Optimal Scheduling of Services in a Queue with Impatience

Emmanuel Hyon[1] Alain Jean-Marie[2]

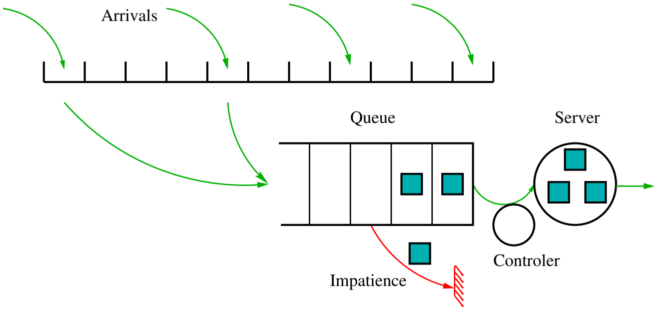
[1]Université Paris Ouest Nanterre la Défense
LIP6

[2]INRIA
LIRMM CNRS/Univ. Montpellier 2

Séminaire OCOQS
Oct 2011

Outline

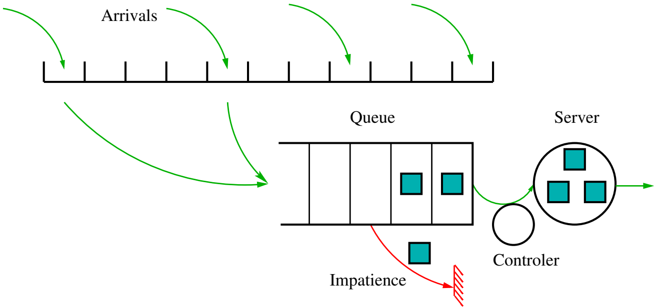
The Discrete Time Model



Arrival

- Customers arrive to an infinite-buffer queue.
- Time is discrete.
- The distribution of arrivals in each slot A_t, arbitrary with mean λ (customers/slot)

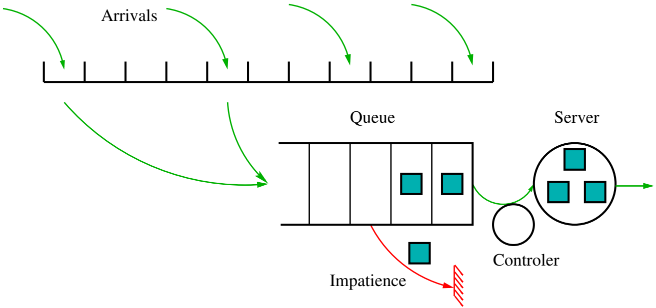
The Discrete Time Model



Services

- Service occurs by *batches* of size B.
- Service time is one slot.

The Discrete Time Model

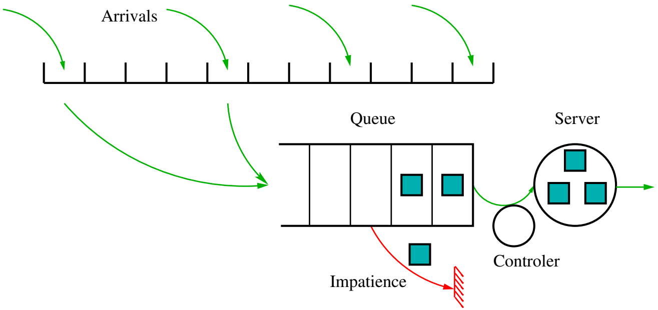


Deadline

Customers are *impatient*: they may leave before service.

- the individual probability of being impatient in each slot: α
- memoryless, geometrically distributed patience

The Discrete Time Model



Control

Service is *controlled*. The controller knows the number of customers but not their amount of patience: just the distribution.

The Question

What is the optimal *policy* π^* of the controller, so as to minimize the θ-*discounted* global cost:

$$v_\theta^\pi(x) \;=\; \mathbb{E}_x^\pi \left[\sum_{n=0}^{\infty} \theta^n \, c(x_n, q_n) \right] \;,$$

where:

- x_n: number of customers at step n;
- q_n: decision taken at step n;

and $c(x, q)$ is the cost incurred, involving:

- c_B: cost for serving a batch (*setup cost*)
- c_H: per capita *holding cost* of customers
- c_L: per capita *loss cost* of impatient customers.

Related Literature

Control of queues and/or impatience (or reneging, abandonment) : a long history.

- Optimal, deadline-based scheduling:
  - Bhattacharya & Ephremides, 1989
  - Towsley & Panwar, 1990
- Optimal admission/service control (without impatience)
  - Deb & Serfozo, 1973
  - Altman & Koole, 1998 (admission)
  - Papadaki & Powell, 2002 (service)
- Optimal *routing* with impatience
  - Kocaga & Ward, 2009
  - Movaghar, 2005

but : No optimal control of batch service in presence of stochastic impatience, so far.

Morever, **Structural Properties** when losses are hard to exhibit.

- Koole 2008,

State dynamics

x_n: number of customers in the queue at time n.
q_n the decision at slot n :
$q_n = 1$ is service occurs, $q_n = 0$ if not.

Sequence of events (at each slot)

1. Begining of the slot
2. Admission in service
3. Impatience on remaining customers
4. Arrivals

State dynamics

System Dynamics

The sequence of events leads to :

$$x_{n+1} \;=\; R(x_n, q_n) \;:=\; S\left([x_n - q_n B]^+\right) \;+\; A_{n+1} \; .$$

$S(x)$: the (random) number of "survivors" after impatience, out of x customers initially present.

$I(x)$: the number of impatient customers.
$\Longrightarrow$ binomially distributed random variables

Definition of a Policy

Definition (Policy)

A policy $\pi = (d_1, d_2, \ldots, d_n, \ldots)$ *is a sequence of decision rules.*

A decision rules from $\mathcal{H}$ the set of information (called *history*) to an action : $\mathcal{H} \mapsto q \in \mathcal{A}$.

Definition (Markov deterministic Policy)

When all the past is reduced to the state, then the rule depends only of the state it is a Markovian policy.

When the rule leads to an unique decision q we say it is deterministic.

Dynamic programming

To a policy correspond a value

Optimization criterion (discounted criteria):

$$v_\theta^\pi(x) \; = \; \mathbb{E}_x^\pi \left[\sum_{n=0}^{\infty} \theta^n \, c(x_n, q_n) \right] \; .$$

The dynamic programming

We are looking for a couple $(\pi^*, V_\theta(x)^*$, where the optimal *value function* $V_\theta(x)^*$ is solution to:

$$V_\theta(x) = \min_{q \in \{0,1\}} \{ c_B q + c_Q [x - Bq]^+ + \theta \mathbb{E} \left(V(S([x - Bq]^+) + A) \right) \}.$$

Optimality Results

Theorem

The optimal policy is of threshold type: there exists a ν such that $d(x) = 1_{\{x \geq \nu\}}$.

Theorem

Let ψ be the number defined by

$$\psi \;=\; c_B \;-\; \frac{c_Q}{1 - \overline{\alpha}\theta} \;.$$

Then,

1. *If $\psi > 0$, the optimal threshold is $\nu = +\infty$.*
2. *If $\psi < 0$, the optimal threshold is $\nu = 1$.*
3. *If $\psi = 0$, any threshold $\nu \geq 1$ gives the same value.*

Method of Proof : Structured policies

Framework: propagation of properties through the dynamic programming operator (Puterman, Glasserman & Yao).

Theorem (Puterman, Theorem 6.11.3)

Let V_w be a set of functions on the state space adequately chosen. Assume that:

0. $\forall\ v \in V_w$, $\exists\ d$, *Markov decision rule, such that* $Lv = L_d v$.

If, furthermore,

1. $v \in V^\sigma$ *implies* $Lv \in V^\sigma$,
2. $v \in V^\sigma$ *implies there exists a decision d such that* $d \in \mathcal{D}^\sigma \cap \arg\min_d L_d v$,
3. V^σ *is a closed by simple convergence.*

Then, there exists an optimal stationary policy $(d^*)^\infty$ *in* Π^σ *with* $d^* \in \arg\min_d L_d v$.

Method of Proof : One step

Property (Submodularity (Topkis, Glasserman & Yao, Puterman))

A function g is submodular if, for any $\overline{x} \geq \underline{x} \in \mathcal{X}$ *and any* $\overline{q} \geq \underline{q} \in \mathcal{Q}$*:*

$$g(\overline{x}, \overline{q}) - g(\underline{x}, \overline{q}) \leq g(\overline{x}, \underline{q}) - g(\underline{x}, \underline{q}).$$

Property (Monotone Control (Topkis, Glasserman & Yao, Puterman))

A control is said monotone if the function d $x \mapsto q$ *is monotone.*

Theorem

If $Tv(x, q)$ *is submodular over* $\mathbb{N} \times \mathcal{Q}$ *then* $x \mapsto \arg\min_q Tv(x, q)$ *is increasing in* x*.*

Proof based on the propagation on v increasing and convex.

Computation of the Optimal Threshold : sample path analysis

The system under threshold ν evolves as:

$$x_{n+1} \;=\; R_\nu(x_n) \;:=\; S\left(\left[x_n - 1_{\{x \geq \nu\}}\right]^+\right) \;+\; A_{n+1}\,. \tag{1}$$

Using (1), direct computations give:

$$V_\nu(x) = \frac{c_Q}{1 - \theta\overline{\alpha}}\left(x \;+\; \frac{\theta\lambda}{1-\theta}\right) \;+\; \psi\,\Phi(\nu, x)$$

$$\Phi(\nu, x) = \sum_{n=0}^{\infty} \theta^n \mathbb{P}(R_\nu^{(n)}(x) \geq \nu)$$

$$\psi = c_B \;-\; \frac{c_Q}{1 - \overline{\alpha}\theta}\,.$$

Lemma

The function $\Phi(\nu, x)$ is decreasing in $\nu \geq 1$, for every x.

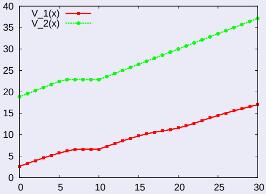
What goes wrong when $B \geq 2$

Numerical experiments and exact results in special cases reveal that:

- The value function $V(x)$ is not convex in general
- The function $TV(x, q)$ is not submodular in general

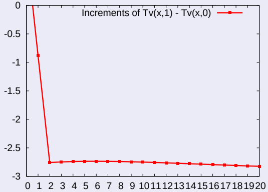
V not convex

$B = 10,\ \lambda = 1,\ \alpha = 0.1,\ \theta = 0.8$

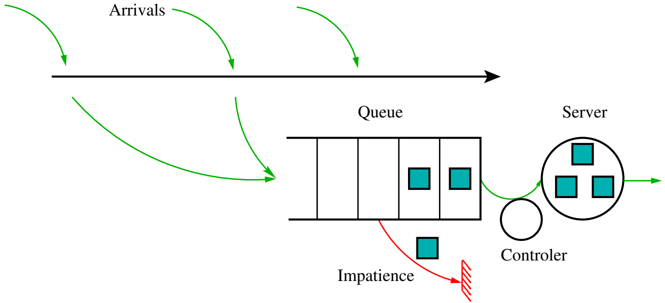


$Tv(x, q)$ not submodular

$B = 2,\ \lambda = 0.1,\ \alpha = 0.9,\ \theta = 0.9$



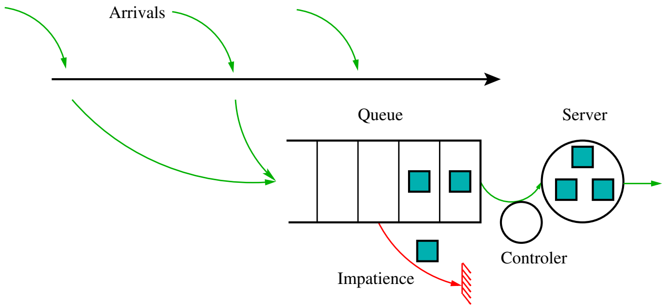
The Continuous Time Model



Arrival

- Customers arrive to an infinite-buffer queue.
- Time is continuous.
- The distribution of arrivals follows a Poisson Process with intensity λ.

The Continuous Time Model



Services

- Service occurs by *batches* of size B.
- Service time is exponentially distributed with parameter μ.
- Service can be launched only if the server is idle.

The Continuous Time Model using Puterman

The instant considered is just before the decision and just after a *natural* transition.

System Dynamic

So the dynamic is :

1. Admission in service or not (*i.e.* decision)
2. Moving instantaneously in the new state (*triggered* transition)
3. Remaining in the state until the next transition (arrival, departure, impatience) (*natural* transition).

The state space is (x, β) with :

- x denotes the number of customers which are waiting in the queue.
- β which is either 1 if the batch is busy or 0 otherwise.

The Continuous Time Model using Puterman II

The Bellman Equation associated with the MDP is (with $y = (x, \beta)$)

$$V^{\pi}(y) = \min_{q} TV^{\pi}(y,q) = C(y,q) + \frac{\Lambda(y,q)}{\Lambda(y,q)+\theta} \sum_{y'} p(y'|(y,q))V^{\pi}(y'), \quad (2)$$

where $\Lambda(y,q)$ is the (state dependant) rate of event.
Where $C(y,q)$ is the lump cost function :

$$C(y,q) = \begin{cases} \frac{\Lambda(y,q)}{\Lambda(y,q)+\theta}(x\alpha c_I) & \text{if } q = 0 \\ c_B + \frac{\Lambda(y,q)}{\Lambda(y,q)+\theta+}((x-B)\alpha c_I) & \text{if } q = 1 \text{ and } \beta = 0 \end{cases}$$

No structural properties appears in this model : "even for the lump cost"

The Continuous Time Model using Koole

The instant considered is just after the decision.

System Dynamic

1. Considering state after *triggered* transition
2. Remaining in this state until the next *natural* transition (arrival, departure, impatience).
3. Decision
4. instantaneous *triggered* transition.

The state space is the same.

The Continuous Time Model using Koole

The Bellman Equation (total cost criteria) is (in $y = (x, \beta)$) :

$$V_n(x,1) = \frac{x\alpha c_I + x c_H}{\Lambda(x,q)} + \frac{1}{\Lambda(x,q)}\Bigg(x\alpha V_{n-1}(x-1,1) + \lambda V_{n-1}(x+1,1) + \mu \min\left(V_{n-1}(x-B,1) + c_B, V_{n-1}(x,0)\right)\Bigg)$$

$$V_n(x,0) = \frac{x\alpha c_I + x c_H}{\Lambda(x,q)} + \frac{1}{\Lambda(x,q)}\Bigg(x\alpha \min\left(V_{n-1}(x-1,0), V_{n-1}(x-1-B,1) + c_B\right) + \lambda \min\left(V_{n-1}(x+1,0), V_{n-1}(x+1-B,1) + c_B\right)\Bigg)$$

Difference of the considered sets

Puterman and Koole consider different set for they structural properties even though they uses the same properties

Puterman

One uses two sets : the state space and the control set. One studies the submodularity considering one comparatively to the other.

Koole

One uses just the state space. One study the effect of a state space with a special properties between its coordinates.
N.B. the decision is binary.