

Control of continuum models of production systems

Michael La Marca* Dieter Armbruster*[†] Michael Herty[‡]
Christian Ringhofer*

August 14, 2008

Abstract

A production system which produces a large number of items in many steps can be modelled as a continuous flow problem. The resulting hyperbolic partial differential equation (PDE) typically is nonlinear and nonlocal, modeling a factory whose cycle time depends nonlinearly on the work in progress. One of the few ways to influence the output of such a factory is by adjusting the start rate in a time dependent manner. We solve two prototypical control problems for this case: i) demand tracking where we determine the start rate that generates an output rate which optimally tracks a given time dependent demand rate and ii) backlog tracking which optimally tracks the cumulative demand. The method is based on the formal adjoint method for constrained optimization, incorporating the hyperbolic PDE as a constraint of a nonlinear optimization problem. We show numerical results on optimal start rate profiles for steps in the demand rate and for periodically varying demand rates and discuss the influence of the nonlinearity of the cycle time on the limits of the reactivity of the production system. Differences between perishable and non-perishable demand (demand vs. backlog tracking) are highlighted.

1 Introduction

Recently Armbruster et al [3] have introduced a continuum model to simulate the average behavior of production systems at an aggregate level. Such a description is appropriate e.g. for a semiconductor fab which produces a large number of items in a large number of steps. The appropriate mathematical variable to describe the production flow in that case is a density variable $\rho(x, t)$ describing the density of products at stage x of the production at a time t . We

*Department of Mathematics and Statistics, Arizona State University Tempe, AZ 85287-1804, USA email:lamarca@mathpost.asu.edu, armbruster@asu.edu, ringhofer@asu.edu

[†]Department of Mechanical Engineering, Eindhoven University of Technology, P.O.Box 513, NL-5600 MB, Eindhoven, The Netherlands

[‡]RWTH Aachen University, D-52056 Aachen, Germany, herty@mathc.rwth-aachen.de

scale $x \in [0, 1]$, $x = 0$ representing the beginning of the production line and $x = 1$ the end.

Assuming mass conservation - a reasonable assumption for a semiconductor fab which does not know without significant testing whether a given wafer is defective -, the time evolution of the product density is described by the continuity equation:

$$\frac{\partial \rho(x, t)}{\partial t} + \frac{\partial}{\partial x} (v(\rho(x, t))\rho(x, t)) = 0 \quad (1)$$

where $v(\rho(x, t))$ is a velocity function that depends on the density $\rho(x, t)$ only. The rate $\lambda(t)$ of products entering and exiting the fab at $x = 0$ and $x = 1$ are defined as

$$\begin{aligned} \lambda(t) &= v(\rho(x, t))\rho(x, t)|_{x=0} \\ \sigma(t) &= v(\rho(x, t))\rho(x, t)|_{x=1} \end{aligned}$$

We choose the velocity $v(\rho, x, t) = v_{EQ}(\rho)$ as a function of the total load in the production line describing the equilibrium velocity of the factory as a whole [3]. We model the equilibrium flow v_{EQ} using the total load (Work in Progress, WIP) $L := \int_0^1 \rho(x, t) dx$. Treating the production line as *one* M/M/1 queue [16, 1] the associated cycle time $\tau = \frac{1}{v_{EQ}}$ for a production rate $\mu = \frac{1}{v_{max}}$ and an average queue length L we obtain $\tau = \mu(1 + L)$ and

$$v(\rho) = \frac{v_{max}}{1 + \int_0^1 \rho(s, t) ds}. \quad (2)$$

Assuming an initial WIP distribution in the factory $\rho_0(x)$, is known, and prescribing the influx, we have a fully determined first order nonlinear hyperbolic PDE for the evolution of the network densities.

$$\frac{\partial \rho(x, t)}{\partial t} + \frac{\partial}{\partial x} (v(\rho)\rho(x, t)) = 0 \quad (3a)$$

$$\lambda(t) = v(\rho)\rho(0, t) \quad (3b)$$

$$\rho_0(x) = \rho(x, 0) \quad (3c)$$

$$v(\rho) = \frac{v_{max}}{1 + \int_0^1 \rho(s, t) ds}. \quad (3d)$$

If the influx $\lambda(t)$ and our initial condition $\rho_0(x)$ are nonnegative, then the density is and will remain nonnegative.

Note that the velocity of the conservative hyperbolic PDE (3a) is constant across the entire system at a given time. This would imply that in a real world factory, all parts move through the factory with the same speed. While in a serial production line, speed through the factory is dependent on all items and machines downstream, in a highly re-entrant factory this is not the case. Since items must visit machines more than once, including machines at the beginning of the production process, their speed through factory is determined by the total number of parts both upstream and downstream from them. Such re-entrant production is characteristic for semiconductor fabs.

Further, the system (3) enjoys the following theoretical properties:

- $v(\rho)$ is positive
- $v(\rho)$ is bounded; $0 < v(\rho) \leq v_{max}$
- $v(\rho)$ has no spatial dependence (it is integrated out)

While the positivity of $v(\rho)$ follows easily from the fact that $v_{max} > 0$, the strict inequality in the second characteristic is not so obvious. The PDE in ρ (3a) is an advection equation, hence whatever is in the system was either there at the beginning or came in through the left boundary condition. Since $\lambda(t) = v(\rho)\rho(0, t)$ is bounded, and assuming that $\rho_0(x)$ is as well, the finite time horizon implies that $\rho(x, t)$ is bounded on $[0, 1] \times [0, \tau]$. Therefore, there exists a $\rho_{max} > 0$ such that

$$0 \leq \rho(x, t) \leq \rho_{max} \text{ for all } (x, t) \in [0, 1] \times [0, \tau]$$

and hence $v(\rho)$ is strictly larger than zero.

The last item is a result of the non-locality of the continuum model and has two important consequences. First, $v(\rho)$ is only time dependent so

$$\frac{\partial}{\partial x} (v(\rho)\rho(x, t)) \equiv v(\rho)\rho_x(x, t)$$

Second, the density propagates with the same speed everywhere in the spatial domain for a given time, which means that no shocks and/or rarefaction waves develop.

1.1 Controlling the continuum model

The ultimate goal of any model for a production system typically is to control it in such a way as to run production in a predetermined way. For a semiconductor fab there are essentially only two ways that influence production rate: The influx and dispatch policies. Dispatch policies are used to control the behavior of the production line on small scales: they act at the individual production stage and they are used to match target production for a short amount of time - typically involving production targets for the next few days or at most a week (see [23]). The only way to influence the output of the whole factory over a longer timescale, e.g. following a seasonal demand pattern or ramping up or down a new product, is to change the influx. However, given that realistic cycle times of a semiconductor factory lie between 30 and 60 days and given that WIP in re-entrant production has a big influence on the speed of a product through a factory (in particular if the factory is run close to capacity) it is far from obvious what kind of starts policy leads to a desired output over a given period of time.

In this paper we will develop an algorithm that controls the outflux of the continuum model solely by regulating the influx. We achieve two distinct yet related goals:

1. Minimize the mismatch between the outflux and a demand rate target over a fixed time period (demand tracking problem).
2. Minimize the mismatch between the total number of parts that have left the factory and the desired total number of parts over a fixed time period (backlog problem).

Mathematically we obtain a control problem subject to hyperbolic PDEs. Whereas the theory of elliptic and parabolic PDEs is well-established [26] far less results are known for hyperbolic problems [6, 28, 27, 4, 13]. We pursue a numerical approach to solve the demand tracking and backlog problem.

We will use a formal approach to the adjoint method of variational optimization. This will allow us to deal with jumps in the initial and boundary values as well as with the nonlocality in the flux function. Alternative approaches to controlling hyperbolic systems [8, 14, 15] are based on classical solutions and due to the nonlocal term in the flux are not applicable to our system. For the same reason the recent results due to Bressan et. al. on optimality conditions [5] are also not directly applicable. Therefore, we focus on the formal derivation and its numerical validation. This approach is an optimize-then-discretize approach to the problem. Other approaches use first a full numerical discretization before proceeding to the optimization problem, e.g. [9]. However, in this way the structural information of the PDE is more difficult to preserve.

The structure of this paper will be as follows. We introduce the demand tracking problem in Section 2 and discuss the adjoint method and its numerical implementation. Results for several demand scenarios will be discussed in Chapter 3. Section 4 will discuss the backlog problem together with some interesting results for different demand experiments. Section 5 will draw conclusions and suggest further areas of investigation. Algebraic details of the derivations of equations for the adjoint method can be found in an appendix 6.1 and 6.2.

2 Demand tracking

Controlling the production rate of a factory or production system is a vital goal in manufacturing: producing too much of an item leads to inventory/holding costs while producing too little leads to lost sales and backlog costs. In order to maximize profitability, a production system must be able to match its projected demand as closely as possible. While demand is stochastic over a given time period, a business typically generates a demand forecast for the next day, week, month, etc. and runs its production system to match this demand accordingly. Although there are numerous ways to generate this demand forecast, the specifics are not important to this research since the only a priori assumption is that a demand rate forecast exists and is in the form of a target outflux. The question of a total item demand forecast will be addressed in the backlog problem.

Assuming costs that penalize both overproduction and underproduction equally and forgetting errors between the outflux and demand target after they

have appeared (the news vendor model [24]), we define a cost functional J by

$$J(\rho, \lambda) := \frac{1}{2} \int_0^\tau (d(t) - v(\rho)\rho(1, t))^2 dt. \quad (4)$$

The problem of minimizing the mismatch between outflux and demand rate target over a fixed time period can be formulated as a mathematical optimization problem. We minimize the cost functional $J(\rho, \lambda)$ subject to the PDE-dynamics introduced previously, i.e., we need to solve

$$\min J(\rho, \lambda) \text{ subject to} \quad (5a)$$

$$\frac{\partial \rho(x, t)}{\partial t} + \frac{\partial}{\partial x} (v(\rho)\rho(x, t)) = 0 \quad (5b)$$

$$\lambda(t) = v(\rho)\rho(0, t) \quad (5c)$$

$$\rho_0(x) = \rho(x, 0) \quad (5d)$$

$$v(\rho) = \frac{v_{max}}{1 + \int_0^1 \rho(s, t) ds}. \quad (5e)$$

In the following we will formally derive first-order necessary conditions for the problem (5). To this end we introduce the Lagrangian function $L(\rho, \lambda, \phi)$ as

$$\begin{aligned} L(\rho, \lambda, \phi) := & \frac{1}{2} \int_0^\tau (d(t) - v(\rho)\rho(1, t))^2 dt \quad (6) \\ & + \int_0^1 \int_0^\tau \phi(x, t) [\rho_t(x, t) + v(\rho)\rho_x(x, t)] dt dx \end{aligned}$$

where ϕ is the multiplier. To find the first-order necessary conditions we take the variations of L with respect to ρ , λ and ϕ , respectively, and enforce them to vanish. The details of *formally* calculating this variational derivative, including the enforcement of the boundary conditions, are contained in Appendix 6.1. We find

$$\rho_t(x, t) + v(\rho)\rho_x(x, t) = 0 \text{ for all } (x, t) \in (0, 1) \times (0, \tau) \quad (7a)$$

$$\lambda(t) = v(\rho)\rho(0, t) \quad (7b)$$

$$\rho_0(x) = \rho(x, 0) \quad (7c)$$

$$v(\rho) = \frac{v_{max}}{1 + \int_0^1 \rho(s, t) ds}, \quad (7d)$$

$$\phi(x, \tau) = 0 \text{ for all } x \in [0, 1] \quad (7e)$$

$$\phi(1, t) = d(t) - v(\rho)\rho(1, t) \text{ for all } t \in [0, \tau] \quad (7f)$$

$$\phi_t(x, t) + v(\rho)\phi_x(x, t) = \quad (7g)$$

$$\frac{v(\rho)^2}{v_{max}} \left[\rho(1, t)d(t) - v(\rho)\rho(1, t)^2 - \int_0^1 \phi(s, t)\rho_x(s, t) ds \right] \quad (7h)$$

$$0 = -\phi(0, t) \quad (7i)$$

The equations have to be solved for (ρ, λ, ϕ) and the conditions are necessary for first-order optimality provided that the functions sufficiently regular. The system (7) enjoys a few interesting properties: As expected the original PDE dynamics (7a–(7d) is part of the optimality system and ensures the feasibility of the solution.

The equations (7f)–(7h) are partial differential equation for ϕ . Given the boundary data (7e) and (7f) the PDE (7g) can be solved for ϕ . The function ϕ is the Lagrange multiplier for the PDE constraints and called adjoint variable. The characteristics for ϕ are depicted below and the equations for ϕ have to be solved backwards in time. The last condition is called optimality condition and

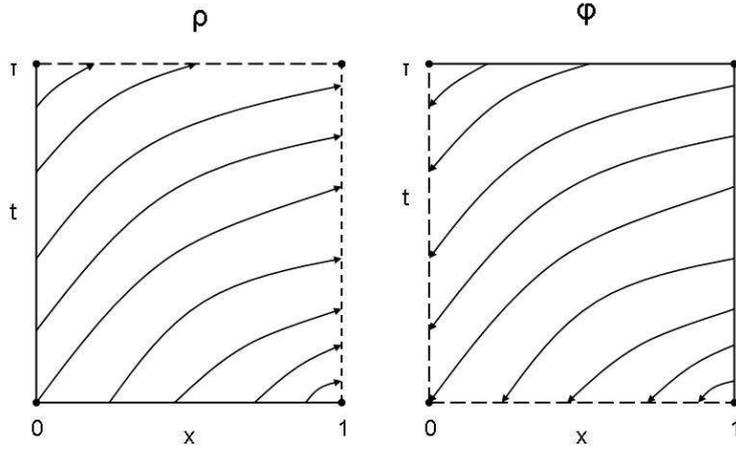


Figure 1: The characteristics for the ρ and ϕ PDEs. The solid border lines represent known data while the dashed border lines represent unknown data. The solution characteristics (the lines with arrows) generate the unknown data forwards in time for the ρ PDE and backwards in time for the ϕ PDE.

is an additional restriction on ϕ . We might interpret the condition as follows: The influx λ has to be chosen in such a way that the backwards solution to (7g) additionally satisfies (7i). For a numerical treatment of the optimization problem it is worthwhile to note that equation (7i) is a descent direction j for the reduced cost functional j . The reduced cost functional is defined by $j(\lambda) := J(\rho(\lambda), \lambda)$. Here, $\rho(\lambda)$ for given λ is the solution to (7a)–(7d). Provided suitable

regularity we have

$$\frac{\partial}{\partial \lambda} j(\lambda) \delta \lambda = -\phi(0, t)$$

if ϕ and ρ are given by (7a)–(7h).

These observations motivate the following numerical procedure to solve the problem (5).

2.1 Numerical issues

To solve equations (7a) and (7g) numerically we discretize ρ and ϕ on a space-time grid (x_i, t_j) . Hence, the optimization problem (5) is a finite-dimensional constrained optimization problem which is solved using a nonlinear conjugate gradient method applied to the reduced cost functional $j(\lambda)$. The gradient of j is determined using the adjoint variables as discussed above:

- Determine the gradient of j for given $\lambda(t_i)$
- Part 1 - Solve the PDE for $\rho(x_i, t_j)$:
We numerically solve the ρ partial differential equation 7a forward in time with initial condition $\rho_0(x)$ and $\lambda(t)$. We use a spatial resolution Δx and a temporal resolution Δt set to $\Delta t = \frac{\Delta x}{v_{max}}$ to satisfy the CFL stability condition [18] and we use a standard upwind method due to the positivity of the velocity function:

$$\rho(x_i, t_j + \Delta t) = \rho(x_i, t_j) - \frac{v(t_n)\Delta t}{\Delta x} (\rho(x_i, t_j) - \rho(x_i - \Delta x, t_j)) \quad (8)$$

where $i := 1 \dots N$, $j := 1 \dots M$ with $x_1 = 0$, $x_N = 1$, $t_1 = 0$ and $t_M = \tau$.

- Part 2 – Solve the adjoint PDE for $\phi(x_i, t_j)$: The adjoint function $\phi(x, t)$ is determined by solving the three equations 7e, 7f and 7g on the same grid and the same upwind method but backwards in time.
- Part 3 - Finding $j'(\cdot)$
Having determined ϕ we evaluate the gradient of our cost functional as

$$j'(t_j) = -\phi(x_1, t_j) \text{ for all } t_j.$$

Having gradient information many methods can be applied to solve the minimization problem for the reduced cost functional [25, 22, 17]. In particular, second-order or Newton-methods could be used for solving the problem. Since we only deal with mid-size problems we simply apply the Polak-Ribière+ nonlinear conjugate gradient method [21] combined with a projection to ensure the positivity of λ and a line-search to guarantee sufficient descent. This method performs well for our problems and is easy to implement.

3 Numerical Results: Demand Tracking Experiments

In this section we discuss several different experiments for the demand tracking problem. Except where explicitly noted, we use a spatial grid of $\Delta x = 10^{-2}$ for the discretization of the PDEs. The stopping criteria for the optimization problem is $\|j'\| \leq \epsilon_{stop} = 10^{-3}$. Most simulations took on the order of 5 minutes. In the following results we demonstrate that the adjoint method works well for the demand tracking problems and the results agree with intuition.

3.1 A Step Demand Function

As a prototypical experiment we study a demand function that is constant and increases by a one step jump half way in the time interval. Specifically we have a constant initial density profile $\rho_0(x) \equiv 1$, a v_{max} of 4, a constant initial $\lambda_0 \equiv 2$ and a demand function that jumps from 2 to 3 halfway through our time interval at $t = 5$.

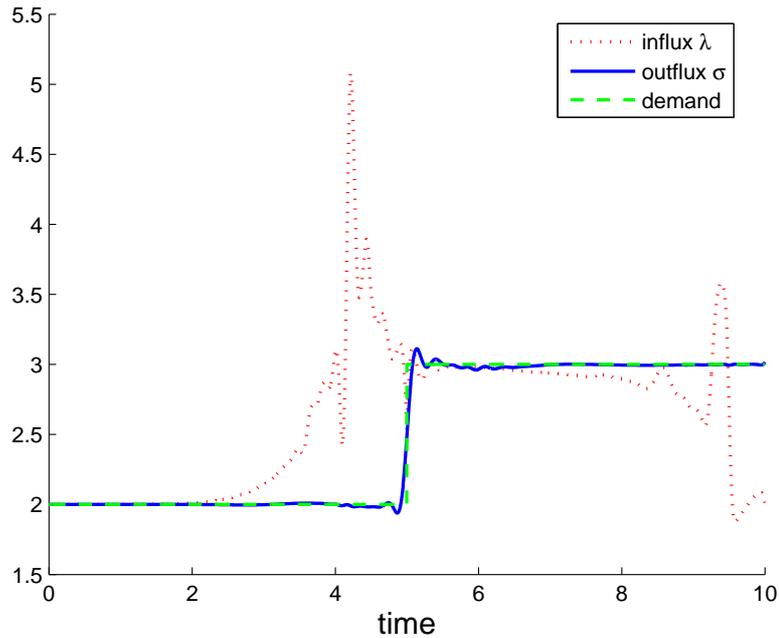


Figure 2: Influx, outflux, and demand for a step demand function from 2 to 3 at $t = 5$. ρ_0 was constant at 1, λ_0 was constant at 2, proceeded for 30 iterations, exiting due to maximum iteration. Note the end effect.

Fig. 2 displays the demand target rate and optimal influx, and outflux

determined via the adjoint algorithm over the time interval $t \in [0, 10]$. The figure shows that we can generate an influx that closely matches the desired outflux by putting a large amount of material into the factory in a relatively short amount of time, resulting in a influx spike. The figure also shows some features that need to be explained: oscillations in the influx near the end time $\tau = 10$, and oscillations of the outflux near the discontinuity. The oscillations near the end are due to our choice of the cost functional: We only try to fit the demand in an average way on $(0, T)$. Hence, there is no information on the pointwise behavior. The reason behind this is illustrated by the characteristic picture for the ϕ PDE (Fig. 1). In the upper left corner, $\phi(0, \tau)$ is set to zero hence the derivative of the cost function there is zero. When the source terms of the ϕ PDE are small or zero, the ϕ PDE becomes a simple advection equation. Therefore, the value of ϕ is constant along its characteristics and hence carries the values at the top boundary conditions $\phi(x, \tau) = 0$ to the gradient, which is $-\phi(0, t)$. Hence, in our domain near $(x = 0, t = \tau)$, the gradient will stay at or near zero, and λ will not be changed or changed very little as a result of the conjugate gradient method. Clearly, this effect will vanish as soon as a terminal constraint is added to the cost functional, e.g.,

$$J(\rho, \lambda) = \frac{1}{2} \int_0^\tau (d(t) - v(\rho)\rho(1, t))^2 dt + \frac{1}{2} (d(T) - v(\rho)\rho(1, T))^2.$$

The oscillations are not induced by the numerics.

3.1.1 Convergence of the cost functional

We examine the cost functional as we iterate λ_n for the experiment shown in Fig. 2. Figure 3 shows the cost $j(\lambda_n)$ on a log scale at each iteration for 75 iterations. This figure displays large drops in cost during the first few iterations until locking on to a local extremum whereby it displays a roughly linear decrease. This indicates that for the first few iterations the method is descending upon a multidimensional saddle point until a new search direction is chosen that results in a more significant drop. The method then proceeds to descend along this direction in much the same way as it initially did before locking on to the local minimum. While the method only finds local minima, the cost function is bounded below by zero.

3.1.2 Convergence to the minimizer $\lambda^*(t)$

To examine the convergence of the adjoint method, we first look at the results of our algorithm for different numbers of iterations. Figs. 4(a), and 4(b)) display the results of the PR+ algorithm for the same experiment as in Fig. 2 for 10 and 300 iterations respectively.

After 10 iterations we are still far from an ideal solution, but after 300 iterations our output matches the demand nearly perfectly. Our control λ , however, develops more and more oscillations. This effect is well known in

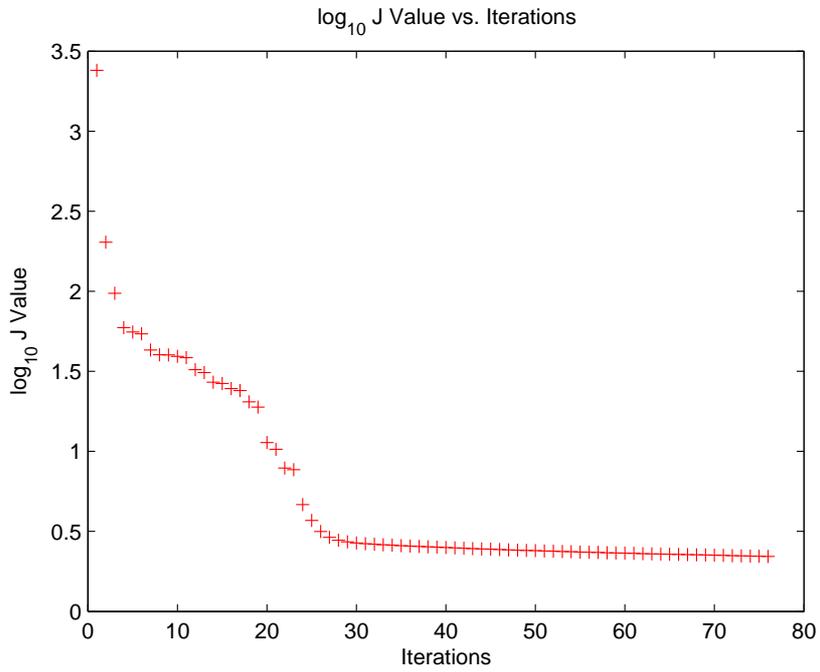


Figure 3: The decrease in $j(\lambda_k)$ with each iteration for the same experiment as in Fig. 2

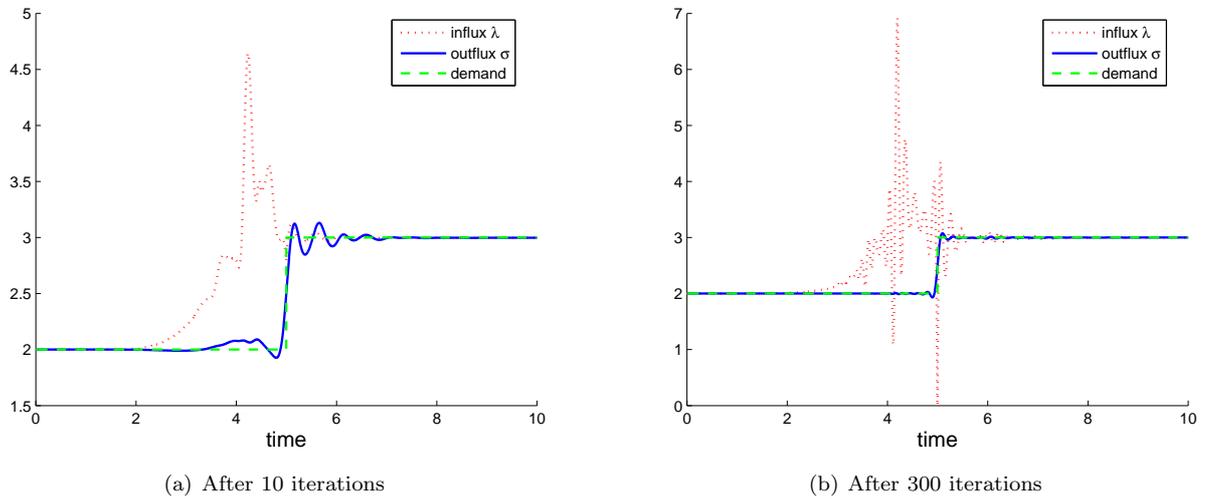


Figure 4: The PR+ conjugate gradient results for different numbers of iterations for the same conditions as in Fig. 2

control theory under the name of *chattering* - the optimal solution will have infinitely many oscillations in a finite time interval.

Smoothing the influx by taking a moving average and examining the corresponding outflux after 100 iterations supports the chattering explanation: Fig. 5 shows that the average influx produces a ramp outflux. Furthermore a smaller moving average window size, which correlates to a less-smooth influx, produces a steeper jump. Therefore, the oscillations in the influx generate an outflux discontinuity. Chattering is further supported by the fact that the frequency of the input oscillations increases as the spatial step Δx and the associated time step Δt become smaller. Note that chattering disappears if the demand increase is a ramp rather than a step function. We conduct an experiment where the demand will increase from a steady demand level of two at $t = 5$ linearly to an steady demand level of three within different time intervals. Fig. 6 shows output and input after 100 iterations of the optimization scheme. These ramps do not develop the oscillations in the influx characteristic for the step demand.

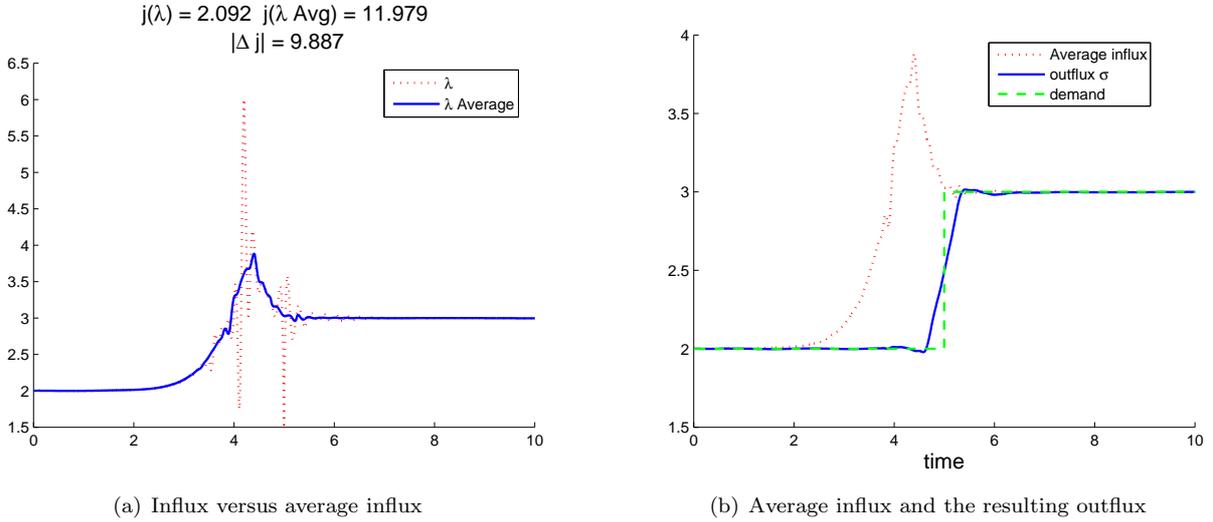
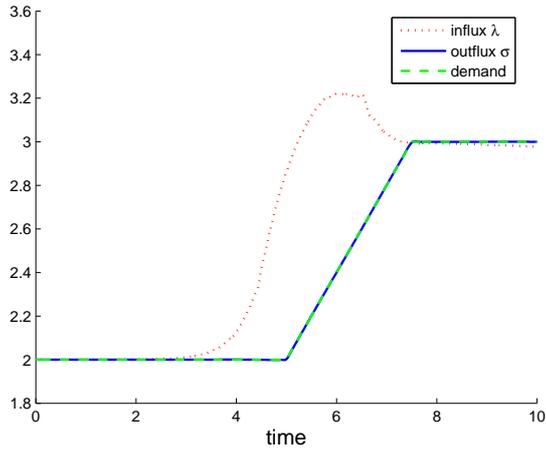


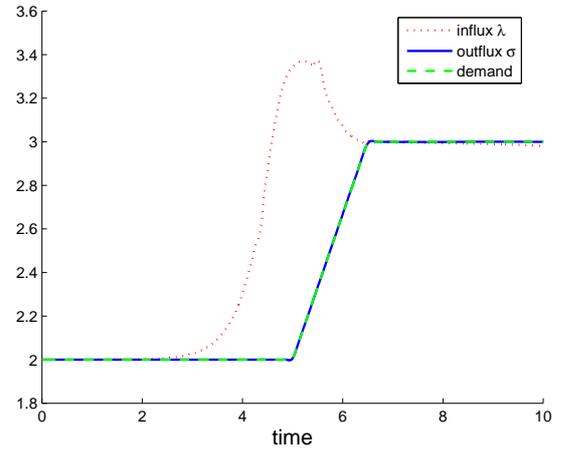
Figure 5: The influx and the moving average influx (window 1/2 time unit centered at the time step) after 100 iterations and the corresponding outflux for the average influx under the same conditions as in Fig. 2

3.2 Factory Reactivity

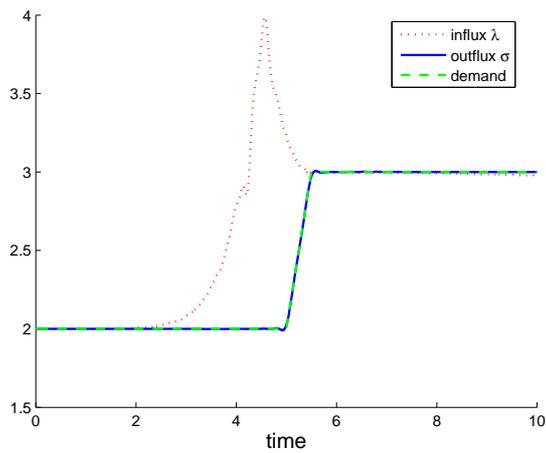
To examine the response of a factory to changes in the demand, known as the *reactivity* of the factory, it is helpful to examine the relationship between the velocity and the load. As Fig. 7 shows, the nonlinearity of the velocity function results in sharply varying behavior for different values of the maximum velocity v_{max} .



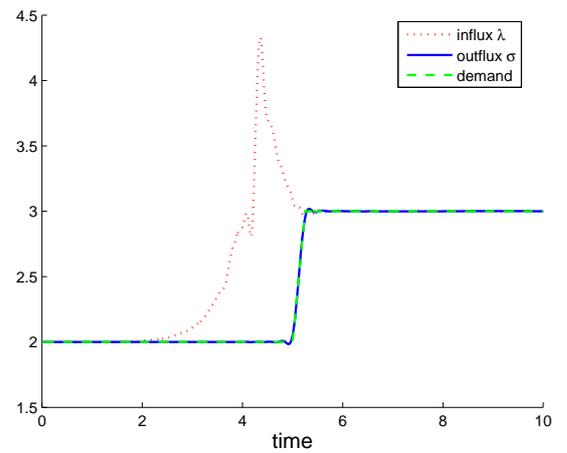
(a) Ramp ends at $t=7.5$



(b) Ramp ends at $t=6.5$



(c) Ramp ends at $t=5.5$



(d) Ramp ends at $t=5.25$

Figure 6: Results for a demand ramp with greater slopes for the same conditions as in Fig. 2. Note that the oscillations that occurred in Fig. 4 are no longer present.

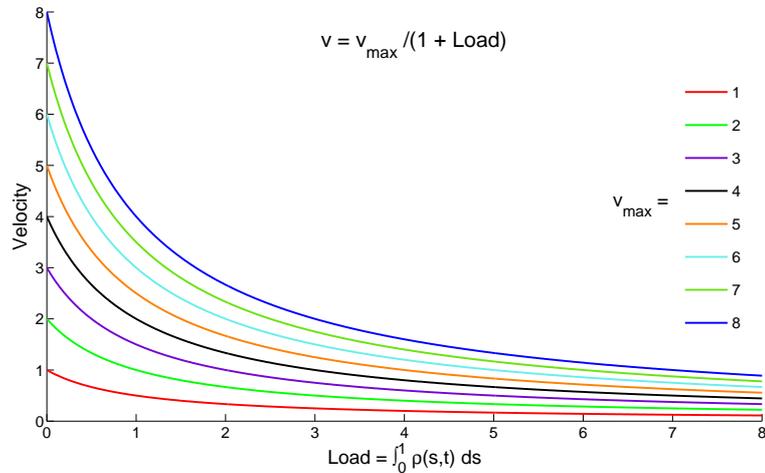


Figure 7: The velocity vs. the load for different values of v_{max}

Note that with large v_{max} , a unit change in the load has a much greater effect in the velocity than for a lower v_{max} , especially for a small load. For high loads the response to an increase in the load is not much different but higher values of v_{max} have velocities at an overall higher level. Therefore, the combination of both the v_{max} and the factory load define the factory's reactivity. The next few examples highlight different system reactivities and their consequences.

3.2.1 The effect of v_{max}

The following scenarios highlight the difference in a system's ability to react to the same oscillatory demand rate target depending on the v_{max} . Note that v_{max} is the reciprocal of the raw processing (cycle) time of the factory. These examples illustrate the benefit of a small cycle time and were run with the exact same conditions with the exception of the different values of v_{max} . Each experiment was run for 50 iterations with a τ of 10 with the sinusoidal demand rate target

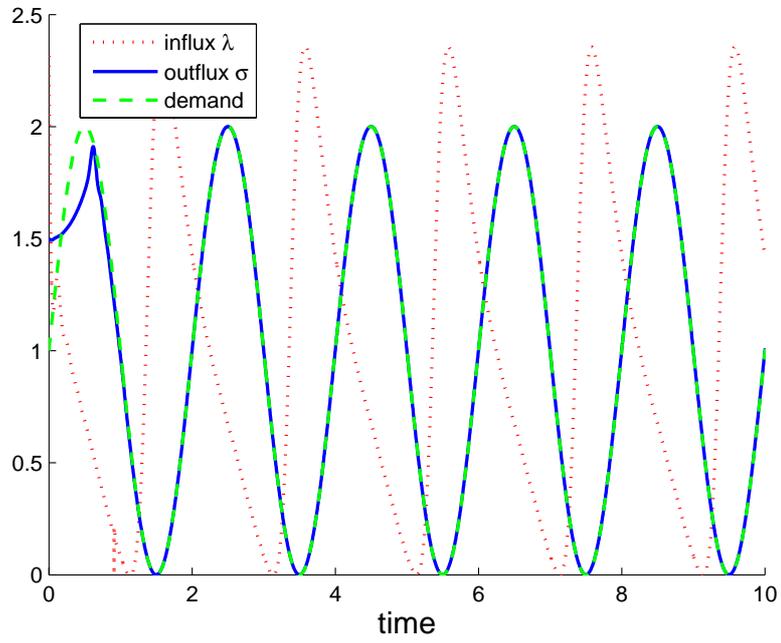
$$d(t) = \sin(\pi t) + 1$$

with constant $\rho_0(x) = 1$ and constant $\lambda_0(t) = 2$.

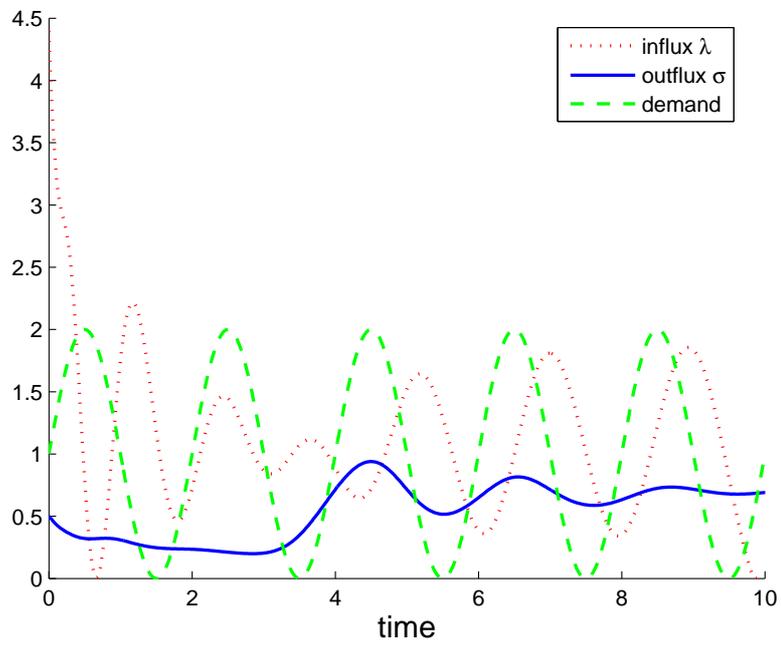
We see in Fig. 8(a) (high v_{max}) that after an initial transient the match between the outflux and demand becomes perfect. More interestingly, for low v_{max} (Fig. 8b)) the outflux can not be changed fast enough to follow the demand oscillation.

3.2.2 The effect of the load

The next experiments show the difference in the system's response depending on it's load. In these the demand function steps up one unit to a new steady



(a) $v_{max} = 3$



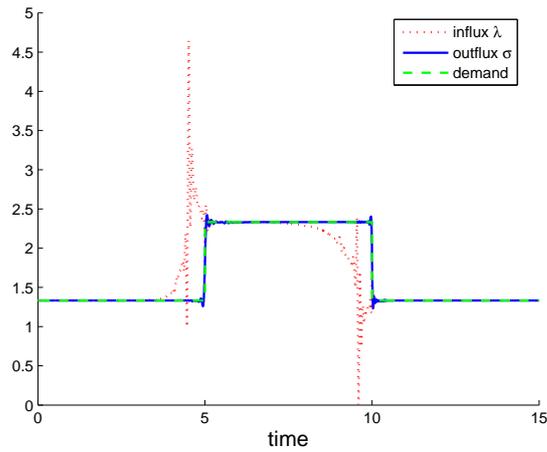
(b) $v_{max} = 1$

Figure 8: The effect of v_{max} on a sinusoidally oscillating demand function

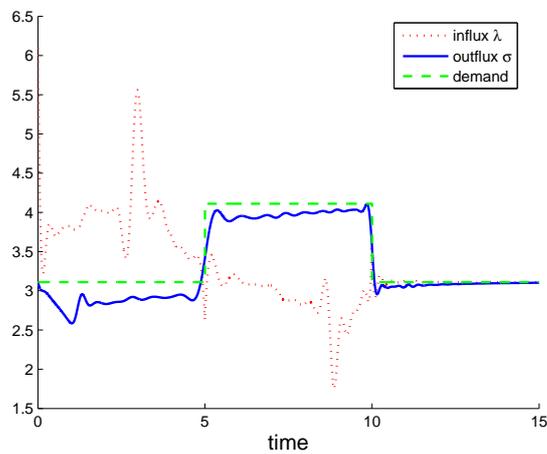
state and then later steps down to the original steady state. The maximum velocity is four for all experiments, Since the steady state is determined by

$$\rho_{ss} = \frac{\lambda_{ss}}{v_{max} - \lambda_{ss}} \quad (9)$$

a steady state cannot be established above an outflux of 4. Figure 9 shows that the system follows easily a step up and down for an underloaded factory (from 12.5% to 37.5% loading) but that a step from 75% to 100% loading is difficult.



(a) $\rho_{ss} = 0.5$

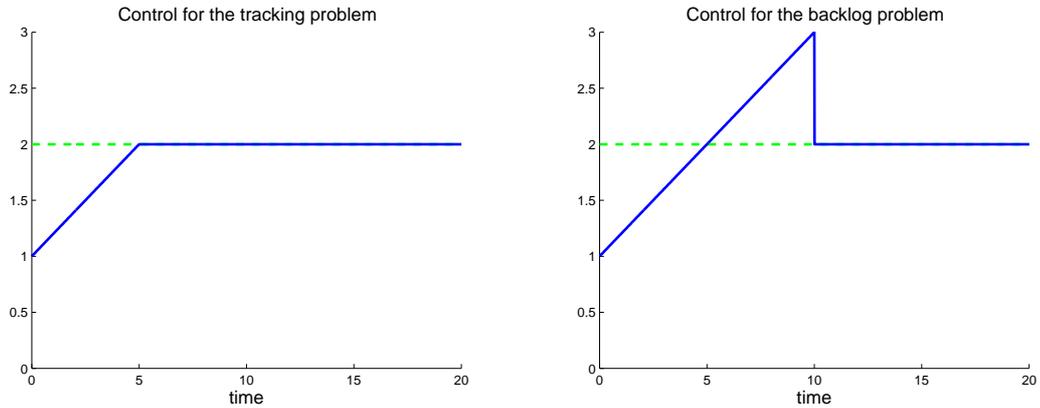


(b) $\rho_{ss} = 3.5$

Figure 9: The effect of the load on a step up/step down demand function

4 The Backlog Problem

The backlog of a production system at a given time t is defined as the total number of items that have been demanded minus the total number of items that have left the factory up to that time. Backlog can be negative or positive, with a negative backlog corresponding to overproduction and a positive backlog corresponding to a shortage. The backlog problem differs from the demand tracking problem in one key way: errors are not forgotten. Fig. 10(a) and Fig. 10(b) schematically illustrate the difference for a demand with a step increase. Given a maximum rate of increase, the best approach in the demand tracking problem is to increase the outflux (solid blue line) until it reaches the demand target (dashed green line) and then stay there. The best approach in the backlog tracking problem is to maximally increase the outflux until the area below the demand (rate) target and above the outflux equals the area above the demand target and below the outflux. At that time the backlog is zero and the outflux will drop to the demand target.



(a) Schematic influx response to a demand jump for demand tracking (b) Schematic influx response to a demand jump for backlog tracking

Figure 10: Demand tracking vs. backlog tracking

4.1 Cost functional and adjoint approach

It is useful to define the cumulative demand and the cumulative outflux at time t as

$$D(t) := \int_0^t d(r) dr$$

$$O(t) := \int_0^t v(\rho)\rho(1, r) dr \equiv \int_0^t \sigma(r) dr.$$

Hence, the backlog at time t becomes

$$b(t) := \int_0^t (d(r) - v(\rho)\rho(1, t)) dr \equiv D(t) - O(t).$$

We define a cost functional over a fixed time period from $[0, \tau]$, as

$$J(\rho, \lambda) = \frac{1}{2} \int_0^\tau b(t)^2 dt \quad (10)$$

where the outflux is generated by our usual PDE model (3a). The cost functional Eq. 10 penalizes overproduction and underproduction equally. This is not usually the case - typically overproduction has small holding costs whereas underproduction may lead to contractual penalties that are much higher. Taking account of different costs will make the cost functional non-smooth. While minimization of the cost functional may still be possible, this question will not be addressed in this research.

Minimizing a Lagrangian using the same approach as in the demand tracking problem but with significantly more involved algebra (see Appendix 6.2) we arrive at a coupled system of two PDEs - one for the density $\rho(x, t)$ and one for the adjoint variable $\phi(x, t)$ that are coupled through the boundary conditions:

$$0 = \rho_t(x, t) + v(\rho)\rho_x(x, t) \quad (11a)$$

$$0 = \phi(x, \tau) \quad (11b)$$

$$\phi(1, t) = \int_t^\tau b(c) dc \quad (11c)$$

$$\phi_t(x, t) + v(\rho)\phi_x(x, t) = \frac{v(\rho)^2}{v_{max}}\rho(1, t) \int_t^\tau b(c) dc - \frac{v(\rho)^2}{v_{max}} \int_0^1 \phi(x, t)\rho_x(x, t) dx \quad (11d)$$

$$0 = -\phi(0, t) \quad (11e)$$

Notice that the boundary condition for the ϕ equation is related to the integrated backlog and can again be determined after the ρ equation has been solved, allowing the same numerical approach for solving the PDEs as in the demand tracking problem and the same iterative algorithm to find a minimum of the cost functional.

4.2 Backlog Results

The following experiments were run with the same parameters as the demand tracking problem unless otherwise noted. A backlog experiment typically takes longer than a comparable experiment for the demand tracking problem,

We will again examine the case of a step demand rate function; We consider a demand rate function that goes from 2 to 3 halfway through our time interval of $\tau = 10$, a constant initial density profile of $\rho_0(x) \equiv 1$, a v_{max} of 4, and a constant

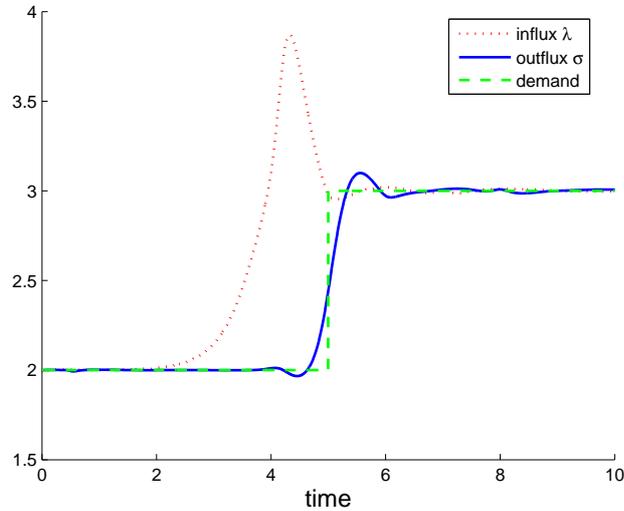


Figure 11: Influx, outflux, and demand rate for a step demand function from 2 to 3 at $t = 5$.

initial influx rate of $\lambda(t) \equiv 2$. Fig.11 shows the results of this experiment after 50 iterations. Notice that the oscillations in λ have disappeared, both at the end as well as near the discontinuity. The explanation for this lies in the boundary condition of the ϕ -equation $\phi(1, t) = \int_t^\tau b(c) dc$. Given that the ρ and the ϕ PDEs are hyperbolic the iteration algorithm for the optimization really involves the following steps $\lambda_n \rightarrow \phi_n(1, t) \rightarrow \lambda_{n+1} \rightarrow \phi_{n+1}(1, t)$ etc. However, since for the backlog problem $\phi(1, t)$ involves an integration step, any oscillations in time will be smoothed out.

Figure 12 shows the optimal influx pattern for a jump in the demand that happens very early in the control interval. The influx seems to follow a heuristic of increasing the influx to make up for missed backlogs and then exponentially reduce to the new steady state influx. Note the strong inverse response showing the decrease of the outflux due to the increase of the influx.

Fig. 13, shows the difference of demand tracking and backlog tracking for the oscillatory demand $d(t) = \sin(\pi t) + 1$ discussed before. Clearly the demand tracking problem minimizes the time for the outflux to match the demand signal whereas the backlog problem overproduces to reduce the backlog to zero as fast as possible.

5 Conclusion

We have shown how formally to generate an input control for the output and backlog tracking problem of a hyperbolic nonlinear and nonlocal transport equa-

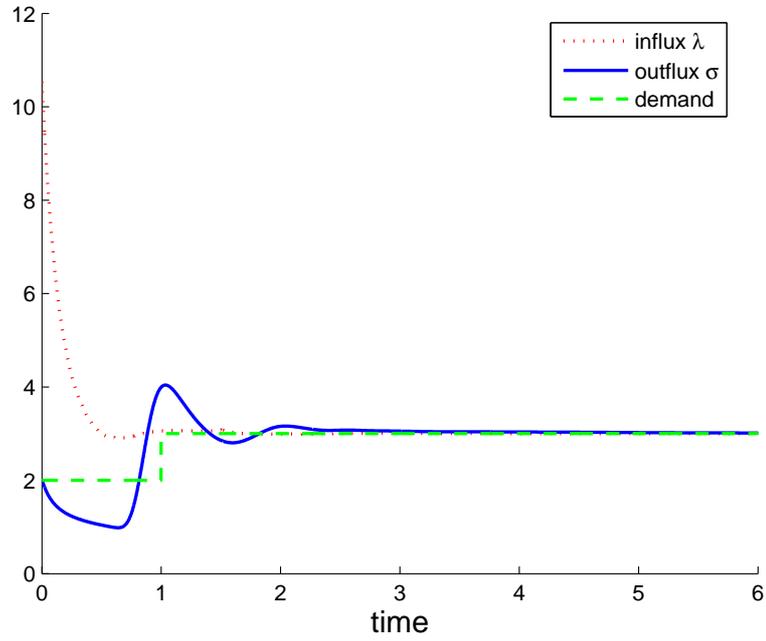


Figure 12: Backlog tracking for a demand jump that comes too early for the influx to generate an outflux that increases at the right time. Note the much higher scale compared to the previous figure.

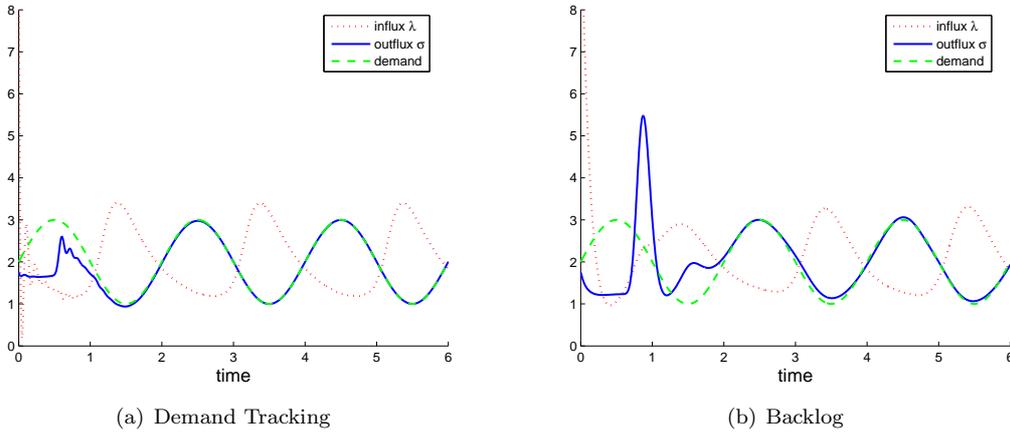


Figure 13: The results of an oscillatory demand for the demand tracking problem and the backlog problem.

tion. The algorithm based on the adjoint method of optimizing a cost functional with constraints is an extremely powerful and useful tool and it applies perfectly to the continuum model of factory production. While we have not been concerned with mathematical fundamentals e.g. question of uniqueness of the minimizer or even the existence of the solutions of the associated PDEs or the existence of a minimizer at all, our extensive simulations indicate that the method allows to solve the tracking problem for most practical cases. Such a statement is relatively easy to make since the results of the optimization algorithms are easy to evaluate: The associated cost function has a lower bound of zero and the associated minimizer can be compared to the minimizing input for a linear hyperbolic equation which is generated by a simple delay of the desired output.

Using the control algorithm we are able to make some interesting observations about the behavior of continuum models of production systems: We can quantify the reactivity of a factory and show how that leads to increased or decreased sensitivity to changes in the input. We also notice that discontinuous changes in the demand seem to lead to chattering. Furthermore we can illustrate the influence of the nonlinear behavior of the production line nicely by its conversion of a saw-tooth like input function into a harmonic outflux.

To specifically use this algorithm for a given production system the following steps will have to be done:

1. The continuum model will have to be parameterized via a suitable clearing function. Extracting such a function out of real or simulation data is not obvious and subject to current research [29].
2. As long as the clearing function is a monotone function of the load or the local density, the adjoint calculus will lead to similar equations as discussed in this paper and hence a optimized input function $\lambda^*(t)$ can be calculated easily.
3. $\lambda^*(t)$ will be discretized to generate daily inputs.

Transferring this algorithm to a real world production system involves additional future research:

- A factory has additional constraints - we need to incorporate a maximal influx λ_{max} as well as a constraint on the maximal change in the influx since factories can not change their production rate arbitrarily fast. This adds additional constraints to the optimization problem.
- Similarly, the cost functional may be made much more realistic at the expense of differentiability.
- Clearing functions involving batch processes are not necessarily monotonically increasing. It is currently unclear whether and how the adjoint algorithm works in this case.
- Our current control algorithm can be considered as a open loop control algorithm. Such algorithms are not robust under perturbations. Clearly

a small stochastic perturbation in the density level at an early time in our control horizon will have a large effect on the error of your demand tracking problem later on. We are working on developing a model predictive control approach [11] that uses the adjoint method to control a discrete event simulation (DES). The resulting optimal influx will be implemented in the DES only for a short control window. The resulting new state will then be fed back into the adjoint algorithm and an updated optimal influx will be generated. We expect that such a feedback loop will lead to a control algorithm that is robust against stochastic fluctuations and model mismatch.

- Control of production systems based on the influx is a large scale control algorithm - long time scales and over many production steps. In [23] we have shown that dispatch policies can be used to effect small scale control - short time scales and acting on individual machines. The integration of both approaches would enable us to control for e.g. seasonal demand swings while at the same time adjust for daily production variation and daily demand.

Furthermore there are several theoretical questions that we are interested to follow up:

- All of the above calculations have been done in a formal way. A full theoretical underpinning needs to discuss the existence and regularity properties of the solutions of the forward and the adjoint partial differential equation.
- The current gradient based optimization search finds a local minimum. Are there special cases when we can prove that the local minimum is actually the global minimum?
- An exciting theoretical question comes from the discretization of the continuum model. Production systems modeling has been analyzing so called fluid models for a long time (see eg. [7]). They are coupled ODEs that mimic the behavior of queues in front of machines. It has been shown that such models in the limit of large number of queues can be modeled by the continuum model [2]. On the other hand optimal control of a finite number of ODEs is a solved problem based on Pontryagin's maximum principle [20]. The convergence of such an algorithm to the adjoint algorithms of controlling PDEs is an open question.

Acknowledgements

D.A. and C.R. are supported by NSF grant DMS-0604986, M.H. is supported by grants from the DFG (SPP 1253) and the DAAD (D/06/28176). D.A gratefully acknowledges support by a grant from the Stiftung Volkswagenwerk

References

- [1] C. Agnew. Dynamic modeling and control of congestion-prone systems. *Operations Research*, 24(3):400–419, 1976.
- [2] D. Armbruster, P. Degond, and C. Ringhofer. A model for the dynamics of large queuing networks and supply chains. *SIAM J. Applied Mathematics*, 66(3):896–920, 2006.
- [3] D. Armbruster, D. Marthaler, C. Ringhofer, K. Kempf, and Tae-Chang Jo. A continuum model for a re-entrant factory. *Operations Research*, 54(5):933–950, 2006.
- [4] François Bouchut and François James. Differentiability with respect to initial data for a scalar conservation law. Fey, Michael (ed.) et al., *Hyperbolic problems: Theory, numerics, applications. Proceedings of the 7th international conference, Zürich, Switzerland, February 1998. Vol. I.* Basel: Birkhäuser. ISNM, Int. Ser. Numer. Math. 129, 113-118 (1999)., 1999.
- [5] A. Bressan and W. Shen. Optimality conditions for solutions to hyperbolic balance laws. *preprint*, 2007.
- [6] Alberto Bressan and Wen Shen. Optimality conditions for solutions to hyperbolic balance laws. Ancona, Fabio (ed.) et al., *Control methods in PDE-dynamical systems. AMS-IMS-SIAM joint summer research conference, Snowbird, UT, USA, July 3–7, 2005.* Providence, RI: American Mathematical Society (AMS). Contemporary Mathematics 426, 129-152 (2007)., 2007.
- [7] J. G. Dai and J.H. Vande Vate. The stability of two-station multitype fluid networks. *Operations Research*, 48.
- [8] J. de Halleux, C.Prieur, J.M. Coron, B. d’Andrea Novel, and G. Bastin. Boundary feedback control in networks of open channels. *Automatica J. IFAC*, 39(8):1365–1376, 2003.
- [9] K. Ehrhardt and M. Steinbach. Nonlinear gas optimization in gas networks. *Modeling, Simulation and Optimization of Complex Processes (eds. H. G. Bock, E. Kostina, H. X. Pu, R. Rannacher)*, Springer Verlag, Berlin, 2005.
- [10] R. Fletcher. *Practical Methods of Optimization*. John Wiley and Sons, New York, 2nd edition, 1987.
- [11] C.E. Garcia, D.M. Prett, and M.Moriari. Model predictive control: Theory and practice- a survey. *Automatica*, 25(3):335–348.
- [12] Jean Charles Gilbert and Jorge Nocedal. Global convergence properties of conjugate gradient methods for optimization. *SIAM Journal on Optimization*, 2:21–42, 1992.

- [13] Edwige Godlewski and Pierre-Arnaud Raviart. The linearized stability of solutions of nonlinear hyperbolic systems of conservation laws. A general numerical approach. *Math. Comput. Simul.*, 50(1-4):77–95, 1999.
- [14] M. Gugat. Optimal nodal control of networked hyperbolic systems: evaluation of derivatives. *Adv. Model. Optim.*, 7:9–37, 2005.
- [15] M. Gugat, G. Leugering, and E.J.P. Georg Schmidt. Global controllability between steady supercritical flows in channel networks. *Math. Methods Appl. Sci.*, 27(7):781–802, 2004.
- [16] U.S. Karmarkar. Capacity loading and release planning in work-in-progress (wip) and lead-times. *J. Mfg. Oper.Mgt.*, 2:105–123, 1989.
- [17] C.T. Kelley. *Iterative methods for optimization*. Frontiers in Applied Mathematics. Philadelphia, PA: Society for Industrial and Applied Mathematics. xv, 180 p. , 1999.
- [18] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, New York, 2002.
- [19] J.J. Moré and D.J. Thuente. Line search algorithms with sufficient decrease. *ACM Transactions on Mathematical Software*, 20:286–307, 1994.
- [20] Desineni Subbaram Naidu. *Optimal Control Systems*. CRC Press, New York, 2003.
- [21] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006.
- [22] Jorge Nocedal and Stephen J. Wright. *Numerical optimization. 2nd ed.* Springer Series in Operations Research and Financial Engineering. New York, NY: Springer. xxii, 664 p. EUR 74.85 , 2006.
- [23] D. Perdaen, D. Armbruster, K. Kempf, and E. Lefebvre. Controlling a re-entrant manufacturing line via the pushpull point. *International Journal of Production*, pages 1–16, 2007.
- [24] David Simchi-Levi, Philip Kaminsky, and Edith Simchi-Levi. *Designing & Managing The Supply Chain*. McGraw-Hill Irwin, New York, 2nd edition, 2003.
- [25] Peter Spellucci. *Numerical methods of nonlinear optimization. (Numerische Verfahren der nichtlinearen Optimierung.)*. ISNM Lehrbuch. Basel: Birkhäuser Verlag. 576 S. , 1993.
- [26] Fredi Tröltzsch. *Optimal control of partial differential equations. Theory, procedures, and applications. (Optimale Steuerung partieller Differentialgleichungen. Theorie, Verfahren und Anwendungen.)*. Wiesbaden: Vieweg. x, 297 p. EUR 34.90 , 2005.

- [27] Stefan Ulbrich. A sensitivity and adjoint calculus for discontinuous solutions of hyperbolic conservation laws with source terms. *SIAM J. Control Optimization*, 41(3):740–797, 2002.
- [28] Stefan Ulbrich. Adjoint-based derivative computations for the optimal control of discontinuous solutions of hyperbolic conservation laws. *Syst. Control Lett.*, 48(3-4):313–328, 2003.
- [29] A. Unver and C. Ringhofer. Estimation of transport coefficients in re-entrant factory models. *submitted*, 2008.

6 Appendix

This appendix shows the calculations for the variational derivatives necessary to derive the systems of equations 7 for the demand tracking problem and 11 for the backlog problem.

6.1 Demand tracking

We would like to find $D_\rho L \delta \rho$, where our Lagrangian $L(\rho, \lambda, \phi)$ is defined as:

$$\begin{aligned} L(\rho, \lambda, \phi) &:= J(\rho, \lambda) + \langle E(\rho, \lambda), \phi \rangle \\ &\equiv \frac{1}{2} \int_0^\tau (d(t) - v(\rho)\rho(1, t))^2 dt \\ &\quad + \int_0^1 \int_0^\tau \phi(x, t) [\rho_t(x, t) + v(\rho)\rho_x(x, t)] dt dx \end{aligned}$$

For clarity we will first find $D_\rho J$ and then find $D_\rho \langle E(\rho, \lambda), \phi \rangle$. By adding the them together we will have $D_\rho L$. Recall that $v'(\rho)\delta\rho = -\frac{v(\rho)^2}{v_{max}} \int_0^1 \delta\rho(s, t) ds$. Using integration by parts we can rewrite L as:

$$\begin{aligned} L(\rho, \lambda, \phi) &= \frac{1}{2} \int_0^\tau (d(t) - v(\rho)\rho(1, t))^2 dt \\ &\quad + \int_0^1 [\phi(x, \tau)\rho(x, \tau) - \phi(x, 0)\rho(x, 0)] dx \\ &\quad + \int_0^\tau [\phi(1, t)v(\rho)\rho(1, t) - \phi(0, t)v(\rho)\rho(0, t)] dt \\ &\quad - \int_0^1 \int_0^\tau \rho(x, t) [\phi_t(x, t) + v(\rho)\phi_x(x, t)] dt dx. \end{aligned}$$

6.1.1 Finding $D_\rho J$

We will be taking derivatives in a variational sense.

$$\begin{aligned}
D_\rho J &= \int_0^\tau (d(t) - v(\rho)\rho(1, t)) \left(\frac{v(\rho)^2}{v_{max}} \rho(1, t) \int_0^1 \delta\rho(s, t) ds - v(\rho)\delta\rho(1, t) \right) dt \\
&\downarrow \\
&= \int_0^\tau \frac{v(\rho)^2}{v_{max}} \rho(1, t) (d(t) - v(\rho)\rho(1, t)) \int_0^1 \delta\rho(s, t) ds dt \\
&\quad - \int_0^\tau v(\rho) (d(t) - v(\rho)\rho(1, t)) \delta\rho(1, t) dt
\end{aligned}$$

Putting terms together with the same variation we get:

$$\begin{aligned}
D_\rho J &= \int_0^\tau \frac{v(\rho)^2}{v_{max}} (\rho(1, t)d(t) - v(\rho)\rho(1, t)^2) \int_0^1 \delta\rho(s, t) ds dt \\
&\quad + \int_0^\tau (v(\rho)^2\rho(1, t) - v(\rho)d(t)) \delta\rho(1, t) dt
\end{aligned}$$

6.1.2 Finding $D_\rho \langle E(\rho, \lambda), \phi \rangle$

$$\begin{aligned}
D_\rho \langle E(\rho, \lambda), \phi \rangle &= \int_0^1 [\phi(x, \tau)\delta\rho(x, \tau) - \phi(x, 0)\delta\rho(x, 0)] dx \\
&\quad + \int_0^\tau \phi(1, t) \left[-\frac{v(\rho)^2}{v_{max}} \rho(1, t) \int_0^1 \delta\rho(s, t) ds + v(\rho)\delta\rho(1, t) \right] dt \\
&\quad - \int_0^\tau \phi(0, t) \left[-\frac{v(\rho)^2}{v_{max}} \rho(0, t) \int_0^1 \delta\rho(s, t) ds + v(\rho)\delta\rho(0, t) \right] dt \\
&\quad - \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho)\phi_x(x, t)] \delta\rho(x, t) dt dx \\
&\quad + \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \rho(x, t)\phi_x(x, t) \int_0^1 \delta\rho(s, t) ds dt dx
\end{aligned}$$

Regrouping:

$$\begin{aligned}
D_\rho \langle E(\rho, \lambda), \phi \rangle &= \int_0^1 [\phi(x, \tau) \delta \rho(x, \tau) - \phi(x, 0) \delta \rho(x, 0)] dx \\
&+ \int_0^\tau [\phi(1, t) v(\rho) \delta \rho(1, t) - \phi(0, t) v(\rho) \delta \rho(0, t)] dt \\
&- \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho) \phi_x(x, t)] \delta \rho(x, t) dt dx \\
&+ \int_0^\tau \frac{v(\rho)^2}{v_{max}} [\phi(0, t) \rho(0, t) - \phi(1, t) \rho(1, t)] \int_0^1 \delta \rho(s, t) ds dt \quad (12) \\
&+ \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \rho(x, t) \phi_x(x, t) \int_0^1 \delta \rho(s, t) ds dt dx \quad (13)
\end{aligned}$$

For the term (12) we may bring out the spatial integral, change the integration variable, and change the order of integration to obtain:

$$\begin{aligned}
&\int_0^\tau \frac{v(\rho)^2}{v_{max}} [\phi(0, t) \rho(0, t) - \phi(1, t) \rho(1, t)] \int_0^1 \delta \rho(s, t) ds dt \\
&= \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} [\phi(0, t) \rho(0, t) - \phi(1, t) \rho(1, t)] \delta \rho(x, t) dt dx
\end{aligned}$$

and for the term (13) we can change the integration variables and integration order and bring out a spatial integral to rewrite as:

$$\begin{aligned}
&\int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \rho(x, t) \phi_x(x, t) \int_0^1 \delta \rho(s, t) ds dt dx \\
&= \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \rho(x, t) \phi_x(x, t) \int_0^1 \delta \rho(\theta, t) d\theta dt dx \\
&= \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \rho(s, t) \phi_x(s, t) \int_0^1 \delta \rho(\theta, t) d\theta dt ds \\
&= \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \left[\int_0^1 \rho(s, t) \phi_x(s, t) ds \right] \delta \rho(\theta, t) dt d\theta \\
&= \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \left[\int_0^1 \rho(s, t) \phi_x(s, t) ds \right] \delta \rho(x, t) dt dx
\end{aligned}$$

Now putting all terms together with the same variation and substituting (by integration by parts)

$$\begin{aligned}
&-\int_0^1 \phi(s, t) \rho_x(s, t) ds \\
&= \phi(0, t) \rho(0, t) - \phi(1, t) \rho(1, t) + \int_0^1 \rho(s, t) \phi_x(s, t) ds
\end{aligned}$$

we obtain:

$$D_\rho \langle E(\rho, \lambda), \phi \rangle = \int_0^1 [\phi(x, \tau) \delta \rho(x, \tau) - \phi(x, 0) \delta \rho(x, 0)] dx \quad (14a)$$

$$+ \int_0^\tau [\phi(1, t) v(\rho) \delta \rho(1, t) - \phi(0, t) v(\rho) \delta \rho(0, t)] dt \quad (14b)$$

$$- \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho) \phi_x(x, t)] \delta \rho(x, t) dt dx \quad (14c)$$

$$- \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \left[\int_0^1 \phi(s, t) \rho_x(s, t) ds \right] \delta \rho(x, t) dt dx \quad (14d)$$

6.1.3 Finding $D_\rho L$

Adding together $D_\rho J$ and $D_\rho \langle E(\rho, \lambda), \phi \rangle$ and grouping all like variations together gives the following equations

$$\begin{aligned} D_\rho L \delta \rho &= \int_0^1 [\phi(x, \tau)] \delta \rho(x, \tau) \\ &- \int_0^1 [\phi(x, 0)] \delta \rho(x, 0) \\ &+ \int_0^\tau [v(\rho) \phi(1, t) + v(\rho)^2 \rho(1, t) - v(\rho) d(t)] \delta \rho(1, t) \\ &- \int_0^\tau [v(\rho) \phi(0, t)] \delta \rho(0, t) \\ &- \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho) \phi_x(x, t)] \delta \rho(x, t) dt dx \\ &+ \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \left[\rho(1, t) d(t) - v(\rho) \rho(1, t)^2 - \int_0^1 \phi(s, t) \rho_x(s, t) ds \right] \delta \rho(x, t) dt dx \end{aligned}$$

We need to explicitly include our initial and boundary conditions

$$\begin{aligned} \lambda(t) &= v(\rho) \rho(0, t) \\ \rho_0(x) &= \rho(x, 0) \end{aligned}$$

into our adjoint method. Since we are given a fixed initial condition $\rho_0(x)$, $\delta \rho(x, 0)$ is zero. Similarly, since we are given a fixed $\lambda(t)$ and we assume that we have existence and uniqueness of solutions, ρ is determined at the left boundary and therefore cannot vary there. and hence the variations $\delta \rho(0, t)$ is zero.

The adjoint method requires that $D_\rho L \delta \rho$ be equal to zero. Hence

$$\begin{aligned}
0 &= D_\rho L \delta \rho \\
&\Downarrow \\
0 &= \int_0^1 [\phi(x, \tau)] \delta \rho(x, \tau) \\
&+ \int_0^\tau [v(\rho)\phi(1, t) + v(\rho)^2\rho(1, t) - v(\rho)d(t)] \delta \rho(1, t) \\
&- \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho)\phi_x(x, t)] \delta \rho(x, t) dt dx \\
&+ \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \left[\rho(1, t)d(t) - v(\rho)\rho(1, t)^2 - \int_0^1 \phi(s, t)\rho_x(s, t) ds \right] \delta \rho(x, t) dt dx.
\end{aligned}$$

By the weak form of the Fundamental Theorem Of The Calculus Of Variations we get

$$\begin{aligned}
0 &= \phi(x, \tau) \text{ almost everywhere in } [0, 1] \\
0 &= v(\rho)\phi(1, t) + v(\rho)^2\rho(1, t) - v(\rho)d(t) \text{ almost everywhere in } [0, \tau] \\
0 &= -\phi_t(x, t) - v(\rho)\phi_x(x, t) \\
&+ \frac{v(\rho)^2}{v_{max}} \left[\rho(1, t)d(t) - v(\rho)\rho(1, t)^2 - \int_0^1 \phi(s, t)\rho_x(s, t) ds \right] \\
&\text{almost everywhere in } (0, 1) \times (0, \tau).
\end{aligned}$$

We will ignore sets of measure zero and since $v(\rho) > 0$

$$\phi(x, \tau) = 0 \text{ for all } x \in [0, 1] \tag{15}$$

$$\phi(1, t) = d(t) - v(\rho)\rho(1, t) \text{ for all } t \in [0, \tau] \tag{16}$$

$$\phi_t(x, t) + v(\rho)\phi_x(x, t) = \tag{17}$$

$$\frac{v(\rho)^2}{v_{max}} \left[\rho(1, t)d(t) - v(\rho)\rho(1, t)^2 - \int_0^1 \phi(s, t)\rho_x(s, t) ds \right]$$

$$\text{for all } (x, t) \in (0, 1) \times (0, \tau)$$

6.2 Backlog problem

In the demand tracking problem we took the derivative of the Lagrangian with respect to ρ , grouped like variational terms together, and set the corresponding equations to zero. This is not possible in the backlog problem as we will have terms involving an integral over a variation $\int_0^1 \delta \rho(s, t) ds$. Therefore, a different technique will be used to find the adjoint equations.

6.2.1 Find $D_\rho L(\rho, \lambda, \phi)$

After integration by parts, we have the Lagrangian

$$\begin{aligned} L(\rho, \lambda, \phi) &= J(\rho, \lambda) + \langle E(\rho, \lambda), \phi \rangle \\ &= \frac{1}{2} \int_0^\tau \left(\int_0^t (d(r) - v(\rho)\rho(1, r)) dr \right)^2 dt + \int_0^1 [\phi(x, \tau)\rho(x, \tau) - \phi(x, 0)\rho(x, 0)] dx \\ &\quad + \int_0^\tau [\phi(1, t)v(\rho)\rho(1, t) - \phi(0, t)v(\rho)\rho(0, t)] dt - \int_0^1 \int_0^\tau \rho(x, t) [\phi_t(x, t) + v(\rho)\phi_x(x, t)] dt dx \end{aligned}$$

Breaking up the derivative into two parts for clarity, we have

$$\begin{aligned} D_\rho J \delta \rho &= \int_0^\tau \left(\int_0^t (d(r) - v(\rho)\rho(1, r)) dr \right) \left(\int_0^t \left[\frac{v(\rho)^2}{v_{max}} \rho(1, r) \int_0^1 \delta \rho(s, r) ds - v(\rho)\delta \rho(1, r) \right] dr \right) dt \\ &\quad \Downarrow \text{substituting } b(t) = \int_0^t (d(r) - v(\rho)\rho(1, r)) dr \\ &= \int_0^\tau b(t) \int_0^t \left[\frac{v(\rho)^2}{v_{max}} \rho(1, r) \int_0^1 \delta \rho(s, r) ds \right] dr dt \\ &\quad - \int_0^\tau b(t) \int_0^t [v(\rho)\delta \rho(1, r)] dr dt. \end{aligned}$$

Since $E(\rho, \lambda)$ has not changed from the demand tracking problem, we can use equations 14 from the demand tracking problem. Setting the variations $\delta \rho(0, t)$ and $\delta \rho(x, 0)$ to zero by the same reasoning as in the demand tracking problem we find

$$\begin{aligned} D_\rho L(\rho, \lambda, \phi) \delta \rho &= \int_0^\tau b(t) \int_0^t \left[\frac{v(\rho)^2}{v_{max}} \rho(1, r) \int_0^1 \delta \rho(s, r) ds \right] dr dt \\ &\quad - \int_0^\tau b(t) \int_0^t [v(\rho)\delta \rho(1, r)] dr dt + \int_0^1 [\phi(x, \tau)\delta \rho(x, \tau)] dx \\ &\quad + \int_0^\tau [\phi(1, t)v(\rho)\delta \rho(1, t)] dt - \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho)\phi_x(x, t)] \delta \rho(x, t) dt dx \\ &\quad - \int_0^1 \int_0^\tau \frac{v(\rho)^2}{v_{max}} \left[\int_0^1 \phi(s, t)\rho_x(s, t) ds \right] \delta \rho(x, t) dt \end{aligned}$$

6.2.2 Pointwise Approach To Equation Derivation

Since we cannot easily group like variational terms together in $D_\rho L$, we employ a different approach to deriving the adjoint equations. The key concept is to assume that, as in the weak version of the Fundamental Theorem Of Variational Calculus, that $D_\rho L = 0$ holds for every variation $\delta \rho(x, t)$. Therefore, by choosing

specific forms of the variation $\delta\rho(x, t)$, we can elucidate more information from $D_\rho L$, knowing that $D_\rho L = 0$ holds pointwise throughout our domain.

Operating formally, we can choose our variation $\delta\rho(x, t)$ to be separable delta functions of x and t with support on the interior of $[0, 1] \times [0, \tau]$, i.e.

$$\delta\rho(x, t) \equiv \delta(x - \alpha)\delta(t - \beta), \quad \text{with } \alpha \in (0, 1) \text{ and } \beta \in (0, \tau)$$

with the delta function(s) possessing the usual properties:

$$\begin{aligned} \delta(u) &= 0 \text{ for } u \neq 0 \\ \int_{\Omega} \delta(u - a) du &= 1 \text{ if } a \in \Omega \\ \int_{\Omega} f(u)\delta(u - a) du &= f(a), \quad a \in \Omega. \end{aligned}$$

Substituting this into $D_\rho L = 0$ (evaluating $\delta\rho$ at endpoints if required and changing the integration constant to reduce confusion) yields:

$$\begin{aligned} 0 &= D_\rho L(\rho(\lambda), \lambda, \phi) \\ &= \int_0^\tau b(c) \left[\int_0^c \frac{v(\rho)^2}{v_{max}} \rho(1, r) \int_0^1 \delta(s - \alpha)\delta(r - \beta) ds dr \right] dc \\ &\quad - \int_0^\tau b(c) \left[\int_0^c v(\rho)\delta(1 - \alpha)\delta(r - \beta) dr \right] dc \\ &\quad + \int_0^1 \phi(x, \tau)\delta(x - \alpha)\delta(\tau - \beta) dx \\ &\quad - \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho)\phi_x(x, t)] \delta(x - \alpha)\delta(t - \beta) dt dx \\ &\quad - \int_0^1 \int_0^\tau \left[\frac{v(\rho)^2}{v_{max}} \phi(x, t)\rho_x(x, t) \int_0^1 \delta(s - \alpha)\delta(t - \beta) ds \right] dt dx \\ &\quad + \int_0^\tau \phi(1, t)v(\rho)\delta(1 - \alpha)\delta(t - \beta) dt \end{aligned}$$

Since α is in $(0, 1)$ and β is in $(0, \tau)$,

$$\delta(1 - \alpha) = \delta(\tau - \beta) = 0$$

and

$$\int_0^1 \delta(s - \alpha) ds = 1$$

and hence

$$0 = \int_0^\tau b(c) \left[\int_0^c \frac{v(\rho)^2}{v_{max}} \rho(1, r)\delta(r - \beta) dr \right] dc \quad (18)$$

$$- \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho)\phi_x(x, t)] \delta(x - \alpha)\delta(t - \beta) dt dx \quad (19)$$

$$- \int_0^1 \int_0^\tau \left[\frac{v(\rho)^2}{v_{max}} \phi(x, t)\rho_x(x, t)\delta(t - \beta) \right] dt dx. \quad (20)$$

Given that for $\beta > c$ the δ function in term (18) does not have any support inside the integral, term (18) is equal to

$$\frac{v(\rho)^2}{v_{max}} \rho(1, \beta) \left[\int_0^\tau b(c) dc - \int_0^\beta b(c) dc \right] = \frac{v(\rho)^2}{v_{max}} \rho(1, \beta) \int_\beta^\tau b(c) dc.$$

With the evaluation of (19) and (20) we find

$$\begin{aligned} 0 &= \frac{v(\rho)^2}{v_{max}} \rho(1, \beta) \int_\beta^\tau b(c) dc \\ &\quad - [\phi_t(\alpha, \beta) + v(\rho)\phi_x(\alpha, \beta)] \\ &\quad - \frac{v(\rho)^2}{v_{max}} \int_0^1 \phi(x, \beta) \rho_x(x, \beta) dx. \end{aligned} \tag{21}$$

Because (α, β) can be any interior point of the domain, we would then expect that (21) holds for every point on the interior of our domain, giving us the adjoint PDE in ϕ

$$\phi_t(x, t) + v(\rho)\phi_x(x, t) = \frac{v(\rho)^2}{v_{max}} \rho(1, t) \int_t^\tau b(c) dc - \frac{v(\rho)^2}{v_{max}} \int_0^1 \phi(x, t) \rho_x(x, t) dx. \tag{22}$$

To determine the coupling condition, we must choose a variation that mimics a delta function on the boundary of our spatial domain at $x = 1$ and a delta function in the interior of the time domain. Spatially we would like a function δ_ϵ that has the following properties for some small parameter ϵ and $a \in \mathbb{R}$:

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} \delta_\epsilon(u - a) &= 0 \text{ for } u \neq a \\ \lim_{\epsilon \rightarrow 0} \delta_\epsilon(0) &= \infty \\ \lim_{\epsilon \rightarrow 0} \int_\Omega \delta_\epsilon(u - a) du &= 1 \text{ if } a \in \Omega \\ \lim_{\epsilon \rightarrow 0} \int_\Omega \delta_\epsilon(u - a) f(u) du &= f(a) \text{ if } a \in \Omega \end{aligned}$$

An example of a function with these properties is:

$$\delta_\epsilon(x) := \frac{1}{\epsilon\sqrt{2\pi}} e^{-x^2/2\epsilon^2}$$

By centering this function about $x = 1$ we can approximate a delta function there, so we take our variation to be:

$$\delta\rho(x, t) \equiv \delta_\epsilon(1 - x)\delta(t - \beta) \quad \text{with } \beta \in (0, \tau)$$

Substituting into $D_\rho L = 0$ gives:

$$\begin{aligned}
0 &= D_\rho L(\rho, \lambda, \phi) \\
&= \int_0^\tau b(c) \left[\int_0^c \frac{v(\rho)^2}{v_{max}} \rho(1, r) \int_0^1 \delta_\epsilon(1-s) \delta(r-\beta) ds dr \right] dc \\
&\quad - \int_0^\tau b(c) \left[\int_0^c v(\rho) \delta_\epsilon(0) \delta(r-\beta) dr \right] dc \\
&\quad + \int_0^1 \phi(x, \tau) \delta_\epsilon(1-x) \delta(\tau-\beta) dx \\
&\quad - \int_0^1 \int_0^\tau [\phi_t(x, t) + v(\rho) \phi_x(x, t)] \delta_\epsilon(1-x) \delta(t-\beta) dt dx \\
&\quad - \int_0^1 \int_0^\tau \left[\frac{v(\rho)^2}{v_{max}} \phi(x, t) \rho_x(x, t) \int_0^1 \delta_\epsilon(1-s) \delta(t-\beta) ds \right] dt dx \\
&\quad + \int_0^\tau \phi(1, t) v(\rho) \delta_\epsilon(0) \delta(t-\beta) dt
\end{aligned} \tag{23}$$

If we take the limit as $\epsilon \rightarrow 0$, we see that terms (23, 24) approach infinity while all other terms are bounded (order 1). Therefore to satisfy $D_\rho L = 0$ we must set these terms (23, 24) to zero (the other terms will be equal to zero due to the ϕ PDE and terminal condition). This yields:

$$\begin{aligned}
0 &= \int_0^\tau \phi(1, t) v(\rho) \delta_\epsilon(0) \delta(t-\beta) dt \\
&\quad - \int_0^\tau b(c) \left[\int_0^c v(\rho) \delta_\epsilon(0) \delta(r-\beta) dr \right] dc \\
&= \delta_\epsilon(0) v(\rho) \left[\phi(1, \beta) - \int_\beta^\tau b(c) dc \right]
\end{aligned}$$

Again since β can be any value in $(0, \tau)$ we would expect this to hold for any time $t \in (0, \tau)$ which gives our coupling condition:

$$\phi(1, t) = \int_t^\tau b(c) dc \quad \forall t \in (0, \tau)$$

We now have the ϕ PDE, the coupling condition, and by examining $D_\rho L = 0$ directly, the terminal condition for the backlog problem as (respectively):

$$\begin{aligned}
0 &= \frac{v(\rho)^2}{v_{max}} \rho(1, t) \int_t^\tau b(c) dc \\
&\quad - [\phi_t(x, t) + v(\rho) \phi_x(x, t)] \\
&\quad - \frac{v(\rho)^2}{v_{max}} \int_0^1 \phi(x, t) \rho_x(x, t) dx \quad \text{for all } (x, t) \in (0, 1) \times (0, \tau)
\end{aligned} \tag{25}$$

$$\phi(1, t) = \int_t^\tau b(c) dc \quad \text{for all } t \in [0, \tau] \tag{26}$$

$$\phi(x, \tau) \equiv 0 \quad \text{for all } x \in [0, 1] \tag{27}$$

7 Appendix

This appendix describes the details of the conjugate gradient numerical method used to find the minimizing influx $\lambda^*(t)$.

7.1 Line Search and β^+

The objective of the line search is to find a step length $s_k > 0$ that minimizes the cost functional $j(\lambda_k)$ in a particular search direction d_k . To find an exact local minimizer of $j(\lambda_{k+1})$ at each iteration would be computationally infeasible for our problem so this cannot be required. However, we would like to guarantee that we have taken a step length that is not too small. We also want that the slope of $j(\lambda_{k+1})$ is small, so that increasing s_k will either increase $j(\lambda_{k+1})$ or decrease it very little. These conditions are encapsulated in the strong Wolfe conditions [21]

$$j(\lambda_k + s_k d_k) \leq j(\lambda_k) + c_1 s_k j'(\lambda_k)^T d_k \quad (28a)$$

$$|j'(\lambda_k + s_k d_k)^T d_k| \leq c_2 |j'(\lambda_k)^T d_k| \quad (28b)$$

with $0 < c_1 < c_2 < 1$. We have implemented a step-length selection algorithm that obeys the strong Wolfe conditions (28) and is detailed in [10] with a pseudocode description in [21]. This algorithm takes in the search direction d_k , the current control λ_k , and a maximum value for s_k and returns a step length s_k that obeys the strong Wolfe conditions. We opted to use a combination of bisection and polynomial interpolation to choose our test step lengths. While this is in inexact line search (s_k is not guaranteed to be a local minimizer), when coupled with our PR+ conjugate gradient method, numerical experience shows that the algorithm is efficient [12], [21]. While not implemented, it is also possible to show global convergence of the PR+ conjugate gradient if the inexact line search method satisfies the strong Wolfe conditions and also the sufficient descent condition

$$j'(\lambda_k)^T d_k \leq -c_3 \|j'(\lambda_k)\|^2, \quad 0 < c_3 \leq 1.$$

Suitable line search methods can be found in [19]. In our work convergence to a desired tolerance could always be achieved and hence more advanced line search methods that guaranteed global convergence were not implemented.

Nonlinear conjugate gradient methods, including the PR+ method, are designed for unconstrained optimization, while we wish to constrain the method to the space where our control constraint

$$0 \leq \lambda(t) \leq \lambda_{max}$$

is not violated. We accomplished this numerically using the line search. First, our line search took in a maximum value for s_k , which we calculated as follows. Since d_k and λ_k are just vectors in \mathbb{R}^M , we can examine each of their components

individually. The only way that $\lambda_{k+1} = \lambda_k + s_k d_k$ can be greater than λ_{max} is if a component of d_k , d_k^i was positive and

$$s_k > \frac{\lambda_{max} - \lambda_k^i}{d_k^i}$$

Similarly, the only way that λ_{k+1} can become negative is if some component of d_k was negative and

$$s_k > \frac{\lambda_i}{|d_k^i|}$$

If d_k^i was zero, it is skipped as the choice of s_k will have no effect. By defining the maximum value of s_k to be the minimum value that made these inequalities false for all components of d_k , we can guarantee that our control constraint (7.1) will hold for all λ_k , regardless of d_k . As result of this and our choice of a nonnegative λ_0 , we ensure that $\rho(x, t)$ remains nonnegative for all iterations as well.

The enforcement of our control constraint had an important effect that needed to be dealt with. The line search method we used assumes that an interval exists where the strong Wolfe conditions hold. However, given our constrained space, it was possible that along a descent direction d_k ,

$$j(\lambda_k + s_k d_k)$$

was approximately linear with slope $j'(\lambda_k)^T d_k$ until the maximum value of s_k . In this case, there would be no choice of step length where the strong Wolfe conditions held. Therefore, s_k was chosen as the maximum value of s_k so as to minimize the function along this direction, which meant that a component of λ was now at either 0 or λ_{max} . The reasoning behind this is that along that search direction, the lowest value of $j(\lambda)$ occurs at a an infeasible step length, so the lowest feasible value occurs at the maximum step length. In future iterations when a component of λ_k was at a boundary and the search direction would take it into the infeasible region, its corresponding component of the search direction d_k was set to zero. This ensured that the control constraint was not violated and that other components of λ_k could still be changed so as to lower $j(\lambda)$ even further.

7.2 Exit conditions

We employed the following two exit conditions on our implementation of the PR+ conjugate gradient system. The most simple of the two was to halt after a maximum number of iterations. The other takes advantage of the fact that at a local minimum of $j(\lambda)$, $j'(\lambda)$ should be close to zero. In order to stop, we required that the norm of the derivative of j at the current control λ_k be significantly less than the norm of the derivative of j at the initial control λ_0 , i.e.

$$\|j'(\lambda_k)\|_2 \leq \epsilon_{stop} \|j'(\lambda_0)\|_2$$

This choice of stopping condition let us define an absolute stopping criteria across all of our experiments, as the number of components in $j'(\lambda)$ and hence the size of $\|j(\lambda_k)\|$ varied depending on the space time grid used in that particular experiment. Of course there are other choices of stopping criteria, such as non-improvement of the cost function or actual computational time. As we will see, in practice the stopping criteria was almost always the maximum iterations, and that due to the relatively fast nature of our algorithm, this stopping criteria could be and was set depending on the desired level of convergence.