



University of Groningen

## Condition-based production and maintenance decisions

uit het Broek, Michiel

DOI:  
[10.33612/diss.118424026](https://doi.org/10.33612/diss.118424026)

**IMPORTANT NOTE:** You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

*Document Version*  
Publisher's PDF, also known as Version of record

*Publication date:*  
2020

[Link to publication in University of Groningen/UMCG research database](#)

*Citation for published version (APA):*  
uit het Broek, M. (2020). *Condition-based production and maintenance decisions*. [Groningen]: University of Groningen, SOM research school. <https://doi.org/10.33612/diss.118424026>

### Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

### Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

*Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.*

# Chapter 6

## Energy-saving policies for temperature-controlled production

**Abstract.** *Numerous practical examples exist of production systems with servers that require heating in order to process jobs. Such production systems may realize considerable energy savings by temporarily switching off the heater and building up a queue of jobs to be processed later, at the expense of extra queueing costs. In this chapter, we optimize this trade-off between energy and queueing costs. We model the production system as an  $M/G/1$  queue with a temperature controlled server that can only process jobs if a minimum production temperature is satisfied. The time and energy required to heat a server depend on its current temperature, hence the setup times and setup costs for starting production are state dependent. We derive the optimal policy structure for a fluid queue approximation, called a wait-heat-clear policy. Building upon these insights, for the  $M/G/1$  queue we derive exact and approximate costs for various intuitive types of wait-heat-clear policies. Numerical results indicate that the optimal wait-heat-clear policy yields average cost savings of over 40% compared to always keeping the server at the minimum production temperature. Furthermore, an encouraging result for practice is that simple heuristics, depending on the queue length only, have near-optimal performance.*

---

This chapter is based on Uit het Broek et al. (2019d): Uit het Broek, M. A. J., G. van der Heide, N. D. van Foreest. Energy-saving policies for temperature-controlled production systems with state-dependent setup times and costs. Revision.

## 6.1 Introduction

The motivating case for this chapter is a circuit board manufacturer that uses a tin bath in their production process. This tin bath has to operate at a high temperature to yield a satisfactory production quality. Jobs arrive sporadically and the specialized nature of the jobs prohibits production in advance, i.e., jobs are make-to-order, and as a result, periods of idle time of the tin bath are unavoidable. Currently, the company continually keeps the tin bath at the minimum production temperature in anticipation of arriving jobs. However, in order to save on energy costs, an interesting option is to switch the heater off when production is idle and switch it on as soon as a certain number of jobs is available. In general, the decision to switch on the heater may even depend on both the queue length and the current temperature of the tin bath. An appropriate control of the heater may result in significant energy savings at the expense of higher queuing costs. This same trade-off applies to various other types of production processes involving high temperatures, such as galvanization, forging of metal, and high-temperature electrolysis.

The commonality in all these production processes is a server that only works at a minimum production temperature and that cools down when the heater is switched off. Therefore, the time and energy required to heat up depend on the current temperature of the bath, in other words, the setup time and setup costs are state dependent. This property makes the control of a heat bath significantly different from regular production processes, in which the setup times or costs depend only on the (sequence of) jobs to be processed. Many different policies to control the heater can be potentially useful, and for cost-effective production, it is important to understand the impact of such state-dependent setup times and costs.

To our knowledge, this chapter is the first to consider control policies for production systems with such state-dependent setup times and costs. We consider an  $M/G/1$  queue with a temperature-controlled server that can only process jobs when the temperature is above the minimum production temperature. In order to understand the dynamics of the system, we first develop a deterministic fluid queue approximation of the stochastic process. For this system, we prove that the optimal control policy has a so-called *wait-heat-clear* structure: when the heater is off it is optimal to *wait* until the queue of jobs reaches a threshold, then *heat* up at the maximum rate until the minimum production temperature is reached, and then *clear* the queue while maintaining this temperature. We then consider the  $M/G/1$  queue and derive exact and approximate costs for various classes of wait-heat-clear policies, and numerically determine (near-)optimal policies for each class. In numerical experiments, we show that policies whose decisions depend only on the queue length are typically very effective, i.e., on average within 1% of the overall optimal policy. However, there are some problem instances where an additional 10% can be saved by using control

policies that also take into account the temperature. Furthermore, we show in which cases it is reasonable to continually keep the heat bath at the production temperature.

The impact of the cost and time of setups in queueing systems has received considerable attention. A classical example is the  $M/G/1$  queue with a setup cost (see e.g., Yadin and Naor, 1963; Balachandran, 1973; Feinberg and Kella, 2002). Other common examples include  $M/G/1$  queues with setup times and/or server vacations (Welch, 1964; Heyman, 1977; Doshi, 1986; Bischof, 2001; Zhang et al., 2011). The combination of setup costs and times has only been considered by a few authors (Reddy et al., 1998; Lan and Olsen, 2006). In all of the above literature, setup costs and times are exogenously given (e.g., constant or exponentially distributed) while we consider setup times and costs that depend on the duration that the server is idle.

Others have studied policies for queueing systems that focus on minimizing energy usage while providing an acceptable service level to customers. Gandhi et al. (2010a) consider a server with high energy usage and constant setup times. Analytical results are derived for the single server case and simulation is used to study the multi-server system. Gandhi et al. (2010b) extend the system with exponentially distributed setup times. Closed-form approximations are derived for the on/off policy that turns off servers once they are idle. Phung-Duc (2017) considers the same system and derives explicit expressions for the queue length distribution. Many extensions such as generally distributed setup times, delayed off policies, and steady-state analysis have been considered (see e.g., Gandhi et al., 2014; Doroudi et al., 2017; Maccio and Down, 2018). The aforementioned studies do consider energy aware policies as we do, however, most immediately switch on a server upon arrival of a job, whereas we allow jobs to wait in a queue while the server is idle. Furthermore, these studies assume exogenously given setup times.

In the context of scheduling, the influence of state-dependent setup times is a well-studied topic (Allahverdi et al., 2008; Allahverdi, 2015). In typical scheduling problems, a prespecified set of jobs with known processing times is to be allocated to one or several machines with the objective to minimize some function of the schedule such as the makespan (see, e.g., Pinedo, 2016). An important class of problems features sequence-dependent setup times, i.e., the setup time of a job depends on the previous job (see, e.g., Nesello et al., 2018; Shen et al., 2018). An important assumption is that setup times are deterministic for a given job sequence. In contrast, setup times/costs for a heat bath are random when the decision to start heating involves the queue length. Thus far, Liu et al. (2017b) are the only to consider scheduling with machines that can be turned off and that require heating before processing starts. The authors approximate state-dependent setup times by using look-up tables with a limited number of values. We instead consider a queueing system and we exactly model the state-dependency of the setups.

The outline of this chapter is as follows. Section 6.2 introduces the production system with state-dependent setup times. In Section 6.3, we derive the optimal policy structure for the fluid queue approximation and we compute the optimal control parameters. In Section 6.4, we develop numerical procedures to determine exact and approximate costs of several control policies for the  $M/G/1$  queue. The efficacy of these different policies are compared in numerical experiments in Section 6.5. Finally, Section 6.6 concludes the chapter.

## 6.2 Model formulation

In this section, we introduce a stochastic model for serving jobs in a production system with a heat bath acting as a single server. In the simplest terms, this system behaves as follows. Jobs arrive at a queue according to a Poisson process with rate  $\lambda$ , have generally distributed processing times, and are served on a FCFS basis. The heat bath is only in a condition to serve jobs when its temperature is high enough.

We are interested in policies to control the temperature of the bath such that the average queueing and energy costs per time unit are minimized. Below we model this process as an  $M/G/1$  queue with a server whose temperature is controlled by a heater. We first introduce the state of the system and the control variable. Then we show how the temperature and queue evolve under a given control process. Finally, we introduce the cost structure and formulate an average cost minimization problem.

We represent the state of the system at time  $t$  by  $(x(t), q(t))$ , where  $x(t)$  is the temperature of the heat bath and  $q(t)$  the number of jobs in the system. The control variable  $u(t) \in [0, \beta]$  specifies the power provided by the heater at time  $t$ , where  $\beta$  is the maximum heating power. We write the control process as  $u = \{u(t)\}$ .

The temperature process  $\{x(t)\}$  depends on the power provided under the control process  $u$  and on dissipation of heat to the environment. Without loss of generality, we scale the temperature such that the ambient temperature is 0. Using the fact of nature that objects warmer than their environment cool down to the ambient temperature, the heat bath dissipates energy according to Newton's law of cooling at rate  $\alpha x(t)$ , where  $\alpha > 0$  is the heat transfer coefficient. Hence, the temperature evolves according to the differential equation

$$\frac{d}{dt}x(t) = u(t) - \alpha x(t), \quad (6.1)$$

with  $x(t) \geq 0$  for all  $t$ .

The server can only process jobs when  $x(t) \geq \bar{x}$ , where  $\bar{x}$  is the minimum production temperature. Thus, if at time  $t$  the temperature of the server is  $x(t) = \bar{x}$  and the heater is switched off, i.e.,  $u(t) = 0$ , then the temperature decreases below  $\bar{x}$ , processing stops,

and a queue starts building up. Note that the setup time, i.e., the time required to reach  $\bar{x}$ , depends on  $x(t)$ , and the longer the heater has been off, the longer it takes to reach  $\bar{x}$ . In order to ensure that  $\bar{x}$  can be reached, we require that the maximum power of the heater exceeds the dissipation rate at the production temperature, that is,  $\beta > \alpha\bar{x}$ . In particular, keeping the temperature steadily at  $\bar{x}$  requires power  $u(t) = \alpha\bar{x}$ , and the temperature  $x(t)$  cannot increase any further when  $\alpha x(t) = \beta$ .

In this chapter, we will often use the heating time  $\ell(x)$  needed to increase the temperature from  $x$  to  $\bar{x}$  when applying the maximal control  $u(t) = \beta$ . From (6.1), this can be found to be

$$\ell(x) = \frac{1}{\alpha} \log \left( \frac{\beta - \alpha x}{\beta - \alpha \bar{x}} \right). \quad (6.2)$$

In order to construct a queueing process, we assume that jobs arrive according to a Poisson process  $\{N(t)\}$  with rate  $\lambda$ . The job processing times  $\{S_i\}$  form a set of independent random variables identically distributed as a common random variable  $S$  with mean  $\mathbb{E}[S] = \mu^{-1}$  and with squared coefficient of variation  $c_S^2$ . As in any queueing system, it is required that the server load  $\rho \equiv \lambda/\mu$  is less than 1.

Now we describe how the queueing process  $\{q(t)\}$  evolves under a given control process  $u$ . Let  $N(t)$  and  $D(t)$  be the number of jobs that have arrived to and departed from the system up to time  $t$ , then the queue length at time  $t$  is

$$q(t) = N(t) - D(t). \quad (6.3)$$

Note that  $N(t)$  is a Poisson process, so it remains to construct  $D(t)$ . For this, we need to account for the fact that the server is operational only when the temperature is high enough and work is available. In terms of indicator functions<sup>1</sup>, we can write this condition at time  $t$  as

$$\mathbb{I}_{x(t) \geq \bar{x}} \mathbb{I}_{N(t) > D(t)} = 1,$$

hence, the total busy time of the server up to time  $t$  is given by

$$S(t) = \int_0^t \mathbb{I}_{x(s) \geq \bar{x}} \mathbb{I}_{N(s) > D(s)} \, ds.$$

Now, starting with  $N(0) = 0$  and  $D(0) = 0$ , we can iteratively construct  $D(t)$  from

$$D(t) = \max \left\{ k : \sum_{i=1}^k S_i \leq S(t) \right\}. \quad (6.4)$$

---

<sup>1</sup>The indicator function  $\mathbb{I}_A = 1$  if condition  $A$  is true and  $\mathbb{I}_A = 0$  otherwise

Here,  $\sum_i^k S_i$  is the total processing time required to complete the first  $k$  jobs. Hence,  $D(t)$  corresponds exactly to the number of jobs that can be completed with the total offered service up to time  $t$ . Observe that we assume that service is not lost when a job is interrupted whenever the temperature decreases below  $\bar{x}$ . We remark in passing that this assumption is not restrictive; in fact, it is easy to show that job interruptions will never happen under an optimal control.

The cost structure is such that each job in the system incurs a queuing cost  $p > 0$  per time unit, and there is a cost  $c > 0$  per unit energy so that heating at power  $u(t)$  costs  $cu(t)$  per time unit. The total expected cost for queuing and heating during  $[0, t]$  is, therefore, given by

$$J(u, t) = \mathbb{E} \left[ p \int_0^t q(s) ds + c \int_0^t u(s) ds \right]. \quad (6.5)$$

We define the long-run average cost of control  $u$  as

$$J(u) = \limsup_{t \rightarrow \infty} \frac{J(u, t)}{t}.$$

Let  $C$  be a class of admissible stationary control policies; we discuss various policy classes in more detail in Section 6.4. We have two goals. The first is to compute the minimal long-run average cost

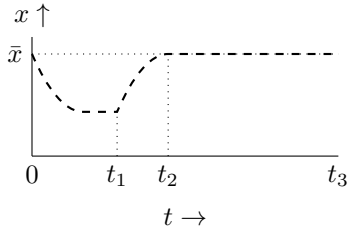
$$J^* = \inf_{u \in C} J(u). \quad (6.6)$$

The second is to determine the optimal stationary policy  $u^* \in C$  that attains  $J^*$ .

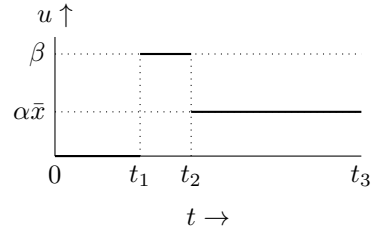
### 6.3 Deterministic fluid queue approximation

In this section, we consider an easy-to-analyze fluid queue approximation of our heat bath model, and we show that the optimal control policy for this system satisfies a wait-heat-clear policy structure. The idea behind the fluid queue approximation is to let the arrival rate  $\lambda$  go to infinity and  $\mathbb{E}[S]$  to zero such that  $\lambda \mathbb{E}[S]$  remains constant. For notational ease, we write  $\lambda$  and  $\mu$  for the rate at which work arrives and is served in the fluid setting, respectively. It turns out that, even though the approximation is deterministic, it provides significant insight into sound control rules for the stochastic setting.

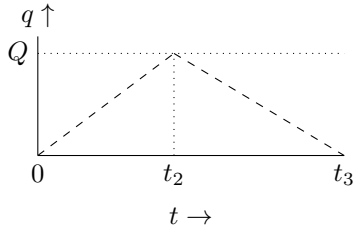
In Section 6.3.1, we sketch the system dynamics of the fluid queue under wait-heat-clear policies. Then, in Section 6.3.2, we prove the optimality of wait-heat-clear policies. Finally, we derive expressions for the average cost under such policies in Section 6.3.3.



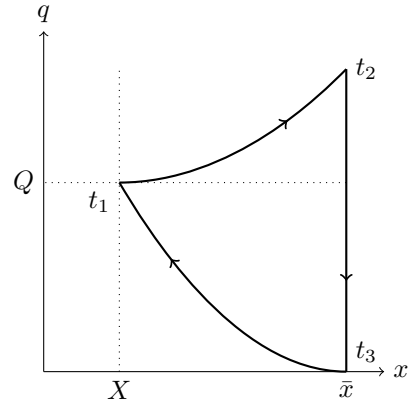
(a) Temperature



(b) Control variable



(c) Queue length



(d) Phase plot

Figure 6.1: Schematic overview of the temperature, control variable, queue length as function of time, and a phase plot, when the system is controlled by a wait-heat-clear policy.

### 6.3.1 System dynamics under wait-heat-clear policies

Assume, without loss of generality, that a cycle starts at  $t = 0$  with temperature  $x(0) = \bar{x}$ , queue length  $q(0) = 0$ , and the heater has just been switched off. The temperature then decreases and the queue length increases until the heater is switched on at time  $t_1$ , see Figure 6.1. We let the heater work at its maximum power  $\beta$  until the temperature reaches the processing temperature  $\bar{x}$  at time  $t_2$ . Now the server is operational and starts serving jobs. The heater keeps the temperature at  $\bar{x}$  up to time  $t_3$ , at which the queue is cleared. Since the system state at  $t_3$  is back at  $(\bar{x}, 0)$ , the heater switches off and a new cycle starts.

Thus, a wait-heat-clear policy satisfies

$$u(t) = \begin{cases} 0 & \text{for } 0 \leq t < t_1, \\ \beta & \text{for } t_1 \leq t < t_2, \\ \alpha\bar{x} & \text{for } t_2 \leq t \leq t_3, \end{cases} \quad (6.7a)$$



with  $0 < t_1 < t_2 < t_3$ . Under this policy, the queue increases with rate  $\lambda$  before  $t_2$  and decreases with rate  $\mu - \lambda$  after  $t_2$ , hence

$$q(t) = \begin{cases} \lambda t & \text{for } 0 \leq t < t_2, \\ \lambda t - \mu(t - t_2) & t_2 \leq t \leq t_3. \end{cases} \quad (6.7b)$$

Finally, we introduce the  $\alpha\bar{x}$ -policy that continually maintains the system at the processing temperature by applying control  $u(t) = \alpha\bar{x}$  for all  $t$ . Observe that the  $\alpha\bar{x}$ -policy is also contained in the class of wait-heat-clear policies, obtained by setting  $t_3 = \infty$  and letting  $t_2 \rightarrow 0$  and, consequently,  $t_1 \rightarrow 0$ .

### 6.3.2 The optimal policy structure

With optimal control theory (see e.g., Sethi and Thompson, 2000), we can prove that a wait-heat-clear policy is optimal for the fluid queue approximation. We will derive some preliminary lemmas before stating this result.

In order to avoid infinite queueing costs, it is evident that any optimal policy should achieve the production temperature  $\bar{x}$  at some point. Lemma 6.1 shows that to reach temperature  $\bar{x}$  at a given time  $t_2 > 0$ , it is optimal to keep the heater off as long as possible, and once on, heat up with the maximum power  $\beta$ .

**Lemma 6.1.** *Suppose, without loss of generality, that the process starts at time  $t = 0$  in state  $x(0) \leq \bar{x}$ , and  $t_2$  is so large that  $\bar{x}$  can be reached, i.e.,  $t_2 \geq \ell(x(0))$  with  $\ell$  given by (6.2). Then the optimal control to reach temperature  $x(t_2) = \bar{x}$  at  $t_2$  is given by*

$$u^*(t) = \begin{cases} 0 & \text{for } 0 \leq t < t_1, \\ \beta & \text{for } t_1 \leq t \leq t_2, \end{cases}$$

and the time  $t_1$  to switch on can be obtained from solving  $t_1 = t_2 - \ell(x(t_1))$ .

*Proof.* See appendix. □

Once the server has reached the production temperature  $\bar{x}$  and we want to process jobs, then it is optimal to keep the temperature constant at  $\bar{x}$ . In other words, it is never optimal to heat up to a temperature above  $\bar{x}$ .

**Lemma 6.2.** *During the clearing phase  $t \in [t_2, t_3]$ , the optimal heating policy is*

$$u^*(t) = \alpha\bar{x}.$$

*Proof.* See appendix. □

It remains to prove that once the temperature is  $\bar{x}$ , it is optimal to completely clear the queue, rather than switching off the heater while there is still work in the system.

**Lemma 6.3.** *While the server is working, it is optimal to keep the temperature at  $\bar{x}$  until the queue is empty.*

*Proof.* Suppose that during each cycle, policy  $(t_2, \bar{q})$  starts processing at time  $t_2$  and stops as soon as  $q(t) \leq \bar{q}$  for some  $\bar{q} > 0$  and  $t \geq t_2$ . Now consider the policy  $(t_2, 0)$ . In steady state, policy  $(t_2, 0)$  has the same cycle length and the same heating cost as  $(t_2, \bar{q})$ , however, the average queueing cost per time unit under policy  $(t_2, 0)$  is  $p\bar{q}$  lower. As we constructed a strictly better policy, a policy  $(t_2, \bar{q})$  with  $\bar{q} > 0$  cannot be optimal, hence  $\bar{q} \leq 0$ .  $\square$

The optimality of wait-heat-clear policies follows by combining the above lemmas.

**Theorem 6.1.** *The average-cost optimal policy for the fluid queue is a wait-heat-clear policy, i.e., of the form (6.7).*

### 6.3.3 Costs under wait-heat-clear policies

We next derive expressions for the average cost under a wait-heat-clear policy. To determine the cycle cost and time, we move backwards in time, step-by-step, from the clearing phase  $[t_2, t_3]$ , to the heating phase  $[t_1, t_2]$ , up to the waiting phase  $[0, t_1]$ . We label these phases such that phase 1 is the waiting phase, phase 2 the heating phase, and phase 3 the clearing phase. For each phase  $i$ , we determine the time  $T_i(\cdot)$  and cost  $V_i(\cdot)$  from the start of phase  $i$  until the end of the cycle.

We start with the clearing phase  $[t_2, t_3]$ . If  $(x(t_2), q(t_2)) = (\bar{x}, q)$ , the time to clear the queue must be  $T_3(q) = q/(\mu - \lambda)$ , as the net drain rate is  $\mu - \lambda$ . Next, there is a heating cost  $c\alpha\bar{x}$  per unit time to keep the system at temperature  $\bar{x}$ . The total queueing cost must be equal to  $p$  times the area of the triangle with height  $q$  and base  $T_3(q)$  (since at time  $T_3(q)$  the queue is empty). Thus, the time and cost to reach  $(\bar{x}, 0)$  starting from  $(\bar{x}, q)$  are given by

$$T_3(q) = \frac{q}{\mu - \lambda},$$

$$V_3(q) = \left(c\alpha\bar{x} + p\frac{q}{2}\right)T_3(q) = \frac{c\alpha\bar{x}q}{\mu - \lambda} + \frac{pq^2}{2(\mu - \lambda)}.$$

Next we consider the heating phase  $[t_1, t_2]$ , which starts in state  $(x(t_1), q(t_1)) = (x, q)$ . The heating time  $\ell(x)$  to heat the bath from temperature  $x$  to  $\bar{x}$  follows from (6.2). As the queue increases by  $\lambda\ell(x)$  during the heating time  $\ell(x)$ , the time

$T_2(x, q)$  and cost  $V_2(x, q)$  to heat *and* clear when the system starts in state  $(x, q)$  thus equals

$$\begin{aligned} T_2(x, q) &= \ell(x) + T_3(q + \lambda\ell(x)) = \frac{q + \mu\ell(x)}{\mu - \lambda}, \\ V_2(x, q) &= p \left( q + \frac{\lambda\ell(x)}{2} \right) \ell(x) + c\beta\ell(x) + V_3(q + \lambda\ell(x)) \\ &= \frac{p}{2} \frac{q^2}{\mu - \lambda} + \frac{c\alpha\bar{x}q}{\mu - \lambda} + \frac{pq\ell(x)}{\mu - \lambda} \\ &\quad + c \left( \frac{\alpha\bar{x}}{\mu - \lambda} + \beta \right) \ell(x) + \frac{p}{2} \frac{\lambda\mu}{\mu - \lambda} (\ell(x))^2. \end{aligned}$$

The final phase is the waiting phase  $[0, t_1]$ . Starting from  $(x(0), q(0)) = (\bar{x}, 0)$ , we wait until the queue length equals  $q$ , so that the waiting time is  $t_1 = q/\lambda$ . From the ODE  $\dot{x}(t) = -\alpha x(t)$  with initial condition  $x(0) = \bar{x}$ , it follows that the temperature at  $t_1$  is  $x(t_1) = \bar{x}e^{-\alpha t_1}$ . So, when the heater switches on when the queue length is  $q$ , the cycle time  $T_1(q)$  and cycle cost  $V_1(q)$  are

$$\begin{aligned} T_1(q) &= t_1 + T_2(x(t_1), q), \\ V_1(q) &= p \frac{q}{2} t_1 + V_2(x(t_1), q). \end{aligned}$$

All in all, the average cost under a policy that starts heating when the queue length is equal to  $q > 0$  is therefore  $J(q) = V_1(q)/T_1(q)$ . The critical points of  $J(q)$  can be found by equating the derivative  $J'(q)$  to 0 and solving for the optimal  $q^*$ . A simple closed-form solution does not exist; in Section 6.5 we deal with this problem numerically. Finally, consider the  $\alpha\bar{x}$ -policy. Because there is no queue when the bath always satisfies the production temperature, the average cost under the  $\alpha\bar{x}$ -policy is  $J(\alpha\bar{x}) = c\alpha\bar{x}$ .

## 6.4 Wait-heat-clear policies for the $M/G/1$ queue

The structural results from the fluid queue approximation are the main motivation to also analyze wait-heat-clear policies for the  $M/G/1$  queue with a temperature-controlled server. We will restrict the analysis to three intuitive types of wait-heat-clear policies, illustrated in Figure 6.2. All policies wait, with the heater off, until a specified threshold level is reached, and then trigger the heating and clearing phase. The  $Q$ -policy switches on the heater when the queue length reaches  $Q$ , irrespective of the temperature. The  $X$ -policy switches on when the heat bath cools down to  $X$ , irrespective of the queue length. The  $Q$ - and  $X$ -policy are typically not capable of reacting to sudden events. For example, when more customers arrive than expected

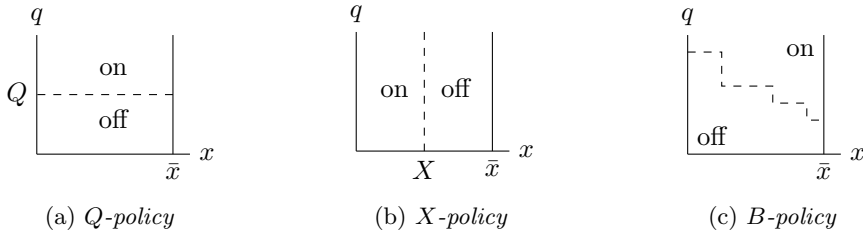


Figure 6.2: Three different policies for deciding when to start heating and clearing.

while the heat bath is still rather warm, it may be worthwhile to turn on the heater and quickly clear the queue. Therefore, we also consider the  $B$ -policy, which depends on *both* the queue length and the temperature, and switches on when a temperature-dependent threshold for the queue length is reached.

We do not formally prove that wait-heat-clear policies are optimal. However, for states on the boundary  $x = \bar{x}$ , we can show that it is optimal to keep the temperature at  $\bar{x}$  until all jobs are cleared, using the same line of proof as in Lemmas 6.2 and 6.3. For all other states, we use some simplifying assumptions to formulate a Markov decision process to determine optimal actions. Without any exception, we find that the optimal policy of the MDP has a wait-heat-clear structure for every considered instance in our numerical analysis. Thus, there is numerical support for the optimality of wait-heat-clear policies.

The remainder of this section is organized as follows. In Section 6.4.1, we derive the exact expected cost for the heating and clearing phase, which apply to all our policies. Then, we derive the exact average costs for the  $Q$ - and  $X$ -policy in Section 6.4.2. Since the  $B$ -policy is numerically difficult to evaluate exactly, we propose an approximation for its expected average cost in Section 6.4.3. Moreover, we provide a heuristic for determining near-optimal  $B$ -policies in Section 6.4.4. Finally, we formulate the aforementioned Markov decision process in Section 6.4.5.

### 6.4.1 Expected cost and time for heating and clearing

Recall that under a wait-heat-clear policy, once the system enters a state in which the heater is switched on, the heater stays on until the system is completely cleared. Thus, for any such policy, we need to know the expected cost of heating and clearing when the heater switches on in some state  $(x, q)$ . Similar to the analysis in Section 6.3.3, we therefore first compute  $V_3(q)$  and  $T_3(q)$ , i.e., the expected cost and time to clear the system once state  $(\bar{x}, q)$  is reached and the processing of jobs can start. Then we find expressions for  $V_2(x, q)$  and  $T_2(x, q)$ , i.e., the expected cost and time for heating and clearing until the end of a cycle when the heater is switched on in state  $(x, q)$ .

The clearing phase starts at  $(\bar{x}, q)$  and process all jobs until  $(\bar{x}, 0)$ . Interestingly, the analysis of an  $M/G/1$  queue under an  $N$ -policy without setup times involves the same phase: once the queue hits level  $N$ , the server switches on and, as there is no setup-time, processing can start right away. This model is analyzed in Tijms (1994, Example 1.3.4), which, after incorporating heating costs, leads to the following result.

**Lemma 6.4.** *If the queue length is  $q$  when the production temperature  $\bar{x}$  is reached, the expected time and cost to clear the system are*

$$T_3(q) = \frac{q}{\mu - \lambda},$$

$$V_3(q) = \frac{p}{2(\mu - \lambda)}q^2 + \left( p \frac{\lambda c_S^2 + \mu}{2(\mu - \lambda)^2} + \frac{c\alpha\bar{x}}{\mu - \lambda} \right) q.$$

*Proof.* Tijms (1994, Eq. 1.3.3–1.3.4) directly implies that  $T_3(q) = q/(\mu - \lambda)$  and

$$V_3(q) = \left( p \frac{q+1}{2} + p \frac{\lambda}{2} \frac{1 + c_S^2}{\mu - \lambda} + c\alpha\bar{x} \right) T_3(q)$$

□

We next consider the expected cost and time for heating the bath and clearing the queue when the heater is switched on in state  $(x, q)$ . Recall that the heating time  $\ell(x)$  is deterministic, the number of arrivals during time  $\ell(x)$  is the Poisson random variable  $N(\ell(x))$  with mean  $\lambda\ell(x)$ . Thus, the queue length at the start of the clearing phase becomes  $q + N(\ell(x))$ . This idea underlies the proof of the next lemma.

**Lemma 6.5.** *Starting at  $(x, q)$  with  $x \leq \bar{x}$ , the expected time and cost for heating and clearing until the end of the cycle is*

$$T_2(x, q) = \ell(x) + T_3(q + \lambda\ell(x)) = \frac{q}{\mu - \lambda} + \frac{\mu}{\mu - \lambda}\ell(x),$$

$$V_2(x, q) = \left( pq + p \frac{\mathbb{E}[N(\ell(x))]}{2} + c\beta \right) \ell(x) + \mathbb{E}[V_3(q + N(\ell(x)))]$$

$$= V_3(q) + \frac{p\lambda\mu}{\mu - \lambda}\ell(x)^2 + \frac{p\mu}{\mu - \lambda}\ell(x)q$$

$$+ \left( \frac{p\lambda + c\alpha\bar{x}\lambda}{\mu - \lambda} + p \frac{\lambda^2(1 + c_S^2)}{2(\mu - \lambda)^2} + c\beta \right) \ell(x).$$

Here,  $\ell(x)$  is given by (6.2), and  $V_3$  and  $T_3$  are given by Lemma 6.4. All involved constants are positive, hence  $V_2(x, q)$  is a quadratic, strictly increasing function of  $q$  and  $\ell(x)$ , and  $T_2(x, q)$  is linear.

*Proof.* See appendix. □

Finally, we are interested in the expected cost and duration of a cycle under the condition that the system switches on at a given state  $(x, q)$ . With these expressions we can easily evaluate the expected average cost for the  $Q$ -,  $X$ - and  $B$ -policies.

**Lemma 6.6.** *For cycles whose sample paths enter state  $(x, q)$  and then switch on the heater, the expected cycle time is*

$$T_1(x, q) = t_1(x) + T_2(x, q),$$

where  $t_1(x) = \frac{1}{\alpha} \log(x/\bar{x})$  is the deterministic time to cool down from  $\bar{x}$  to  $x$ . For cycles whose sample paths enter  $(x, q)$  due to an arrival, the expected cost is

$$V_1(x, q) = (q - 1)t_1(x)/2 + V_2(x, q).$$

Otherwise, for cycles whose sample paths enter  $(x, q)$  due to a temperature decrease, the expected cost is

$$V_1(x, q) = qt_1(x)/2 + V_2(x, q).$$

*Proof.* It is evident that the expected cycle time is the waiting time  $t_1$  plus the time  $T_2$  to entirely clear all jobs from the system once the heater is switched on. For the cost, suppose the sample path enters state  $(x, q)$  due to an arrival. Just prior to entering  $(x, q)$  the state was  $(x, q - 1)$ , hence the average queue length until the heater is switched on equals  $(q - 1)t_1(x)/2$ . Otherwise, the state just prior to entering  $(x, q)$  must have been  $(x + \epsilon, q)$  for  $\epsilon \downarrow 0$ . In that case, the average queue length must be  $qt_1(x)/2$ .  $\square$

It is important to observe that the  $Q$ - and  $B$ -policy switch on only due to arrivals, and the  $X$ -policy only due to decreases in temperature. The expected cost and time of a given policy follow by determining the combined probability of all sample paths entering state  $(x, q)$  and then taking the corresponding expectations over all  $(x, q)$  where the policy switches on the heater.

#### 6.4.2 Exact costs of $Q$ - and $X$ -policies

We continue by deriving the exact costs of the  $Q$ - and  $X$ -policy. Recall that in the  $Q$ -policy, the heating phase starts as soon as the queue length hits  $Q$ , and, since jobs arrive as single units, the queueing process can not ‘jump’ over  $Q$ . As the cycle starts with an empty queue, we have to wait for  $Q$  arrivals. Since the interarrival times of the jobs are exponentially distributed, the time  $t_1$  is Erlang( $\lambda, Q$ ) distributed. Since, evidently,  $q(t_1) = Q$ , we obtain the following.

**Lemma 6.7.** *Provided  $Q > 0$ , the expected average cost of a  $Q$ -policy is given by  $J(Q) = V_1(Q)/T_1(Q)$ , where*

$$T_1(Q) = \frac{Q}{\lambda} + \mathbb{E}[T_2(x(t_1), Q)]$$

$$V_1(Q) = p \frac{Q(Q-1)}{2\lambda} + \mathbb{E}[V_2(x(t_1), Q)].$$

Here,  $x(t_1) = \bar{x}e^{-\alpha t_1}$  and  $t_1$  is an Erlang( $\lambda, Q$ ) distributed random variable with mean  $\mathbb{E}[t_1] = Q/\lambda$ , and  $T_2$  and  $V_2$  are given in Lemma 6.5.

Note that, by following the reasoning of Lemma 6.6, the queueing costs in  $V_1(Q)$  are incurred only for the first  $Q - 1$  arrivals, as the  $Q$ -th arrival occurs exactly at  $t_1$ .

The  $X$ -policy starts the heating phase when the temperature equals  $X$ . The time  $t_1$  to cool down from  $\bar{x}$  to  $x$  is deterministic, hence the number of arrivals  $N(t_1)$  during  $t_1$  is Poisson( $\lambda t_1$ ) distributed. Consequently, we obtain the following.

**Lemma 6.8.** *Provided  $X < \bar{x}$ , the expected average cost of an  $X$ -policy is  $J(X) = V_1(X)/T_1(X)$ , where*

$$T_1(X) = t_1 + \mathbb{E}[T_2(X, N(t_1))] = \frac{\mu}{\mu - \lambda}(t_1 + \ell(X)),$$

$$V_1(X) = p \frac{\lambda t_1^2}{2} + \mathbb{E}[V_2(X, N(t_1))].$$

Here,  $t_1 = \frac{1}{\alpha} \log(\bar{x}/X)$  is the solution for  $X = \bar{x}e^{-\alpha t}$ .

Simple closed-form expressions for the average costs do not exist. We therefore evaluate the expressions in Lemmas 6.7 and 6.8 numerically.

In case  $Q = 0$  or  $X = \bar{x}$ , we need to use the average cost  $J(\alpha\bar{x})$  of the  $\alpha\bar{x}$ -policy. In this case the server is always operational, hence the queue length process is the same as for the  $M/G/1$  queue. Using Little's law and the Pollaczek-Khintchine equation for the average sojourn time of a job in the system, c.f., Tijms (1994), the average cost is

$$J(\alpha\bar{x}) = p\rho + p \frac{1 + c_S^2}{2} \frac{\rho^2}{1 - \rho} + c\alpha\bar{x}. \quad (6.8)$$

Here, the first term corresponds to the cost of having a job in service, the second term to the average time in queue, and the last term to the heating costs.

*Remark 6.1.* Typically, the coefficient of variation of an Erlang random variable is quite small, so it makes sense to replace the random variable  $t_1$  by its mean. For the  $Q$ -policy we then obtain from Lemma 6.7,

$$T_1 \approx Q/\lambda + T_2(x(Q/\lambda), Q),$$

$$V_1 \approx pQ(Q-1)/2\lambda + V_2(x(Q/\lambda), Q),$$

which are closed-form because the expressions in Lemmas 6.4 and 6.5 are closed-form.

### 6.4.3 Approximate costs of $B$ -policies

As it is numerically challenging to determine exact costs of  $B$ -policies, we propose an approximation of the cycle time and cycle costs that can be evaluated efficiently. Let us first explain the numerical challenge with exact costs. Ideally, we would like to determine the combined probability of all sample paths entering state  $(x, q)$  and then use Lemma 6.6 to calculate the expected time and cost. In order to enter  $(x, q)$ , a sample path should not visit any other state where the heater is switched on. For the  $Q$ - and  $X$ -policy this is automatically the case: the heater switches on only when  $Q$  or  $X$  is reached. However, for the  $B$ -policy sample paths may exist where the heater is switched on because the queue exceeds the temperature-dependent threshold for some temperature higher than  $x$ . Hence, we need to condition on the sample paths being below this threshold for all temperatures higher than  $x$ . This conditioning gives rise to integral expressions that are difficult to evaluate.

For the above reason, we discretize the temperature scale and assume exponential interarrival times for temperature changes. The temperature is discretized with a step size  $\delta$ , so that  $x \in \{0, \delta, 2\delta, \dots, \bar{x} - \delta, \bar{x}\}$ . With step size  $\delta$ , the temperature decreases from  $x$  to  $x - \delta$  at rate  $\alpha x/\delta$ . This exponential approximation is reasonable when  $\delta$  is small. We can now apply finite Markov chain theory (see, e.g., Kemeny and Snell, 1976) in order to evaluate the expected cycle time and cycle costs of  $B$ -policies.

We define a  $B$ -policy in terms of a stopping set and a continuation set. Let  $B$  be a decreasing step-function of the temperature, such as in Figure 6.2c. Then define the stopping set  $\mathcal{D}$  as

$$\mathcal{D} = \{(x, q) : q \geq B(x)\},$$

and the continuation set  $\mathcal{C}$  as the complement of  $\mathcal{D}$ , i.e.,

$$\mathcal{C} = \{(x, q) : q < B(x)\}.$$

For instance, in Figure 6.2a,  $\mathcal{C}$  is the set of points below the line  $Q$ , while in Figure 6.2c,  $\mathcal{C}$  is the set of points below the decreasing step-function. As long as the system is in  $\mathcal{C}$ , the heater stays off, but when the stopping set  $\mathcal{D}$  is hit, the heater switches on. Note that this analysis makes sense only when  $B(\bar{x}) \geq 1$ , because when  $B(\bar{x}) = 0$ , the continuation set is not reachable from state  $(\bar{x}, 0)$ . In that case, we should use the results from the  $\alpha\bar{x}$ -policy. For the numerical evaluation, we remark in passing that there must be some  $q_{\max}$ , perhaps large, at which it is optimal to switch on for any  $x = 0, \dots, \bar{x} - \delta$ .



Suppose that we know the expected time  $N(x, q)$  spent in state  $(x, q) \in \mathcal{C}$  when the process starts in  $(\bar{x}, 0)$  and the heater just switched off. Suppose, furthermore, that we have the probability  $P(x, q)$  of being absorbed in  $(x, q) \in \mathcal{D}$ . The expected cycle time under a  $B$ -policy is then

$$T_1(B) = \sum_{(x,q) \in \mathcal{C}} N(x, q) + \sum_{(x,q) \in \mathcal{D}} P(x, q)T_2(x, q),$$

which is simply the time until reaching  $\mathcal{D}$  plus the time spent on heating/clearing after switching on. Similarly, the expected cycle cost is

$$V_1(B) = \sum_{(x,q) \in \mathcal{C}} pqN(x, q) + \sum_{(x,q) \in \mathcal{D}} P(q, x)V_2(x, q).$$

As before, we define the average cost as  $J(B) = V_1(B)/T_1(B)$ . For the computation of  $N$ , observe first that the time spent in state  $(\bar{x}, 0)$  is

$$N(\bar{x}, 0) = \frac{1}{\lambda + \alpha\bar{x}/\delta}.$$

The other values for  $N(x, q)$ ,  $(x, q) \in \mathcal{C} \setminus \{(\bar{x}, 0)\}$ , can be obtained recursively by using the ‘rate in = rate out’ principle. Applying this at state  $(x, q)$  gives straightaway

$$\left(\lambda + \frac{\alpha x}{\delta}\right) N(x, q) = \lambda N(x, q - 1) + \frac{\alpha(x + \delta)}{\delta} N(x + \delta, q).$$

For this to be properly defined everywhere, we take  $N(x, -1) = 0$  for all  $x$ ,  $N(\bar{x} + \delta, q) = 0$  for all  $q$ , and  $N(x, q) = 0$  for all  $(x, q) \in \mathcal{D}$ .

It remains to determine the probability of being absorbed in state  $(x, q) \in \mathcal{D}$ . However, this is straightforward when we know  $N(x, q)$ . Since jobs come in one-by-one, we only have to deal with points on the boundary of  $\mathcal{D}$  and  $\mathcal{C}$ , that is, points such that  $(x, q) \in \mathcal{D}$  and  $(x, q - 1) \in \mathcal{C}$ . For such points, then,

$$P(x, q) = \lambda N(x, q - 1),$$

since  $\mathcal{D}$  can only be entered by a job arrival.

*Remark 6.2.* The above expression for  $P(x, q)$  cannot be used to evaluate the costs of the  $X$ -policy, as for such a policy, the stopping set is entered by a decrease in temperature, not by a job arrival. However, it is straightforward to analyze  $X$ -policies using a similar approach.

**Algorithm 4** Local improvement heuristic

---

Set  $B'_0(0) = \bar{x} + \delta$  and  $B'_0(q) = 0$  for  $q = 1, \dots, q_{\max}$ .
 $i = 0$ **repeat** $i = i + 1$  $B'_i = B'_{i-1}$ **for**  $q = 1, \dots, q_{\max} - 1$  **do** $x^* = \arg \min_{B'_i(q+1) \leq x \leq B'_i(q-1)} J(f_q(B'_i, x)).$  $B'_i(q) = x^*$ **end for****until**  $B'_i = B'_{i-1}$  **or**  $i \geq i_{\max}$ **6.4.4 Improvement heuristic for  $B$ -policies**

We propose a simple local improvement heuristic to obtain effective  $B$ -policies. The heuristic is easiest expressed in terms of the queue, so we first carry out a policy transformation. We define  $B'(q)$  be the threshold temperature when the queue is  $q$ , so that the heater switches on when  $x \geq B'(q)$ . In order to start in the continuation set and to reach the stopping set with probability 1, we require that  $B'(0) > \bar{x}$  and  $B'(q_{\max}) = 0$  for some large  $q_{\max}$ . Since  $B'(q)$  decreases in  $q$ , we can uniquely transform a policy  $B'$  into an equivalent policy  $B$  using  $B(x) = \arg \min_q \{B'(q) \leq x\}$ . Therefore, the costs  $J(B')$  of policy  $B'$  follow by first transforming it and then using the costs from Section 6.4.3.

The pseudocode in Algorithm 4 conveys the main idea of the heuristic. As starting policy  $B'_0$ , we use the  $Q = 1$  policy. The heuristic iteratively improves  $B'$  until either a local optimum is reached or the maximum number of iterations  $i_{\max}$  is exceeded. The function  $f_q(B', x)$  gives the policy where the  $q$ -th threshold of  $B'$  is changed to  $x$ . During each iteration  $i$ , we sequentially update  $B'_i(q)$  for each  $q$  to the temperature  $x^*$  that minimizes costs, given the current values for all other thresholds. The reason to search between  $B'_i(q + 1)$  and  $B'_i(q - 1)$  is to ensure  $B'_i$  stays decreasing. We need to carry out several iterations, because when  $B'_i(q)$  is changed for some  $q > q'$ , the current value for  $B'_i(q')$  need no longer be cost-minimizing.

*Remark 6.3.* Note that  $q$  is an integer. When  $\delta$  is sufficiently small, there are typically multiple values of  $x$  for which  $B(x) = q$  and mostly one value of  $q$  for which  $B'(q) = x$ . Thus, by formulating the heuristic in terms of  $q$ , we set the correct thresholds for multiple values of  $x$  simultaneously.

*Remark 6.4.* In our experiments, we determine  $x^*$  for each  $q$  by enumerating the costs of all policies in the specified range. This requires a considerable number of policy cost evaluations, especially if  $\delta$  is small. A faster alternative is using bisection search in the specified range to obtain  $x^*$  heuristically.

### 6.4.5 Markov decision process formulation

To study the optimal policy structure, we formulate a Markov decision process (MDP) under some simplifying assumptions. We apply a similar approximation as for  $B$ -policies in Section 6.4.3, i.e., we discretize time into intervals of  $\delta$  and let the time between temperature changes be exponentially distributed. Furthermore, we let the processing times of jobs also be exponentially distributed.

We allow the heating intensity  $u$  to be changed in any system state, but based on the insights obtained from the fluid queue, we limit the possible actions to  $u \in \{0, \alpha x, \beta\}$ . Hence, the heater is either off ( $u = 0$ ), maintains the current temperature ( $u = \alpha x$ ), or heats at maximum intensity ( $u = \beta$ ). Clearly, this MDP contains all wait-heat-clear policies. Hence, the minimal average cost for this MDP is at least as low as that of the best  $B$ -policy, provided the processing times are exponential in both cases. We remark that this neglects minor numerical differences because the heating time in the MDP is discretized, while it is exact in the  $B$ -policy.

We apply uniformization to convert the continuous-time MDP to an equivalent discrete-time MDP. Hence, we convert transition rates to probabilities by dividing by the fastest transition rate and we include self-transitions so that the total transition probability of each state-action pair is 1. Since the fastest transition rate out of any state-action pair is

$$K = \lambda + \max(\mu + \alpha \bar{x}/\delta, \beta/\delta),$$

the transition probabilities for each state-action pair are given by

$$\begin{aligned} P_u(x, q; x, q+1) &= \frac{\lambda}{K} \mathbb{I}_{q < q_{\max}}, & \text{for } u \in \{0, \alpha x, \beta\}, \\ P_u(x, q; x, q-1) &= \frac{\mu}{K} \mathbb{I}_{x=\bar{x}} \mathbb{I}_{q > 0}, & \text{for } u \in \{\alpha x, \beta\}, \\ P_u(x, q; x-\delta, q) &= \frac{\alpha x}{K\delta}, & \text{for } u = 0, \\ P_u(x, q; x+\delta, q) &= \frac{\beta - \alpha x}{K\delta} \mathbb{I}_{x < \bar{x}}, & \text{for } u = \beta. \end{aligned}$$

Here, jobs arrive with the same probability for all actions, up to a maximum of  $q_{\max}$  jobs in total. Jobs depart the system only when  $x = \bar{x}$ ,  $q > 0$ , and  $u \neq 0$ . The temperature can decrease when  $u = 0$  and increase when  $u = \beta$ , to at most  $\bar{x}$ . Since temperature is discretized by  $\delta$ , we need to divide the rates by  $\delta$  to let the temperature change at the correct rate. For each state-action pair, the self-transition probability  $P_u(x, q; x, q)$  contains the remaining probability.

The costs for action  $u$  in state  $(x, q)$  are  $C_u(x, q) = pq + cu$ . We use policy iteration to find average cost optimal policy, see Tijms (1994).

Table 6.1: *Overview of the considered policies*

Policy	Description	Reference
$\alpha\bar{x}$	Always maintain the production temperature	Eq. (6.8)
$B$	Wait-heat-clear policy with a threshold involving both queue length and temperature	Section 6.4.4
$B_{\text{mdp}}$	Optimal policy from the MDP	Section 6.4.5
$Q$	Wait-heat-clear policy with a queue length threshold	Lemma 6.7
$Q_f$	Approximate $Q$ by using the optimal parameter of the fluid queue approximation	Section 6.3.3
$Q_m$	Approximate $Q$ by replacing the expectation by its mean	Remark 6.1
$X$	Wait-heat-clear policy with a temperature threshold	Lemma 6.8

## 6.5 Numerical Results

In this section, we provide numerical insights into the performance of the various policies developed in this chapter. The policies are summarized in Table 6.1 for quick reference; details on the numerical implementation are given in Appendix 6.B.

We focus mainly on the  $Q$ -policy since it is simple to execute in practice and it turns out to be effective in many instances. In Section 6.5.1, we shortly discuss the costs savings that can be obtained in a real-life case by using a  $Q$ -policy. In Section 6.5.2, we study optimal policies in two problem instances to illustrate the impact of state-dependent setup times and to explain when the  $Q$ -policy performs well and when not. Finally, in Section 6.5.3, we carry out a full-factorial experiment in order to obtain statistical insights into the effect of parameters on the performance of all policies.

### 6.5.1 Real-life case

As starting point for our numerical analysis, we take the real-life case of a circuit board manufacturer at which some of our students did an internship. The company currently uses a policy resembling the  $\alpha\bar{x}$ -policy: the heater is switched on in the morning, irrespective of the presence of jobs, and switched off at the end of the working day. The manufacturer estimates its yearly energy expenses for the heat bath at around €50 000. Based on estimates for the parameter values, which we derive in 6.C, we find that a  $Q$ -policy with  $Q = 20$  has the potential to save around €15 000 yearly compared to the  $\alpha\bar{x}$ -policy—roughly one third of a yearly employee salary. However, using a threshold of  $Q = 20$  may be too long, as the daily order demand is  $\lambda = 5$ . Setting  $Q = 5$  so that the bath switches on once a day, on average, already results in a cost saving of about €10 000. Thus, simple wait-heat-clear policies can lead to considerable cost savings.

### 6.5.2 Optimal policy structure

To provide insights into the optimal policy structure, we examine two particular problem instances that are solved with the Markov decision process formulation discussed in Section 6.4.5. Both instances have a low server load ( $\rho = 0.1$ ) and share the same parameters except for the heat transfer coefficient and the energy costs. We select these two particular instances from our full-factorial experiment because for these instances the  $Q$ -policy achieves its best and worst performance.

Figure 6.3 shows results for the instance where  $Q$ -policy achieves its best performance. Here, the heat transfer coefficient and energy costs are high, so that the heat bath cools down fast and heating up is expensive. The left graph shows the structure of the optimal policy, which consists of three regions: in the black area the heater is switched off, in the white area the heater is switched on at maximum power, and in the gray area (on the right border) the production temperature is maintained if the queue is non-empty. Thus, the optimal policy is a  $B$ -policy. The border of the black and white area defines the threshold of the  $B$ -policy, and it is clear that this threshold decreases in the temperature. Most notably, the longer the system cools down, the longer the queue should be before the heater is switched on. This is a consequence of the fact that the setup costs (energy costs) to heat up the server are higher for lower temperatures. We are only willing to incur these setup costs if the number of jobs to be processed is large enough, similar to what we observe for the order quantity in an economic order quantity model when the fixed order costs increase, see e.g., Tijms (1994, Section 1.5.1).

The right graph in Figure 6.3 shows a heat map of the stationary distribution of the temperature-queue process  $\{(x(t), q(t))\}$  under the optimal policy. Dark states are visited most often, while white states are essentially never visited. Each cycle of the process starts in the lower-right corner state. When comparing the stationary distribution with the threshold in the right graph, we see that when the heater is off, the process is mostly in states far below the threshold. Because the process remains below the threshold, in almost all cases it drifts to the left boundary  $x = 0$ . At this boundary, we observe that the heating phase starts once the queue length becomes 44. The dark horizontal line indicates that the heating phase finishes quite fast; typically between 0 and 2 jobs arrive during heating. After heating, the process remains at the right boundary  $x = 100$  until the queue is cleared. Since heating almost always starts in state  $x = 0$  and  $q = 44$ , the optimal policy is practically identical to a  $Q$ -policy with  $Q = 44$ , explaining why the  $Q$ -policy is near-optimal in this instance.

On the other hand, the instance in Figure 6.4 has a low heat transfer coefficient and energy cost. Since the heat bath cools down slowly and setups are relatively inexpensive, we can see that threshold levels in Figure 6.4 are much lower than in Figure 6.3. Comparing the left and right graph in Figure 6.4, we see that now relatively

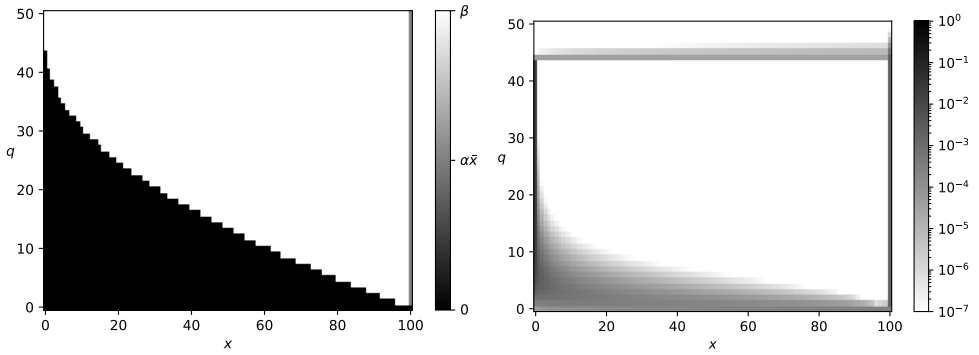


Figure 6.3: *Optimal policy (left) and heat map of its stationary distribution (right). Parameters:  $\lambda = 1$ ,  $\mu = 10$ ,  $c_S^2 = 1$ ,  $\bar{x} = 100$ ,  $\alpha = 0.7$ ,  $\beta = 1000$ ,  $p = 1$ ,  $c = 10$ .*

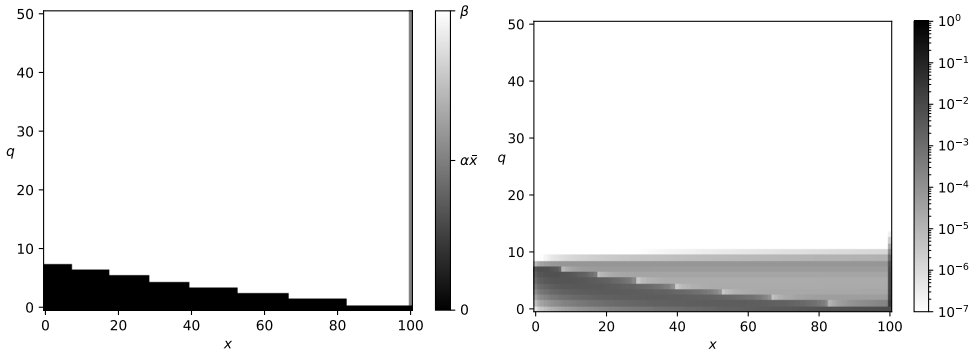


Figure 6.4: *Optimal policy (left) and heat map of its stationary distribution (right). Parameters:  $\lambda = 1$ ,  $\mu = 10$ ,  $c_S^2 = 1$ ,  $\bar{x} = 100$ ,  $\alpha = 0.3$ ,  $\beta = 1000$ ,  $p = 1$ ,  $c = 0.5$ .*

much time is spent in states around the threshold for any temperature. This implies that the temperature-dependency of the setup time plays a very important role in this instance. The  $B$ -policy is able to exploit the fact that the heating time is short when the temperature is still high. In contrast with this, since the  $Q$ -policy neglects the temperature in its decision, it cannot adequately react to events where more jobs than expected arrive after the heater has been switched off. However, even in these unfavorable conditions, the costs of the optimal  $Q$ -policy are only 8.27% higher than the optimal  $B$ -policy.

### 6.5.3 Full-factorial experiment

In order to obtain further insights, we want to answer the following questions by carrying out a full-factorial experiment. First, how do the policies in Table 6.1 perform compared to the optimal policy? Second, in which types of instances is the always-on policy optimal? Third, which parameters have the largest effect on the performance of the  $Q$ -policies? And, finally, how can we easily determine an effective queue length threshold for the  $Q$ -policy? The answers to these questions help to provide simple guidelines for determining when and when not to use a certain policy, which is also of importance for our real-life case, as there is considerable variation in the values of the parameters, for instance in average monthly demand, electricity prices, and so on.

In the full-factorial experiment, we vary various parameters from very small to very large, with values as shown in Table 6.2. In all instances, we scale the queueing cost to  $p = 1$ . Furthermore, we set the squared coefficient of variation at  $c_S^2 = 1$ , so that the policies can be compared with the optimal policy from the MDP. All in all, this results in  $5^6$  instances in total. We remark that  $\mu$  follows from  $\mu = \lambda/\rho$  and  $\beta$  from  $\beta = r\bar{x}$ .

As a first major point, we observe from the experiment that for every individual instance the costs are ordered as

$$J^*(B) \leq J^*(B_{\text{mdp}}) \leq J^*(Q) \leq J^*(X) \leq J(\alpha\bar{x}),$$

where  $J^*$  is the cost of the optimal policy in the specified policy class. This shows that the best  $Q$ -policy performs at least as good as the best  $X$ -policy. The inequalities also show the evident fact that a policy with a threshold based on queue and temperature, i.e., a  $B$ -policy, is better than policies that depend on a single threshold.

A second major point is that, in every instance, the optimal  $B_{\text{mdp}}$ -policy from the MDP satisfies the wait-heat-clear policy structure. This provides numerical support for the optimality of wait-heat-clear policies. Furthermore, we find that the costs of the  $B$ -policy are essentially equal to those of the  $B_{\text{mdp}}$ -policy in every instance, hence our improvement heuristic seems to return optimal solutions.

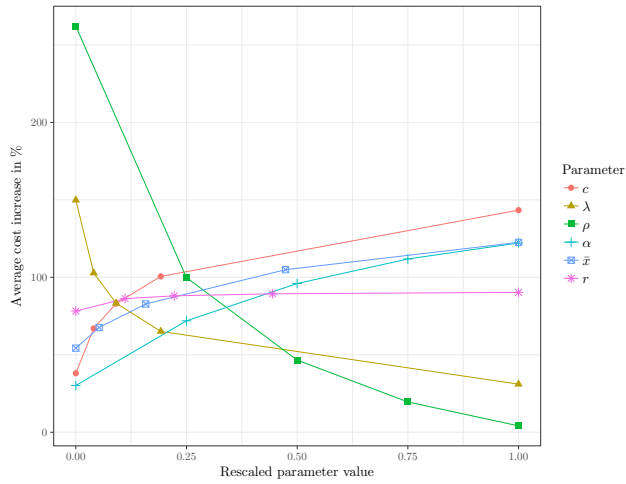


Figure 6.5: Percentage cost increase from using the  $\alpha\bar{x}$ -policy compared to the  $B$ -policy

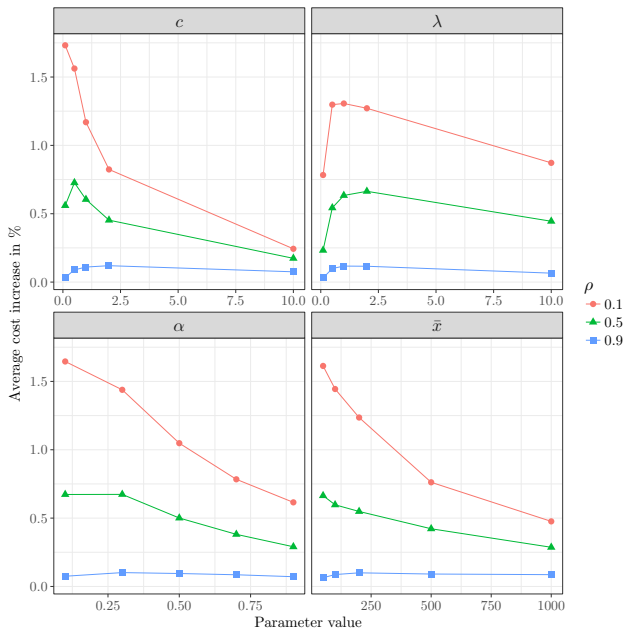


Figure 6.6: Percentage cost increase from using the  $Q$ -policy compared to the  $B$ -policy



Table 6.2: *Parameter values in the full-factorial experiment.*

Parameter	Symbol	Very low	Low	Medium	High	Very high
Energy cost	$c$	0.1	0.5	1	2	10
Arrival rate	$\lambda$	0.1	0.5	1	2	10
Server load	$\rho$	0.1	0.3	0.5	0.7	0.9
Heat transfer coefficient	$\alpha$	0.1	0.3	0.5	0.7	0.9
Production temperature	$\bar{x}$	50	100	200	500	1000
Heating power ratio	$r = \beta/\bar{x}$	1	2	3	5	10

Table 6.3: *Summary statistics of the relative cost increase compared to the B-policy.*

	Average	Min.	1st Quartile	Median	3rd Quartile	Max.
$\alpha\bar{x}$	86.43	0.00	5.32	29.88	92.63	854.37
$X$	11.19	0.00	1.06	3.60	10.86	230.89
$Q$	0.55	0.00	0.00	0.07	0.49	8.27
$Q_f$	0.59	0.00	0.00	0.07	0.52	11.29
$Q_m$	0.60	0.00	0.00	0.07	0.53	11.29
$B_{mdp}$	0.00	0.00	0.00	0.00	0.00	0.06

To get a better idea of the effectiveness of the different policies, Table 6.3 shows summary statistics for the percentage increase in costs of the policies compared to the  $B$ -policy. Here, we see that  $Q$ -policies perform in general much better than  $X$ -policies, and  $\alpha\bar{x}$ -policies should be used cautiously as they can be (very) costly. In most instances, the  $Q$ -policy is within 1% of the  $B$ -policy, showing that a rather simple policy based only on the queue length is remarkably effective.

The results for the  $\alpha\bar{x}$ -policy indicate that the cost of never turning off the heater can vary from excellent to extremely poor. In order to understand this in more detail, Figure 6.5 depicts, for each parameter value in the experiment, the average percentage increase in costs of the  $\alpha\bar{x}$ -policy compared to  $B$ -policy. In order to show all effects in the same graph, we rescaled all parameters from Table 6.2 between 0 (very low) and 1 (very high). As could be expected, the  $\alpha\bar{x}$ -policy performs comparatively best when the system is busy (large values of  $\rho$  and  $\lambda$ ) and when heating is inexpensive (low values of  $\alpha$ ,  $\bar{x}$ , and  $c$ ). The parameter  $\rho$  has the strongest impact: the  $\alpha\bar{x}$ -policy is expensive when  $\rho$  is low (about 300% higher than optimal), but it is reasonable when  $\rho$  is high. On the other hand, the heating power ratio  $r$  seems to have almost no impact.

Next, we want to understand which parameters influence the effectiveness of the  $Q$ -policy. Figure 6.6 shows the percentage increase in cost for the most interesting parameters, with lines shown for several values of  $\rho$ . We observe in all graphs that the

$Q$ -policy becomes better as  $\rho$  increases, because both the  $Q$ - and  $B$ -policies become  $\alpha\bar{x}$ -policies when the server load is high. Interestingly, the effect of the energy cost  $c$  can be decreasing, increasing, or both, depending on the server load. In the graphs for  $\lambda$ , we see that the  $Q$ -policy is relatively worst for intermediate values of  $\lambda$ , because when  $\lambda$  is very low, the heat bath is typically already cooled down when the first customers arrive, and when  $\lambda$  is very high, it is typically optimal to remain at the processing temperature. Combining all graphs, we see the  $Q$ -policy has the largest difference with the  $B$ -policy in instances with low server load and inexpensive heating (very low  $\alpha$ ,  $\bar{x}$  and  $c$ ). These are instances for which it makes most sense to quickly turn on the heater and clear all jobs when more jobs than expected arrive. However, even in such instances the  $Q$ -policy is well within 10% from the optimal policy.

Finally, given that  $Q$ -policies perform well, it is important for practice to have a simple method to determine effective thresholds. In principle, it is not difficult to calculate the average cost of the  $Q$ -policy, but it does involve taking an expectation. A simpler alternative is to use an approximation with a closed-form expression for the average cost, as is the case for the  $Q_f$ - and  $Q_m$ -policies. These approximations are easy to implement in spreadsheet software and optimal parameters can be found by inspecting a graph. Interestingly, the costs of  $Q_f$ - and  $Q_m$ -policies are equal to those of the  $Q$ -policy in nearly every instance. Therefore, we can use these simple approximations to obtain satisfactory policy parameters.

*Remark 6.5.* To study the impact of variance in the service time distribution, we repeated the full-factorial experiment with  $c_S^2 = 0.5$  and  $c_S^2 = 2$ . Although the average costs increase in  $c_S^2$ , we find that variance has a negligible effect on optimal policy parameters and the relative performance of policies. This result is intuitive because a similar insensitivity holds for the closely related  $M/G/1$  queue without setups, where the optimal  $Q$ -policy does not depend on  $c_S^2$  (Tijms, 1994, Example 1.3.4).

## 6.6 Conclusion

We considered a production system with a server that requires heating in order to process jobs. Considerable cost savings may be realized by temporarily switching off the heater and queuing up arriving jobs until sufficient work is available. We model the production system as an  $M/G/1$  queue with a temperature controlled server that can only process jobs if a minimum production temperature is satisfied. The time and energy required to heat a server depend on the current temperature of the server, hence the setup times and setup costs are state dependent. Our main contribution is optimizing the trade-off between energy costs and queuing costs, while accounting for state-dependent setup costs and times.

Analytical results are derived for a fluid queue approximation of the production system. We show that the optimal control policy satisfies a *wait-heat-clear* structure, that is, first the heater is off and new jobs wait in a queue, then the server heats at the maximum rate, and once the minimum production temperature is reached, all jobs are served until the queue is cleared. For the approximation, it is straightforward to numerically obtain the optimal queue length to start heating by using closed-form expressions for the average cost.

A numerical analysis, based on a Markov decision process formulation of the system, suggests that the optimal policy in the stochastic case also satisfies the wait-heat-clear structure. The optimal policy leads to considerable cost savings (on average over 40%) compared to the situation where the server always stays at the minimum production temperature. Based on this and on the insights provided by the fluid queue approximation, we analyze various intuitive wait-heat-clear policies for the  $M/G/1$  queue. The  $Q$ -policy, that starts heating at a given queue length while neglecting temperature, has near-optimal performance in most cases. However, in several cases it is important to base the heating decision on both the queue length and the temperature, in particular when the server load is low and when the server cools down slowly.

For practice we recommend to use a  $Q$ -policy since it is simple to execute and effective in many cases. Furthermore, a near-optimal threshold for the  $Q$ -policy can be obtained straightforwardly from the fluid queue approximation of the system.

Further research could extend the analysis into two directions. First, one can allow for batch arrivals by studying the  $M^X/M/1$  queue with a temperature controlled server. This is relevant for systems where the number of items in a job have a large variability. Second, one can study the  $M/G/1$  queue where the control of the heater depends on the waiting time rather than the queue length. For both extensions, it is interesting to prove that wait-heat-clear policies are optimal, and to clarify how the optimal policy can be efficiently computed.

## Appendix

### 6.A Proofs of lemmas

**Lemma 6.1.** *Suppose, without loss of generality, that the process starts at time  $t = 0$  in state  $x(0) \leq \bar{x}$ , and  $t_2$  is so large that  $\bar{x}$  can be reached, i.e.,  $t_2 \geq \ell(x(0))$  with  $\ell$  given by (6.2). Then the optimal control to reach temperature  $x(t_2) = \bar{x}$  at  $t_2$  is given by*

$$u^*(t) = \begin{cases} 0 & \text{for } 0 \leq t < t_1, \\ \beta & \text{for } t_1 \leq t \leq t_2, \end{cases}$$

and the time  $t_1$  to switch on can be obtained from solving  $t_1 = t_2 - \ell(x(t_1))$ .

*Proof.* We can obtain the policy until time  $t_2$  by solving the following optimal control problem

$$\begin{aligned} & \text{maximize } \int_0^{t_2} -cu(t) dt \\ & \text{subject to } x(0) = x_0, \\ & \quad x(t_2) = \bar{x}, \\ & \quad \dot{x} = u - \alpha x, \\ & \quad u \in [0, \beta]. \end{aligned}$$

This is a fixed-end-point problem, so we follow the approach from Sethi and Thompson (2000, Ch. 3). The Hamiltonian  $H = (\lambda - c)u - \lambda\alpha x$  is maximized by a control policy of the type  $u = \text{bang}[0, \beta; \lambda - c]$ . The adjoint should satisfy

$$\dot{\lambda} = -H_x = \lambda\alpha.$$

The transversality condition is  $\lambda(t_2) = c_1$ , where  $c_1$  is a constant to be determined. The solution of this differential equation is

$$\lambda(t) = c_1 e^{-\alpha(t_2 - t)},$$

which is positive and strictly increasing if  $c_1 > 0$ . Provided  $c_1 > c$ , there exists a  $t_1$  so that  $\lambda(t) < c$  for all  $t < t_1$  and  $\lambda(t) \geq c$  for all  $t_1 \leq t \leq t_2$ . Hence, the optimal control is  $u(t) = 0$  for  $0 \leq t < t_1$  and  $u(t) = \beta$  for  $t_1 \leq t \leq t_2$ . Using this optimal control in the differential equation with boundary condition  $x(t_2) = \bar{x}$ , we obtain

$$t_1 = t_2 + \frac{1}{\alpha} \log \left( 1 - \frac{\alpha \bar{x}}{\beta} (1 - e^{-\alpha t_2}) \right).$$

The value of  $c_1$  then follows from  $\lambda(t_1) = c$ , i.e.,

$$c_1 = \frac{c\beta}{\beta - \alpha\bar{x} + \alpha x_0 e^{-\alpha t_2}} > c.$$

Hence, we have obtained an optimal control that satisfies all conditions. □

**Lemma 6.2.** *During the clearing phase  $t \in [t_2, t_3]$ , the optimal heating policy is*

$$u^*(t) = \alpha\bar{x}.$$

*Proof.* Starting at  $x(t_2) = \bar{x}$ , we solve the following optimal control problem between  $t_2$  and  $t_3$ .

$$\begin{aligned} & \text{maximize } \int_{t_2}^{t_3} -cu(t) dt \\ & \text{subject to } x \geq \bar{x}, \\ & \quad x(t_2) = \bar{x}, \\ & \quad \dot{x} = u - \alpha x, \\ & \quad u \in [0, \beta], \end{aligned}$$

The pure state constraint  $x \geq \bar{x}$  ensures that processing continues. The Hamiltonian is

$$H = (\lambda - c)u - \lambda\alpha x$$

which implies the optimal control to be

$$u^* = \begin{cases} \text{bang}[0, \beta, \lambda - c], & \text{if } x > \bar{x}, \\ \text{bang}[\alpha\bar{x}, \beta, \lambda - c], & \text{if } x = \bar{x}. \end{cases}$$

Because of the constraint  $x \geq \bar{x}$ , we need to set  $u^* \geq \alpha\bar{x}$  when  $x = \bar{x}$ .

As we have mixed inequality constraints and a pure state inequality constraint, we follow Sethi and Thompson (2000, Ch.4). The mixed inequality constraints are  $g_1(x, u, t) = u \geq 0$ ,  $g_2(x, u, t) = \beta - u \geq 0$ . The pure state inequality constraint is  $h(x, t) = x - \bar{x} \geq 0$  with  $h^1(x, u, t) = u - \alpha x$ . Writing this in Lagrangian form gives

$$L = -cu + \lambda(u - \alpha x) + \mu_1 u + \mu_2(\beta - u) + \eta(u - \alpha x),$$

with complementary slackness conditions

$$\begin{aligned} \mu_1 &\geq 0, & \mu_1 u &= 0, \\ \mu_2 &\geq 0, & \mu_2(\beta - u) &= 0, \\ \eta &\geq 0, & \eta(x - \bar{x}) &= 0, & \dot{\eta} &\leq 0. \end{aligned}$$

The adjoint has to satisfy

$$\dot{\lambda} = -L_x = \alpha(\lambda + \eta)$$

and  $\lambda(t_3) = 0$ . Solving this gives

$$\lambda(t) = \eta(e^{-\alpha(t_3-t)} - 1).$$

Clearly,  $\lambda(t) \leq 0$  for all  $t \in [t_2, t_3]$ . This implies that the optimal policy is  $u(t) = 0$  when  $x > \bar{x}$  and  $u(t) = \alpha\bar{x}$  when  $x = \bar{x}$ . With such a control policy it is evident that when starting at  $x(t_2) = \bar{x}$  we will have  $x(t) = \bar{x}$  for all  $t \in [t_2, t_3]$ .  $\square$

**Lemma 6.5.** *Starting at  $(x, q)$  with  $x \leq \bar{x}$ , the expected time and cost for heating and clearing until the end of the cycle is*

$$\begin{aligned} T_2(x, q) &= \ell(x) + T_3(q + \lambda\ell(x)) = \frac{q}{\mu - \lambda} + \frac{\mu}{\mu - \lambda}\ell(x), \\ V_2(x, q) &= V_3(q) + \frac{p\lambda\mu}{\mu - \lambda}\ell(x)^2 + \frac{p\mu}{\mu - \lambda}\ell(x)q \\ &\quad + \left( \frac{p\lambda + c\alpha\bar{x}\lambda}{\mu - \lambda} + p\frac{\lambda^2(1 + c_S^2)}{2(\mu - \lambda)^2} + c\beta \right)\ell(x). \end{aligned}$$

Here,  $\ell(x)$  is given by (6.2), and  $V_3$  and  $T_3$  are given by Lemma 6.4. All involved constants are positive, hence  $V_2(x, q)$  is a quadratic, strictly increasing function of  $q$  and  $\ell(x)$ , and  $T_2(x, q)$  is linear.

*Proof.* Write, for convenience,  $\ell = \ell(x)$ . The number of arrivals during heating,  $N = N(\ell)$  is Poisson distributed with parameter  $\lambda\ell$ . This implies that for  $N$  given, the job arrival epochs are uniformly distributed over the time interval  $[0, h]$ . Thus, since we start with a queue length  $q$ , the expected queueing costs during heating become

$$p\mathbb{E} \left[ \int_0^\ell q(t) dt \mid N \right] = p\ell \left( q + \frac{N}{2} \right).$$

Taking the expectation and using that  $\mathbb{E}[N] = \lambda\ell$ , we obtain for the total expected

cost during  $[0, \ell]$ ,

$$(pq + c\beta)\ell + p\frac{\lambda}{2}\ell^2.$$

Next, consider the expected cost to clear the queue. Note from Lemma 6.4 that  $V_3(q)$  has the form  $V_3(q) = aq^2 + bq$ , with

$$a = \frac{p}{2(\mu - \lambda)}, \quad b = p\frac{\lambda c_s^2 + \mu}{2(\mu - \lambda)^2} + c\frac{\alpha\bar{x}}{\mu - \lambda}.$$

We obtain

$$\begin{aligned} \mathbb{E}[V_3(q + N)] &= a\mathbb{E}[(q + N)^2] + b\mathbb{E}[q + N] \\ &= a(q^2 + 2q\lambda\ell + \mathbb{E}[N^2]) + b(q + \lambda\ell) \\ &= V_3(q) + a\lambda^2\ell^2 + (a + 2aq + b)\lambda\ell, \end{aligned}$$

since  $\mathbb{E}[N^2] = \lambda^2\ell^2 + \lambda\ell$ . Adding both cost components gives  $V_2(x, q)$  after some algebra.  $\square$

## 6.B Implementation details

In our computations, we restrict all policy parameters to integers. For a fair comparison, we use the approximation from Section 6.4.3 to evaluate the time and cost of all policies. Specifically, the time and cost during the waiting phase are approximated using exponentially distributed temperature decreases (step size  $\delta = 1$ ), while the time and cost during heating and clearing are calculated exactly using the expressions in Section 6.4.1. The costs of the  $B_{\text{mdp}}$ -policy from the MDP – a  $B$ -policy in every instance – have been recomputed using the costs expressions for  $B$ -policies. All policies have the  $\alpha\bar{x}$ -policy as special case, e.g., set  $Q = 0$  or  $X = \bar{x}$ , however, the average cost cannot be evaluated using the approximation since the cycle time is 0. Therefore, in such cases we use the exact average cost  $J(\alpha\bar{x})$  of the  $\alpha\bar{x}$ -policy.

For the different policies, we limited the maximum queue length  $q_{\text{max}}$  in the following ways. For the  $Q$ -,  $Q_f$ -,  $Q_m$ -, and  $B$ -policies, we set  $q_{\text{max}} = 1000$ , which by far exceeded the optimal parameters found in all instances. For the  $X$ -policy, we need to make sure that all arriving customers can still enter the queue before the temperature drops to 0. Therefore, we set  $q_{\text{max}}$  as the 99.9999% quantile of the Poisson number of arrivals during twice the expected time to cool down to 0 (assuming exponentially distributed temperature decreases with step size  $\delta = 1$ ). For the MDP, it is possible to never use the heater and incur cost  $pq_{\text{max}}$  each time unit. Therefore, we set  $q_{\text{max}} = \max(1000, \lceil J(\alpha\bar{x})/p \rceil)$ , ensuring that at least the  $\alpha\bar{x}$ -policy is better than

never heating. Finally, for policy iteration we need an initial policy. Experimentally, we found that a good initial policy is a  $B$ -policy with  $B(0) = Q_f$  and  $B(\bar{x}) = 1$ , with the thresholds  $B(x)$  for all intermediate temperatures  $0 < x < \bar{x}$  based on linear interpolation between these two values and rounded up to the nearest integer.

We implemented all policies in Python and compiled performance-intensive parts in C using the Numba package. We used a computing cluster to solve all instances of the MDP, because the computation time per instance can be up to 24 hours in the worst case. In comparison, the  $B$ -policy has an average solution time of approximately 20 seconds per instance. All other policies have negligibly small computation times.

## 6.C Real-life case parameter estimates

We measure all parameters in time units of a day. The manufacturer receives about 1000 orders a year and the shop floor is open for 200 days a year, hence  $\lambda = 5/d$ . A job contains on average about 50 items, and each item spends about 1 minute in the bath. Typically an operator requires an additional amount of 10 minutes to position the carriers for the items and some other activities. As job sizes vary quite considerably, we model job service times as exponential with a mean duration of 1 per hour. A working day contains typically 10 hours so that  $\mu = 10$ , from which it follows that  $\rho = 1/2$ .

The working temperature of the bath is slightly above the melting temperature of tin; we take  $\bar{x} = 250^\circ\text{C}$ . The bath switches off at 6 pm, and switches on at 8 am. After cooling down for 14 hours—neglecting weekends—the bath is about  $100^\circ\text{C}$ , which is still somewhat higher than the room temperature  $\theta = 20^\circ\text{C}$ . By solving for  $\alpha$  in  $x(t_1) = (\bar{x} - \theta)e^{-\alpha t_1} = 100$  with  $t_1 = 14$ , we find that  $\alpha = 1.4$ . Furthermore, since it takes about 3 hours to heat up the bath, we solve for  $\beta$  in  $\ell(100) = 3$  in (6.2) to obtain  $\beta = 1450^\circ\text{C}/d$ .

Currently the company keeps the heater on the entire day, corresponding to the  $\alpha\bar{x}$ -policy. The yearly heating cost is about  $e50,000$ , hence the heating costs per day are  $c\alpha\bar{x} = 50,000/200 = e250$ . As  $\alpha$  and  $\bar{x}$  are known from the above,  $c$  follows.

Finally, for the queueing cost  $p$ , as the yearly revenue is around  $e20\text{M}$  and the number of orders is 1000, the average order value is  $\text{€}20\text{K}$ . Assuming that half of the value is spent on raw materials and the bank lending rate is 5%,

$$p = \frac{10Ke \cdot 5\%}{200d} = 2.5e/d.$$



*If people do not believe that mathematics is simple,  
it is only because they do not realize how complicated life is*  
John von Neumann