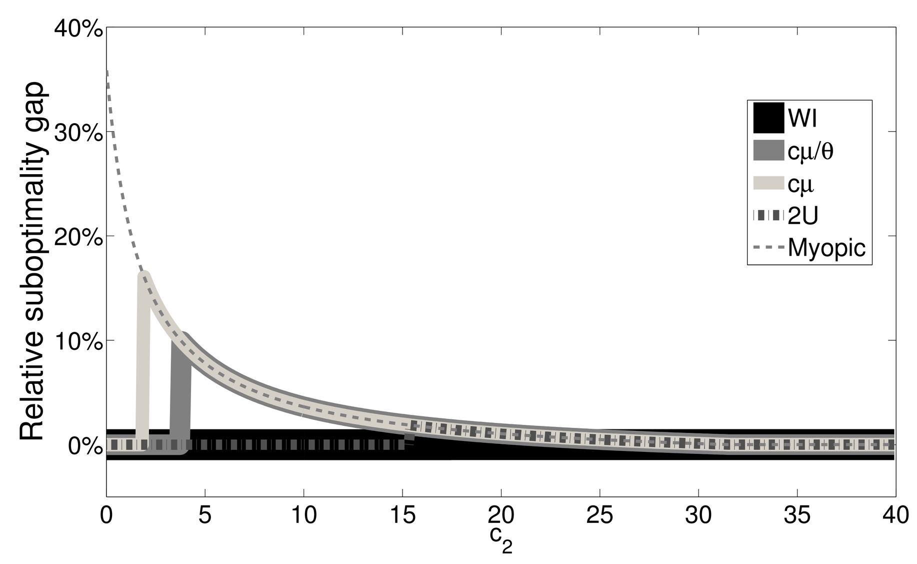
$$\xi^{\mathrm{WI}}_{k,1} := \begin{cases} C_k \mu_k = [d_k - c_k (1/\mu_k - 1/\theta_k)] \mu_k, & \text{if } C_k \geq 0 \\ C_k \theta_k = [d_k - c_k (1/\mu_k - 1/\theta_k)] \theta_k, & \text{if } C_k < 0 \end{cases}$$

Interpretation of the WI index rate

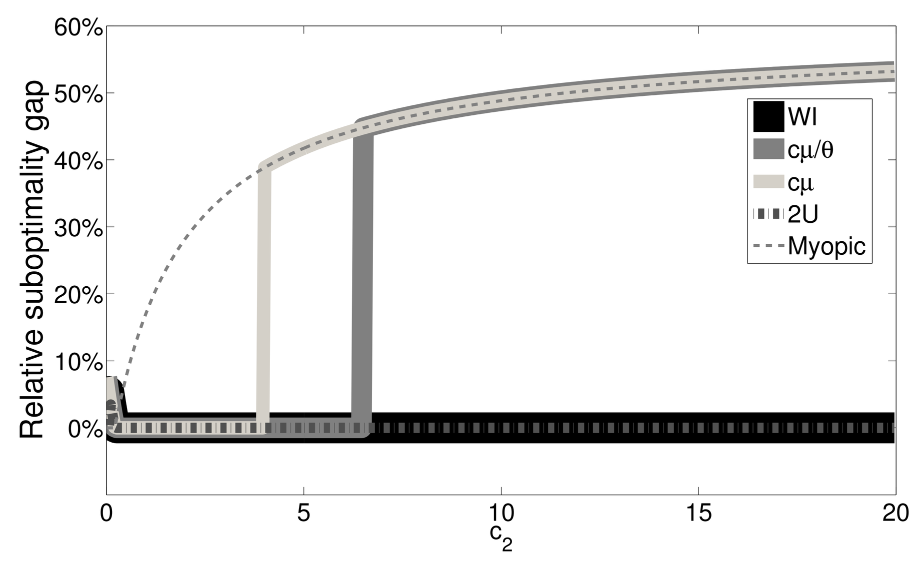
- measuring the *expected profit rate*

Computational Experiments

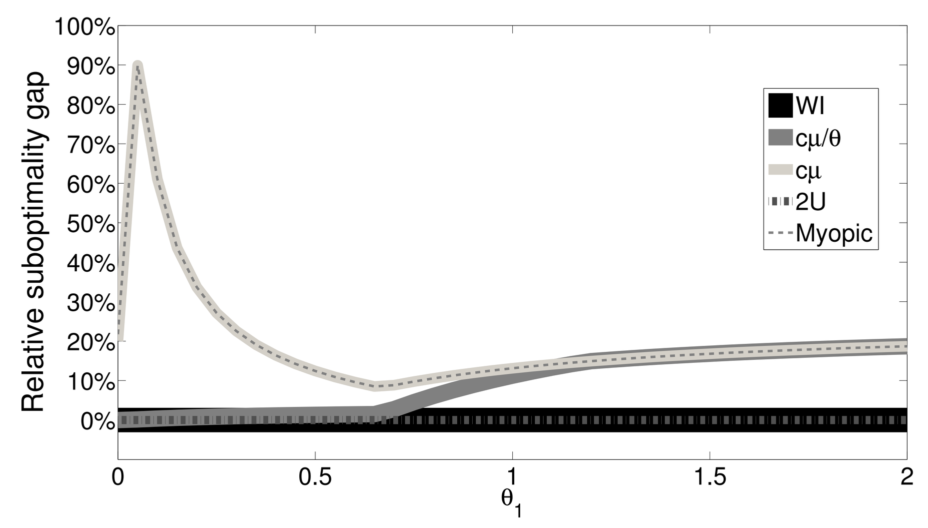
Single-server idling system



Single-server non-idling system



Multi-server idling system



Conclusion

- WI is optimal for the majority of values of the varied parameter in all the scenarios;
- WI is almost always equivalent to or outperforms $c\mu/\theta$, which is in turn almost always equivalent or outperforms $c\mu$;
- In cases in which the optimal policy chooses to idle instead of serving, WI is much better than $c\mu/\theta$ or $c\mu$;
- The switching point of 2U is often very close to WI, but usually its suboptimality region is larger;
- WI achieves *near-optimal performance* both in single-/multi-server cases, overload/underload regimes and idling/non-idling systems
- WI has *asymptotically optimal performance* (in fluid limit sense)
- Solution extends also to the discounted case