

A Model for the Dynamics of large Queuing Networks and Supply Chains *

D. Armbruster [†]; P. Degond [‡]; C. Ringhofer [§]

March 1, 2004

Abstract

We consider a supply chain consisting of a sequence of buffer queues and processors with certain throughput times and capacities. Based on a simple rule for releasing parts from the buffers into the processors we derive a hyperbolic conservation law for the part density and flux in the supply chain. The conservation law will be asymptotically valid in regimes with a large number of parts in the supply chain. Solutions of this conservation law will in general develop concentrations corresponding to bottlenecks in the supply chain.

KEYWORDS: Supply chains, conservation laws, asymptotics.

AMS CLASSIFICATION: 65N35, 65N05

1 INTRODUCTION

This paper is concerned with the development and analysis of continuum models for supply chains. We consider a chain of M suppliers or processors S_0, \dots, S_{M-1} . In the generic picture of a supply chain (see c.f. [12] for an overview) each supplier processes a certain good (measured in units of parts) and passes it on to the next supplier in the chain. Labelling the parts by the

*This work was supported by NSF grant DMS-0204543.

[†]Department of Mathematics, Arizona State University, Tempe, AZ 85287-1804, USA. (dieter@math.la.asu.edu)

[‡]MIP, Laboratoire CNRS (UMR 5640), Université Paul Sabatier, 118, route de Narbonne, 31062 Toulouse Cedex 04, France (degond@mip.ups-tlse.fr)

[§]Department of Mathematics, Arizona State University, Tempe, AZ 85287-1804, USA (ringhofer@asu.edu).

index n , we denote by $\tau(m, n)$ the time at which part number n passes from supplier number $m - 1$ to supplier number m . The goal of supply chain modelling and control is to derive rules governing the evolution of the times $\tau(m, n)$, and, in further consequence, design such rules to, in some pre-defined sense, optimally manage the supply chain. There is a hierarchy of models available for this purpose. If the times $\tau(m, n)$ are used as primary variables, and therefore each part is considered individually, this leads to so called discrete event simulation models (see [7] for an overview), which represent the most exact, and computationally most expensive, simulation tool. On the other end of the spectrum lie so called fluid models, which replace the individual parts by a continuum and use rate equations for the flow of product through a supplier (see [1], [8] for an overview). For a large number of parts, fluid models are much less expensive but necessarily represent an approximation to the actual situation. As a compromise between the two extremes, so called traffic flow models have received a lot of attention recently. The name derives from the analogy of the parts moving like cars on a highway and the use of a large already developed body of theory for modelling traffic flows. This theory employs the methodology of an even older, and better developed theory, namely that of gas dynamics. So, discrete event simulation takes the place of particle based (i.e. Monte Carlo type) models for gases, which can be approximated by the equations of gas dynamics (see [10] for an overview) and so on. The analogy is of course not one to one, since the basic rules governing the parts in a supply chain, the cars on a highway and the molecules in a gas will be different [2],[3],[4], [6], [11], [16].

This paper is concerned with the derivation of a type of traffic flow model, namely a conservation law for a partial differential equation, out of very simple principles governing the evolution of the times $\tau(m, n)$. Given the times $\tau(m, n)$ conservation of the number of parts is expressed via the introduction of so called N- curves (originally defined by Newell [17]). The N-curve $U(t)$ at supplier S_m is given by the number of parts which have passed from processor S_{m-1} to processor S_m at time t , i.e. by

$$U(m, t) = \sum_{n=0}^{\infty} H(t - \tau(m, n)) , \quad (1)$$

where H denotes the usual Heaviside function. The flux from processor S_{m-1} into processor S_m is given by the derivative of $U(m, t)$, i.e.

$$F(m, t) = \frac{d}{dt}U(m, t) = \sum_{n=0}^{\infty} \delta(t - \tau(m, n)), \quad m = 0, \dots, M , \quad (2)$$

holds with $F(0, t)$ and $F(M, t)$ the total influx and outflux of the supply chain. So N- curves are just the antiderivatives of fluxes. The work in progress (WIP) $W(m, t)$ of processor S_m , the total number of parts currently at the supplier S_m at time t , is now given by the difference of two consecutive N-curves, i.e.

$$W(m, t) = U(m, t) - U(m + 1, t) + K(m), \quad m = 0, \dots, M - 1, \quad (3)$$

where the time independent constants $K(m)$ are determined from the initial situation. Combining (2) and (3) yields the conservation law

$$\frac{d}{dt}W(m, t) = F(m, t) - F(m + 1, t), \quad (4)$$

for the WIP $W(m, t)$ and the flux $F(m, t)$, both given in terms of the transition times $\tau(m, n)$. Note that in (4) $W(m, t)$ is a step function in time while $F(m, t)$ is a superposition of δ - functions. Furthermore, if each of the suppliers S_m has a given minimal processing time $T(m)$, $\tau(m + 1, n) \geq \tau(m, n) + T(m)$ will hold, which implies, amongst other things, that the WIP $W_m(t)$ can never become negative. Fluid and traffic models replace the WIP W and the flux F by continuous functions and eliminate the dependence on individual parts, either by ad hoc assumptions, constitutive relations derived from stochastic queuing theory in quasi steady state (see c.f. [9]), or via asymptotic methodology borrowed from the theory of gas dynamics [4], [5], [14][15], [18]. In the simplest fluid models, the fluxes $F(m, t)$ in (4) are prescribed and the WIP's $W(m, t)$ are computed from F . Constraints have to be placed on the fluxes in order to guarantee non - negative WIP's. This is usually done in a linear programming framework [19].

The basic concept of the approach presented in this paper is somewhat different. Rather than artificially constraining the fluxes, we will derive a continuum model which contains as an input parameter a release rate μ , and in which the WIP's $W(m, t)$ will always be non - negative. The model is based on very simple assumptions, namely that each supplier functions as a single processor with a processing time T and a linear buffer queue in front of it. Based on this assumption, we derive, in a continuum limit, a conservation law of the form

$$\partial_t \rho + \partial_x \min\{\mu, \frac{W}{T}\} = 0, \quad (5)$$

where the artificial continuous variable x numbers the suppliers and the $\rho(x, t)$ denotes the product density over x , i.e. $W = \int \rho dx$ holds. If the number of parts considered is very large, then solving the conservation law (5) is obviously much more effective than to directly compute the $\tau(m, n)$.

This paper is organized as follows. In Section 2 we define the basic rule governing the transition times $\tau(m, n)$, modelling one supplier in the chain as a processor with a given throughput time and a linear buffer queue in front of it. We heuristically derive simplified formulas to compute the WIP density ρ and the flux from the transition times. These formulas are simpler than (2) and (3), in the sense that they depend only locally on the $\tau(m, n)$. This will allow us to derive simple constitutive relations for the flux and WIP density leading to the conservation law (5). However, with this simple constitutive relation, the conservation law (5) will only be satisfied approximately. In Section 3 we show that (5) is satisfied asymptotically in the limit for a large number of suppliers. The main difficulty here is that, as it turns out, the conservation law (5) will in general have only distributional solutions. $\rho(x, t)$ will develop δ - function concentrations, corresponding to bottlenecks in the

supply chain. We will resolve this problem by instead deriving the corresponding hyperbolic equation for the N-function U in (1). This will also allow us to numerically compute the distributional solutions of (5) in a reasonable way. The assumption of a large number of nodes in the supply chain is actually unreasonable for many applications. In Section 4 we remove this assumption by replacing one individual supplier with an arbitrary number of virtual supplier, allowing us to pass to a continuum limit in almost every situation. Section 5 is devoted to numerical experiments. We demonstrate the asymptotic validity of the continuum model on two examples, one with only a few nodes in the supply chain where we utilize the concept of virtual suppliers, and one example of a long supply chain with randomly generated processing times and capacities.

2 THE BASIC MODEL

In this section we first define the basic rules governing the supply chain. We then give a more or less heuristic reasoning for a formula which expresses the flux and the density of parts locally in time, i.e. dependent only on differences of neighboring transition times τ , in what is essentially a large time regime. With these local formulas we derive in Theorem T1 a constitutive relation which expresses the flux in terms of the density. We first present the basic model for a single node in the supply chain. We assume that the node consists of a processor which processes parts at a rate μ . In front of this 'machine' we assume a buffer queue, i.e. parts arrive at the end of the queue wait until they reach the front, and then are fed into the processor. We denote by a_n , $n = 0, 1, \dots$ the time part number n arrives at the end of the queue and by b_n the 'release time', i.e. the time part number n reaches the front of the queue and is fed into the processor. If the queue is full, the interval between two consecutive release times b_n will be given by the processing rate μ , i.e.

$$b_n = b_{n-1} + \frac{1}{\mu}$$

will hold as long as $a_n \leq b_{n-1} + \frac{1}{\mu}$ holds, meaning that part number n has already arrived when we want to feed it into the processor. If, on the other hand, the queue is empty, i.e. if at the desired release time $b_{n-1} + \frac{1}{\mu}$ part number n has not arrived yet at the end of the queue, then we wait for its arrival and then immediately feed it into the processor. So, $a_n > b_{n-1} + \frac{1}{\mu}$ will imply $b_n = a_n$. This gives altogether the relation

$$b_n = \max\left\{a_n, b_{n-1} + \frac{1}{\mu}\right\}. \tag{6}$$

We assume that the processor takes a time T to finish the part. and denote by $e_n = b_n + T$ the time the part leaves the processor (and enters the next queue). Inserting this relation into (6) gives

$$e_n = \max\left\{a_n + T, e_{n-1} + \frac{1}{\mu}\right\} \tag{7}$$

as the basic law relating the arrival times a_n to the exit times e_n . We now consider a chain of M suppliers S_0, \dots, S_{M-1} and denote with $\tau(m, n)$ the time part number n arrives at supplier S_m . Using the obvious change of notation $a_n \rightarrow \tau(m, n)$ and $e_n \rightarrow \tau(m+1, n)$ we obtain from (7)

$$\tau(m+1, n) = \max\left\{\tau(m, n) + T(m), \tau(m+1, n-1) + \frac{1}{\mu(m, n-1)}\right\}, \quad m = 0, \dots, M-1, \quad n \geq 1. \quad (8)$$

Here $T(m)$ denotes the processing time of processor S_m and we have made the processing rates μ time dependent, i.e. dependent on the part index n as well. The reason for the latter is that the release rates μ are used to control the supply chain. So, after part number $n-1$ has been fed into processor S_m we wait a time interval $\frac{1}{\mu(m, n-1)}$ before feeding in the next part. We assume that the processor belonging to the node S_m has a finite capacity $C(m)$, so

$$\mu(m, n) \leq C(m), \quad m = 0, \dots, M-1, \quad n \geq 0 \quad (9)$$

has to hold, but otherwise the μ 's can be chosen arbitrarily. The recursion (8) still needs initial and boundary conditions. They are of the form

$$\tau(0, n) = \tau^A(n), \quad n \geq 0, \quad \tau(m, 0) = \tau^I(m), \quad m = 0, \dots, M.$$

$\tau^A(n)$ simply denotes the arrival time of part n in the first processor of the chain. The interpretation of $\tau^I(m)$ is somewhat more subtle. Obviously, $\tau^I(m)$ denotes the time the first part has arrived at supplier S_m . So, $\tau^I(m+1) - \tau^I(m) - T(m)$ denotes the time the first part has waited in the buffer in front of the processor at S_m . Assuming a constant release rate μ in the past, $\mu(m, 0)[\tau^I(m+1) - \tau^I(m) - T(m)]$ would be the number of parts in the queue at the time part number 0 arrives. This, in a sense, records the history of what has happened in the system before the first part went through and determines the queue length at the initial time. This somewhat awkward definition is necessitated by the fact that, for an actual simulation, we have to start somewhere. This issue will be resolved once the problem is formulated in terms of an approximate conservation law.

The goal of this paper is to asymptotically replace (8) by a conservation law with a simple constitutive relation. The rest of this section is devoted to considerations of what the appropriate form of the constitutive relation $F = F(W)$ in (4) should be. In the next section we will then show that with this relation an equivalent of (5) holds in a weak sense. We start by re-defining the flux. First, we map (4) onto a grid in an artificial spatial variable x , called the 'Degree of Completion' (DOC). We define a mesh $0 = x_0 < \dots < x_M = X$ and replace $F(m, t)$ by $F(x_m, t)$. So, parts enter the supply chain at the DOC $x = 0$ and leave at the DOC $x = X$. Next, we observe that, for an arbitrary test function $\psi(t)$

$$\int_{\tau^I(m)}^{\infty} \psi(t) F(x_m, t) dt = \sum_{n=0}^{\infty} \psi(\tau(m, n))$$

holds. We re- write this into a Riemann sum for an integral as

$$\int_{\tau^I(m)}^{\infty} \psi(t)F(x_m, t) dt = \sum_{n=0}^{\infty} \psi(\tau(m, n))\Delta_n\tau(m, n)f(x_m, \tau(m, n)) , \quad (10)$$

where $\Delta_n\tau$ denotes the difference of $\tau(m, n)$ in the index n and the function $f(x, t)$ is given at $x = x_m$ and $t = \tau(m, n)$ as the reciprocal difference, i.e.

$$(a) \Delta_n\tau(m, n) := \tau(m, n+1) - \tau(m, n), \quad (b) f(x_m, \tau(m, n)) := \frac{1}{\Delta_n\tau(m, n)} \quad (11)$$

holds. On a time scale, where $\Delta_n\tau$ is small, (10) becomes

$$\int_{\tau^I(m)}^{\infty} \psi(t)F(x_m, t) dt \approx \int_{\tau^I(m)}^{\infty} \psi(t)f(x_m, t) dt .$$

So (11)(b) will be the definition of our approximate flux f , which is given on the grid $\tau(m, n)$ for $x = x_m$. To find an approximate expression for the density ρ of parts per unit DOC, we consider the case, when the arrival times τ would be distributed continuously, i.e. if they were given as a function $\tau(x, y)$. In this case (11)(b) would become $f(x, \tau(x, y)) = \frac{1}{\partial_y\tau(x, y)}$. The N- function $U(x, t)$, the antiderivative of the flux, would then satisfy the relations

$$(a) \frac{d}{dy}U(x, \tau(x, y)) = \partial_t U(x, \tau)\partial_y\tau = 1, \quad (b) \frac{d}{dx}U(x, \tau(x, y)) = \partial_x U(x, \tau) + \partial_t U(x, \tau)\partial_x\tau \quad (12)$$

Setting $\rho(x, t) = K(x) - \partial_x U(x, t)$ for an arbitrary function K , in analogy to (3), (12) becomes

$$\frac{d}{dx}U(x, \tau(x, y)) = K(x) - \rho(x, \tau) + f(x, \tau)\partial_x\tau .$$

Now (12)(a) implies that $\frac{d}{dx}U(x, \tau(x, y))$ is a function of the DOC variable x only, which we set equal to the arbitrarily chosen function $K(x)$. So, for a continuum $\tau(x, y)$ of arrival times, we set $f(x, \tau) = \frac{1}{\partial_y\tau}$ and $\rho(x, \tau) = \frac{\partial_x\tau}{\partial_y\tau}$. Direct calculus yields that, if so defined, ρ and f satisfy a conservation law of the form $\partial_t\rho + \partial_x f = 0$. Motivated by this, we define the approximate density ρ and the approximate flux f from the arrival times τ by

$$(a) f(x_m, \tau(m, n)) = \frac{1}{\Delta_n\tau(m, n)}, \quad m = 0, \dots, M, \quad n = 0, 1, \dots \quad (13)$$

$$(b) \rho(x_m, \tau(m+1, n)) = \frac{\Delta_m\tau(m, n+1)}{h_m\Delta_n\tau(m+1, n)}, \quad m = 0, \dots, M-1, \quad n = 0, 1, \dots,$$

$$(c) \Delta_m\tau(m, n) := \tau(m+1, n) - \tau(m, n), \quad h_m := x_{m+1} - x_m .$$

The density and flux defined by (13) are approximate in the sense that they will, as will be seen in Section 3, satisfy an approximate or discretized version of the conservation law. However, the definition (13) allows us to derive a simple constitutive relation of the form

$f = f(\rho)$. Under what circumstances ρ and f satisfy an approximate conservation law will be the subject of the next section. We have the following

Theorem T1: *Let the arrival times $\tau(m, n)$ satisfy the recursion (8). Let the approximate density ρ and flux f be defined by (13). Then the approximate flux can be written in terms of the approximate density via a constitutive relation of the form*

$$f(x_m, \tau(m, n)) = \phi_{mn}(\rho(x_{m-1}, \tau(m, n))), \quad m = 1, \dots, M, \quad n \geq 0,$$

with the flux function ϕ_m given by

$$\phi_{mn}(\rho) = \min\left\{\mu(m-1, n), \frac{h_{m-1}\rho}{T(m-1)}\right\}. \quad (14)$$

The proof of Theorem T1 is rather lengthy and therefore deferred to the Appendix.

The advantage of the approximative constitutive law (14) over the exact law given by (2) and (3) lies in the fact that it does not involve the transition times $\tau(m, n)$ anymore. The subject of the next two sections will be if, and in what sense, ρ and f will still satisfy a conservation law of the form $\partial_t \rho + \partial_x f = 0$.

3 ASYMPTOTIC VALIDITY OF THE CONSERVATION LAW

In this section we show that the approximate density ρ and flux f , defined by (13) satisfy, in a certain sense, a conservation law of the form $\partial_t \rho + \partial_x f = 0$ asymptotically. The asymptotic regime we consider is one for a large number of nodes in the supply chain and for large time scales. The assumption of a large number of nodes is to some extent artificial and will be removed in Section 4. As it turns out the limiting density ρ will in general not be a classical function but a distribution. We therefore show the asymptotic validity for the corresponding hyperbolic differential equation for the limiting N-curve U in (1).

3.1 SCALING AND DIMENSIONLESS FORMULATION

We define by T_0 the average processing time, i.e.

$$T_0 = \frac{1}{M} \sum_{m=0}^{M-1} T(m)$$

holds. So MT_0 would be the time for a part to be processed in the empty system, without waiting in any queue. This is chosen as the over all time scale, whereas we scale the individual

processing times $T(m)$ and release rates $\mu(m, n)$ by T_0 . Denoting scaled variables with the subindex s , this gives

$$\tau(m, n) = MT_0\tau_s(m, n), \quad T(m) = T_0T_s(x_m), \quad \mu(m, n) = \frac{\mu_s(x_m, \tau_s(m+1, n))}{T_0}. \quad (15)$$

We will consider a regime where $M \gg 1$ holds and set $\varepsilon = \frac{1}{M} \ll 1$ from here on. With this scaling the recursion (8) becomes

$$(a) \tau_s(m+1, n+1) = \max\left\{\tau_s(m, n+1) + \varepsilon T_s(x_m), \tau_s(m+1, n) + \frac{\varepsilon}{\mu_s(x_m, \tau_s(m+1, n))}\right\}, \quad (16)$$

$$m = 0, \dots, M-1, \quad n = 0, 1, \dots,$$

$$(b) \tau_s(0, n) = \tau_s^A(n), \quad n \geq 0 \quad \tau_s(m, 0) = \tau_s^I(m), \quad m = 0, \dots, M,$$

where we have scaled τ^A and τ^I in the same way as $\tau(m, n)$. Also we have made grid functions out of the throughput times and processing times. We assume that the differences between two consecutive arrival times τ are of the same order as the average processing time T_0 . This is reasonable since otherwise the total WIP would either go to zero or infinity. So we set

$$\Delta_n \tau(m, n) = \tau(m, n+1) - \tau(m, n) = T_0 \Delta_{ns} \tau_s(m, n),$$

$$\Delta_m \tau(m, n) = \tau(m+1, n) - \tau(m, n) = T_0 \Delta_{ms} \tau_s(m, n),$$

giving

$$\tau_s(m+1, n) = \tau_s(m, n) + \varepsilon \Delta_{ms} \tau_s(m, n), \quad \tau_s(m, n+1) = \tau_s(m, n) + \varepsilon \Delta_{ns} \tau_s(m, n).$$

In accordance with (13), we scale the density ρ and the flux f by

$$f(x, t) = \frac{1}{T_0} f_s(x, \frac{t}{MT_0}), \quad \rho(x, t) = \frac{M}{X} \rho_s(x, \frac{t}{MT_0}),$$

where X is the length of the DOC interval. This gives

$$(a) f_s(x_m, \tau_s(m, n)) = \frac{1}{\Delta_{ns} \tau_s(m, n)}, \quad m = 0, \dots, M, \quad n = 0, 1, \dots \quad (17)$$

$$(b) \rho_s(x_m, \tau_s(m+1, n)) = \frac{\varepsilon X \Delta_{ms} \tau_s(m, n+1)}{h_m \Delta_{ns} \tau_s(m+1, n)}, \quad m = 0, \dots, M-1, \quad n = 0, 1, \dots$$

as a definition for the scaled flux and density with $f_s, \rho_s = O(1)$. The scaled version of the constitutive relation (14) then reads

$$(a) f_s(x_m, \tau_s(m, n)) = \phi_s(x_{m-1}, \tau_s(m, n), \rho_s(x_{m-1}, \tau_s(m, n))) \quad (18)$$

$$(b) \phi_s(x_{m-1}, t, \rho_s) = \min\left\{\mu_s(x_{m-1}, t), \frac{h_{m-1} \rho_s}{\varepsilon X T_s(x_{m-1})}\right\}.$$

(18)(b) suggests a natural choice for the grid in x - direction, namely

$$h_m = \varepsilon X T_s(x_m) = \frac{X T(m)}{\sum_{m'=0}^{M-1} T(m')}, \quad m = 0, \dots, M-1, \quad (19)$$

which makes the propagation velocity in (18) equal to unity, i.e. we assign an interval in the DOC variable x to processor S_m which is proportional to its processing time. We will use this choice of the mesh from here on. Also, from now on we will drop the subscript s for simplicity.

3.2 INTERPOLATION AND WEAK FORMULATION

We now proceed to show the asymptotic validity of a conservation law in the limit $\varepsilon \rightarrow 0$. The goal is an initial boundary value problem for a conservation law of the form

$$\partial_t \rho + \partial_x f = 0, \quad f = \min\{\mu(x, t), \rho\}, \quad f(0, t) = f^A(t), \quad (20)$$

together with some initial condition. There are several complications in this approach.

- First, the resulting initial boundary value problem not be defined on a strip in (x, t) plane but on a domain bounded by $t > \tau^I$. This is more of a notational inconvenience, but impacts the definition of initial conditions.
- From the original definition of the problem we cannot assume any kind of smooth relation between two consecutive processors, i.e. we cannot assume that the throughput times $T(x_m)$ and release rates $\mu(x_m, t)$, defined by (15) will converge to a smooth function in the limit $\varepsilon \rightarrow 0$. The limiting problem therefore has to be defined weakly.
- The most severe problem is that the flux function f can become discontinuous. This can be seen from the following consideration. Since we cannot assume any smooth relation between consecutive processors, we have to allow for the possibility of a sharp drop in the release rate μ , i.e. $\mu(x_m, t) > \mu(x_{m+1}, t)$, which does not vanish in the limit $M \rightarrow \infty$. At this point we can easily construct a situation where $f(x_m, t) > \mu(x_{m+1}, t)$ holds. Since $f(x_{m+1}, t)$ is cut off by the \min -function, the limiting flux f will have to be discontinuous. Because mass still has to be conserved, this discontinuity has to be compensated by a δ -function concentration in the density ρ at this point. This corresponds to a bottleneck situation, where we feed into a processor at a rate higher than its capacity over a significant period of time. Consequently, the queues will grow, which is expressed as a δ -function in the limit. This situation will actually occur right in the beginning of the supply chain if the boundary flux $f^B(t)$ is chosen larger than the capacity of the first processor.

We deal with the above problem by re- defining our concept of a solution. Instead of deriving a conservation law for the density ρ we derive a hyperbolic equation for the limiting N -

function U in (2). We denote its approximation by u , set $\rho(x, t) = -\partial_x u(x, t)$, and integrate (20) once with respect to x . This gives

$$\partial_t u = \min\{\mu(x, t), -\partial_x u\}, \quad \lim_{x \rightarrow 0^-} u(x, t) = g^A(t), \quad \frac{d}{dt} g^A(t) = f^A(t) \quad (21)$$

Clearly, if the solution $u(x, t)$ is continuous and has a bounded x - derivative, we obtain a solution $\rho(x, t)$ of (20) by differentiating u with respect to x . However, the equation (21) allows for shock solutions which result in δ - functions in the variable ρ . Although the x - derivative of u in this case becomes unbounded, the flux will remain bounded because of the min - function. ($-\partial_x u = \rho$ will always be bounded from below by zero.) We will therefore show that, in the limit $\varepsilon \rightarrow 0$ the N-function u satisfies a hyperbolic problem of the form (21) weakly in x and t . To do so, we first have to define the variables given on the nonuniform and non-rectangular mesh in (x, t) for continuous arguments by piecewise constant interpolation. For a given gridpoint x_m we first interpolate the grid functions ρ and f defined by (17) in time direction by

$$(a) f_1(x_m, t) = f(x_m, \tau(m, n)), \quad \tau(m, n) \leq t < \tau(m, n+1), \quad m = 0, \dots, M-1, \quad n \geq 0 \quad (22)$$

$$(b) \rho_1(x_m, t) = \rho(x_m, \tau(m+1, n-1)), \quad \tau(m+1, n-1) \leq t < \tau(m+1, n), \quad m = 0, \dots, M-1, \quad n \geq 1,$$

$$(c) f^A(t) = \frac{1}{\Delta_n \tau^A(n)}, \quad \tau^A(n) \leq t < \tau^A(n+1).$$

Next we define the N-function $u(x_m, t)$ by

$$u_1(x_{m+1}, t) = u_1(x_m, t) - \frac{h_m}{X} \rho_1(x_m, t), \quad m = 0, \dots, M-1, \quad u_1(x_0, t) = \int_{\tau(0,0)}^t f^A(s) ds \quad (23)$$

Given the functions ϕ_1, u_1 which are now defined for continuous time and discrete space, we define the functions

$$(a) f_2(x, t) = f_1(x_{m+1}, t), \quad x_m \leq x < x_{m+1}, \quad m = 0, \dots, M-1 \quad (24)$$

$$(b) \tau_2^I(x) = \tau^I(m+1), \quad x_m \leq x < x_{m+1}, \quad m = 0, \dots, M-1$$

$$(c) u_2(x, t) = u_1(x_{m+1}, t), \quad x_m \leq x < x_{m+1}, \quad m = 0, \dots, M-1,$$

as functions of continuous space and time.

3.3 THE LIMIT $\varepsilon \rightarrow 0$

We can now show that the so defined interpolant u_2, f_2 satisfies a weak version of (21). We have

Theorem T2 *Given the scaled density and flux at the discrete points $x_m, \tau(m, n)$, as defined in (16). Let the piecewise constant interpolant u_2 and f_2 be defined as in (22). Let the scaled*

throughput times $T(x_m)$ stay uniformly bounded, i.e. $h_m = O(\varepsilon)$ holds uniformly in m . Assume finitely many bottlenecks for a finite amount of time, i.e. let $\Delta_m \tau(m, n)$ be bounded for $\varepsilon \rightarrow 0$ except for a certain number of nodes m and a finite number of parts n , which stays bounded as $\varepsilon \rightarrow 0$. Then, for $\varepsilon \rightarrow 0$ and $\max h_m \rightarrow 0$ the interpolated N- function and flux u_2, f_2 satisfy the initial boundary value problem

$$(a) \quad \partial_t u_2 = f_2, \quad t > \tau_2^I(x), \quad 0 < x < X, \quad (25)$$

$$(b) \quad u_2(x, \tau^I(x)) = 0, \quad \lim_{x \rightarrow 0-0} u_2(x, t) = \int_{\tau_2(0,0)}^t f^A(s) ds$$

in the limit $\varepsilon \rightarrow 0$, weakly in x and t .

The proof of Theorem T2 is deferred to the Appendix.

Remark: Theorem T2 establishes the asymptotic validity of the integrated conservation law (25)(a) for any N- curve u and any flux function f , derived from an arbitrary sequence τ via the definition (13) and the interpolation formulas (22) and (24). The constitutive relation $f_2 = \min\{\mu, -\partial_x u_2\}$ is a consequence of the recursion relation satisfied by the sequence $\{\tau(m, n)\}$ and, consequently, of Theorem T1.

Remark: In unscaled variables Theorem T2 implies that the density $\rho(x, t)$ can be approximately computed as $\rho = -\partial_x u$ where the unscaled N-function $u(x, t)$ is the solution of

$$\partial_t u = \min\{\mu_-, -\frac{X}{MT_0} \partial_x u\}, \quad 0 < x < X, \quad \lim_{x \rightarrow 0-0} u(x, t) = \int_{\tau(0,0)}^t f^A(s) ds \quad (26)$$

$$\mu_-(x, t) := \lim_{y \rightarrow x-0} \mu(y, t),$$

Remark: The assumptions of Theorem T2 state that the number of nodes in the supply chain is large, that the number of bottlenecks is small compared to the number of processors, and that each of the processing times is small compared to the over all throughput time, i.e. $T(m) \ll \sum_{m'=0}^{M-1} T(m')$ holds. At first glance, these assumptions might seem rather restrictive. We will remove these restrictions in the next section by introducing the concept of virtual processors, which will allow us to arbitrarily increase M .

4 VIRTUAL PROCESSORS

As pointed out in Section 3, the asymptotic validity of the differential equation (26) is only given for the case when the number M of processors is large and the each of the individual processing times $T(m)$ is small compared to the total processing time $MT_0 = \sum_{m=0}^{M-1} T(m)$. So, it excludes c.f. the situation where one processor takes up half of the over all processing time. In this section we will relax this restriction by introducing the concept of virtual

processors. The basic idea is that one processor with a processing time T and a release rate μ can be replaced by K virtual processors with the same release rate μ and processing times $\frac{T}{K}$. Thus we can make the total number of processors as large as we like, and the relative processing times as small as we like, by introducing enough virtual processors. The purpose of this section is to make this statement precise. Since, in doing so, we will keep the release rates μ constant but decrease the processing times T , eventual bottlenecks will only occur in the first virtual processor and the queues of the additional virtual processors will always remain empty. Given the recursion formula (8), we therefore derive a condition for queues being always empty.

Lemma L1 : *Given the recursion (8) for the arrival times $\tau(m, n)$, Let the arrival rate in node S_m be below the release rate μ , i.e. let*

$$\tau(m, n+1) - \tau(m, n) \geq \frac{1}{\mu(m, n)}, \quad n = 0, \dots, \quad (27)$$

hold. Let furthermore the queue be empty at the arrival of the first part, i.e. let

$$\tau(m, 1) + T(m) \geq \tau(m+1, 0) + \frac{1}{\mu(m, 0)} \quad (28)$$

hold. Then

$$\tau(m+1, n) = \tau(m, n) + T(m), \quad n = 1, \dots$$

holds.

Proof of Lemma L1: Define waiting time in the queue number m as $Q(m, n) = \tau(m+1, n) - \tau(m, n) - T(m)$. Inserting this into (8) gives

$$Q(m, n+1) = \max\left\{0, Q(m, n) + \tau(m, n) - \tau(m, n+1) + \frac{1}{\mu(m, n)}\right\}. \quad (29)$$

as a recursion for the waiting times $Q(m, n)$. In particular,

$$Q(m, 1) = \max\left\{0, \tau(m+1, 0) - T(m) - \tau(m, 1) + \frac{1}{\mu(m, 0)}\right\} = 0$$

holds because of (28). Because of (27), the term $\tau(m, n) - \tau(m, n+1) + \frac{1}{\mu(m, n)}$ is always non-positive and therefore the recursion (29) has the trivial solution $Q(m, n) = 0$ for $n \geq 1$. ■

Lemma L1 will provide the basic tool to split a processor into K virtual processors. The basic building block of the underlying idea is to split one processor into two. Without loss of generality we perform this split on the first node in the supply chain. We have

Lemma L2: *Let the flow of parts in processor S_0 be governed by*

$$\tau(1, n + 1) = \max\{\tau(0, n + 1) + T(0), \tau(1, n) + \frac{1}{\mu(0, n)}\}, \quad (30)$$

with $\tau(0, n)$, $n \geq 0$ and $\tau(1, 0)$ given and satisfying the compatibility condition $\tau(1, 0) \geq \tau(0, 0) + T(0)$. We replace (30) by two virtual nodes with the same processing rates and the same total throughput time; i.e.

$$(a) \hat{\tau}(1, n + 1) = \max\{\hat{\tau}(0, n + 1) + \hat{T}(0), \hat{\tau}(1, n) + \frac{1}{\mu(0, n)}\}, \quad (31)$$

$$(b) \hat{\tau}(2, n + 1) = \max\{\hat{\tau}(1, n + 1) + \hat{T}(1), \hat{\tau}(2, n) + \frac{1}{\mu(0, n)}\},$$

$$(c) \hat{\tau}(0, n) = \tau(0, n), \quad n = 0, 1, \dots \quad \hat{\tau}(1, 0) = \tau(1, 0) - \hat{T}(1), \quad \hat{\tau}(2, 0) = \tau(1, 0)$$

holds with $\hat{T}(0) + \hat{T}(1) = T(0)$. Then the system (31) produces the same outflux as the system (30), i.e. $\hat{\tau}(2, n) = \tau(1, n)$ $n \geq 0$ holds.

Proof of Lemma L2:

We show that the second virtual processor, i.e. the times $\hat{\tau}(1, n)$ and $\hat{\tau}(2, n)$, satisfy the assumptions of Lemma L1 . Because of (31)(a)

$$\hat{\tau}(1, n + 1) \geq \hat{\tau}(1, n) + \frac{1}{\mu(0, n)}, \quad n \geq 0$$

holds, giving (27). To show (28) we note that

$$\hat{\tau}(1, 1) + \hat{T}(1) \geq \hat{\tau}(1, 0) + \hat{T}(1) + \frac{1}{\mu(0, 0)} = \tau(1, 0) + \frac{1}{\mu(0, 0)} = \hat{\tau}(2, 0) + \frac{1}{\mu(0, 0)}$$

holds. Because of Lemma L1

$$\hat{\tau}(2, n) = \hat{\tau}(1, n) + \hat{T}(1), \quad n = 1, \dots \quad (32)$$

holds, and (32) trivially holds for $n = 0$ as well because of the initial condition (31)(c). We now eliminate $\hat{\tau}(1, n)$ by inserting $\hat{\tau}(1, n) = \hat{\tau}(2, n) - \hat{T}(1)$ into (31)(a) and obtain

$$\hat{\tau}(2, n + 1) = \max\{\hat{\tau}(0, n + 1) + T, \hat{\tau}(2, n) + \frac{1}{\mu(0, n)}\},$$

i.e. $\hat{\tau}(0, n), \hat{\tau}(2, n)$ satisfy the same difference equation and initial and boundary conditions as $\tau(0, n), \tau(1, n)$. ■

By repeatedly using Lemma L2, we immediately obtain, as a corollary

Theorem T3: *Let the first processor S_0 in the chain be governed by (8). If we replace the single processor by K virtual processors with the same processing rates and the same total throughput time, i.e. by*

$$\hat{\tau}(m+1, n+1) = \max\left\{\hat{\tau}(m, n+1) + \frac{T(0)}{K}, \hat{\tau}(m+1, n) + \frac{1}{\mu(m, n)}\right\}, \quad m = 0, \dots, K-1,$$

$$\hat{\tau}(0, n) = \tau(0, n), \quad \hat{\tau}(m, 0) = \tau(1, 0) - \left(1 - \frac{m}{K}\right)T(0), \quad m = 1, \dots, K,$$

then we obtain the same outflux, i.e.

$$\hat{\tau}(K, n) = \tau(1, n), \quad n \geq 0$$

holds.

So, in order to create the conditions appropriate for the application of Theorem T2, we would proceed as follows.

1. Given the processing times $T(m)$, $m = 0, \dots, M-1$, cut each of the processors S_m , $m = 0, \dots, M-1$ into $K(m)$ virtual processors, such that $T_1 = \frac{T}{K}$ is roughly equidistributed, giving $M_1 = \sum_{m=0}^{M-1} K(m)$ virtual processors.
2. If the number of virtual processors M_1 is still too small for the asymptotic regime in Theorem T2 to be valid, cut each of the virtual processors into additional L sub-processors to arrive at $M_2 = LM_1$ total processors.

Clearly, the number M_2 of virtual processors can be made as large as we like. No additional bottlenecks are created by this procedure since μ remains constant within each virtual processor belonging to one real processor and a bottleneck only can occur if there is a drop in the processing rate μ . So, the number of bottlenecks remains finite as $M_2 \rightarrow \infty$. There is, however, a limit to this process since we have used the average processing time T_0 also to scale the release rates μ in (15). So, sending $T_0 \rightarrow 0$ would result in the scaled release rates μ , and therefore also the fluxes, going to zero. To obtain a reasonable limiting problem we should choose M_2 and T_0 in such a way that $T_0 C_0 = O(1)$ holds, where C_0 is some characteristic value for the capacities, the bounds on μ in (9). So, with the introduction of virtual nodes in the supply chain, the results of Section 3 really apply to the case when $C_0 M_2 T_0 = C_0 \sum_{m=0}^{M-1} T(m) \gg 1$ holds. For a stochastic queuing model in steady state, this is according to Little's law (see c.f. [13]), a measure of the number of parts in the system. So the hyperbolic equation (25) in Theorem T2 is asymptotically valid for a large number of individual parts, i.e. precisely in situations where continuum models are computationally more efficient than discrete event simulators.

5 NUMERICAL EXPERIMENTS

In this section we conduct two numerical experiments to verify Theorem T2 by comparing the solution of the hyperbolic problem (26) with the direct solution of the recursion (8) for the transition times τ . In both cases we solve (8), compute the WIP W the N-curve U and the flux F according to (1), (2) and (3), and compare it to ρ , u and f computed from the solution of the hyperbolic equation (26). The hyperbolic problem for the approximate N-curve u is solved via a standard finite difference scheme of the form

$$(a) \quad u(x_m, t_{n+1}) = u(x_m, t_n) + \Delta t f(x_m, t_n), \quad m = 0, \dots, M, \quad \Delta t = t_{n+1} - t_n \quad (33)$$

$$(b) \quad f(x_m, t) = \begin{pmatrix} \min\{\mu(x_{m-1}, t), -\frac{X}{MT_0 \Delta x_{m-1}} [u(x_m, t) - u(x_{m-1}, t)]\} & m = 1, \dots, M \\ f^A(t_n) & m = 0 \end{pmatrix}.$$

For simplicity, we use constant time steps satisfying a CFL condition of the form $\Delta t \leq \frac{MT_0}{X} \min\{\Delta x_m\}$. If the spatial meshsizes Δx_m of the discretization of the conservation law are chosen equal to the h_m in (19), i.e. if we assign one gridpoint to one node in the supply chain, this would give $\Delta x_m = \frac{XT(m)}{MT_0}$, $m = 0, \dots, M-1$ and a CFL condition $\Delta t \leq \min\{T(m)\}$. While this seems a natural choice it is not a necessary one. In particular, in regions where the release rates μ vary slowly, a larger spatial meshsize might be appropriate. Regardless of the choice of the spatial mesh the node S_m in the supply chain will always occupy an interval of length h_m . The influx f^A is computed according (11)(b) by

$$f^A(\tau^A(n)) = \frac{1}{\tau^A(n+1) - \tau^A(n)}$$

and piecewise linear interpolation. Note, that the discretization (33) is equivalent to discretizing the conservation law directly, i.e. if we define the discretized density ρ by

$$\rho(x_m, t_n) = -\frac{u(x_{m+1}, t_n) - u(x_m, t_n)}{\Delta x_m}, \quad m = 0, \dots, M-1$$

the discrete equation (33) becomes

$$(a) \quad \rho(x_m, t_{n+1}) = \rho(x_m, t_n) - \frac{\Delta t}{\Delta x_m} [f(x_{m+1}, t_n) - f(x_m, t_n)], \quad m = 0, \dots, M-1 \quad (34)$$

$$(b) \quad f(x_m, t) = \begin{pmatrix} \min\{\mu(x_{m-1}, t), \frac{X\rho(x_{m-1}, t)}{MT_0}\} & m = 1, \dots, M \\ f^A(t_n) & m = 0 \end{pmatrix}.$$

So, the discretization (33) is equivalent to directly discretizing the conservation law for the density ρ , ignoring the issue of distributional solutions. Of course u in (33) will still be discontinuous at bottlenecks and ρ in (34) will grow like $\frac{1}{\Delta x}$ at these gridpoints. The discretization (34) represents only the simplest first order upwinding scheme for the hyperbolic conservation law. One could of course solve the hyperbolic problem (26) by more sophisticated high resolution methods on a correspondingly coarser mesh. Since this paper is concerned with

the model per se, we felt that using a higher order method would somehow cloud the issue of model properties by introducing the artifacts of the numerical method.

In the first example we consider a supply chain of three suppliers with throughput times $T(0) = 1, T(1) = 3, T(2) = 1$ time units and capacities $C(0) = 15, C(1) = 10, C(2) = 15$ parts per time unit. Setting the characteristic value for the capacity $C_0 = 10$, this gives a value of $C_0MT_0 = 50 \gg 1$ for the average number of parts in steady state. Thus, we can create the regime of theorem T2 by introducing virtual processors according to Section 4. We split the nodes S_0 and S_2 into ten virtual nodes each with capacities of 15 parts per unit time and node S_1 into 30 virtual nodes with capacities of 10 parts per unit time. All of the 50 virtual nodes now have a throughput time of 0.1 time units and, setting the length X of the DOC interval equal to unity, the original suppliers will occupy the intervals $[0, 0.2], [0.2, 0.8], [0.8, 1]$. We simply set the release rates μ equal to the capacities, giving

$$\mu(x, t) = \begin{pmatrix} 15 & \text{for } 0 < x < 0.2 \\ 10 & \text{for } 0.2 < x < 0.8 \\ 15 & \text{for } 0.8 < x < 1 \end{pmatrix} .$$

We expect two possible bottlenecks, namely at $x = 0.2$, where the capacity drops, and possibly at $x = 0$ if the influx exceeds 15 parts per unit time. We first solve the recursion (8) for the transition times τ , starting with all empty queues, i.e. $\tau^I(m+1) - \tau^I(m) = T(m) = 0.1$ holds, and set $\tau^I(M) = 0$. We compute the arrival times randomly according to $\tau^A(n+1) - \tau^A(n) = \frac{1}{f^A(\tau^A(n))}$, $\tau^A(0) = \tau^I(0)$. To study the development of bottlenecks, we choose a function $f^A(t)$ as the influx rate, which is first below the minimum capacity $C(1) = 10$, then between the minimum and the maximum capacity 10 and 15, then above the maximum capacity, and finally drops back to its original value. We add a random perturbation to a piecewise constant function. The influx rate f^A is shown in Figure 1. We compute fluxes and densities from the recursion (8) and the discretized conservation law (34). Figure 2 shows the corresponding WIP and Figure 3 shows the outflux of each node in the supply chain. The dots are computed from a time averaging of the solution of the recursion (8) for the transition times τ and the solid line is computed from the conservation law (34). So, the WIP of node S_0 in the chain is computed as $\int_0^{0.2} \rho(x, t) dx$.

Figures 4 5 and 6 depict the solution of the hyperbolic problem. Figure 4 shows the antiderivative of the density ρ in the DOC direction, i.e. $-u(x, t) + u(0, t)$, and Figure 5 shows the flux. As expected, we see bottlenecks, i.e. discontinuities, developing and vanishing again at $x = 0$ and $x = 0.2$. As long as the nodes in the supply chain work below capacity, i.e. as long as the governing equation is $\partial_t \rho + \frac{X}{MT_0} \partial_x \rho = 0$ we see the propagation of the fluctuations in the influx f^A through the system. As soon as the nodes go into saturation, i.e. as soon as $\partial_t \rho + \partial_x \mu$ holds, the solution becomes constant but develops discontinuities at the bottlenecks. Figure 6 shows the density ρ , which develops concentrations at $x = 0$ and $x = 0.2$, on a logarithmic scale.

As a second example, we consider a 'long' supply chain with unstructured throughput times

and capacities. We choose $M = 80$ and choose 80 random throughput times between $T = 1$ and $T = 5$ time units. For simplicity, we set $C(m) = \mu(m) = \frac{1}{T(m)}$, $m = 0, \dots, M - 1$. So each processor handles only one part per unit time and we use the maximally possible release rates μ . Figure 7 shows the corresponding mesh in the DOC variable $0 \leq x \leq X = 1$ and the capacities. So the meshsizes h_m are according to (19) randomly distributed. All the assumptions of Theorem T2 are satisfied, except that we cannot guarantee a relatively small number of bottlenecks, since the release rates μ now have an arbitrary number of significant drops. We choose $\Delta x_m = h_m$, i.e. we assign precisely one gridpoint per node for the numerical solution of the partial differential equation. Again, this goes beyond standard convergence theory, since we do not resolve the rapidly varying function $\mu(x)$ in the continuous formulation. Figure 8 shows the influx and the outflux of the last node in the supply chain. Again, the dots denote the time averaged results computed from the recursion formula (8). The influx is chosen at and below the minimum capacity $C_{min} = 0.2$, with a spike at $t \approx 1500$. Figure 9 shows the corresponding total WIP of the whole supply chain. We observe almost perfect agreement although we have not resolved the release rate function $\mu(x)$ on the computational mesh. Figure 10 shows the density ρ on a logarithmic scale. We see the development of six bottlenecks. So, although the relationship between capacities of neighboring processors is completely random, the supply chain organizes itself to produce only a few bottlenecks and the assumptions of Theorem T2 are still satisfied.

6 CONCLUSIONS

We have derived a partial differential equation modelling a supply chain of arbitrary length with a large number of parts. Other than in similar approaches, this model is not based on some quasi steady state assumptions about the stochastic behavior of the involved queues, but rather on a simple deterministic rule for releasing parts from the buffer queues into the processors. The presented model incorporates the concept of the capacity of a processor in a natural way in a transient setting, while models based on queuing theory have to achieve this through a relation between throughput time and Work in Progress which is somehow extrapolated from the steady state situation. The model contains a distributed parameter (the release rates), which is constrained by the capacities, and can be used to control the behavior of the supply chain. It can be expected, that relatively simple rules can be found, governing these release rates which guarantee a certain behavior of the supply chain, c.f. the avoidance of bottlenecks.

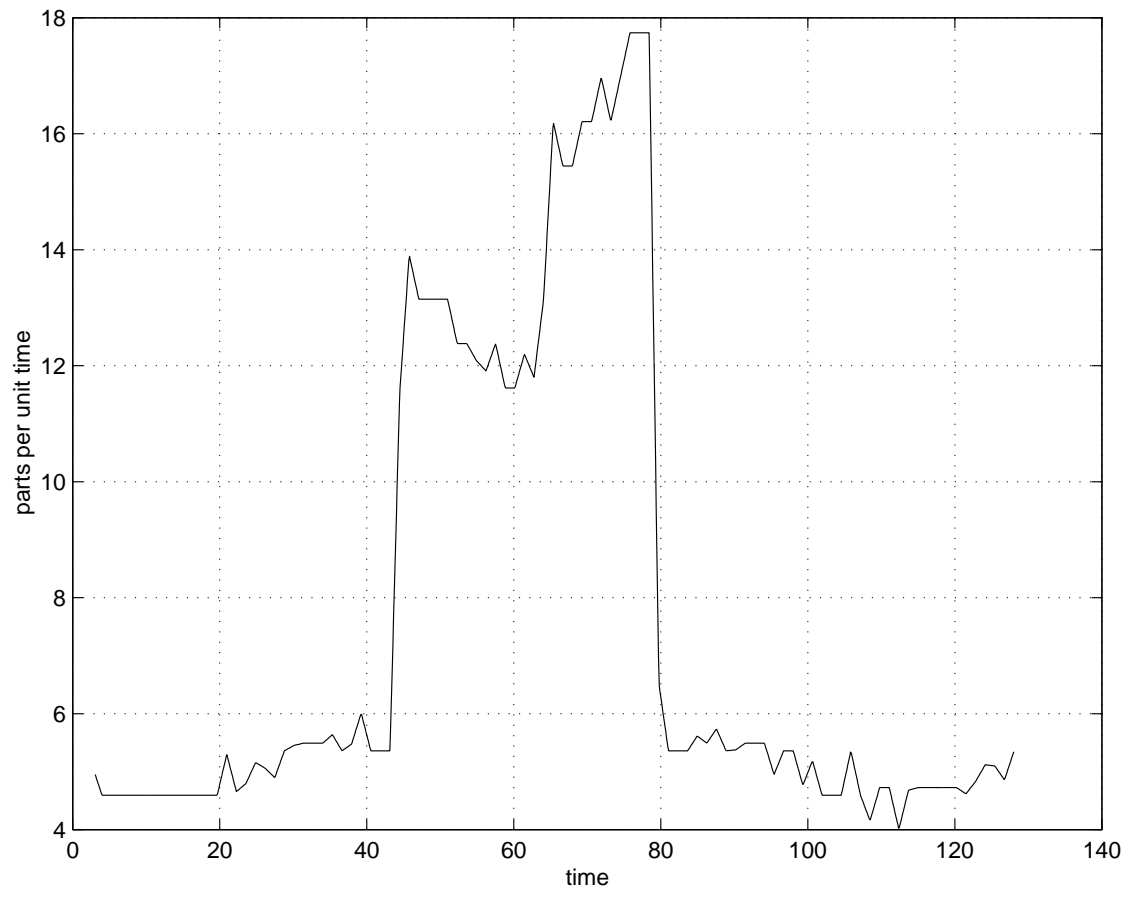


Figure 1: Influx.

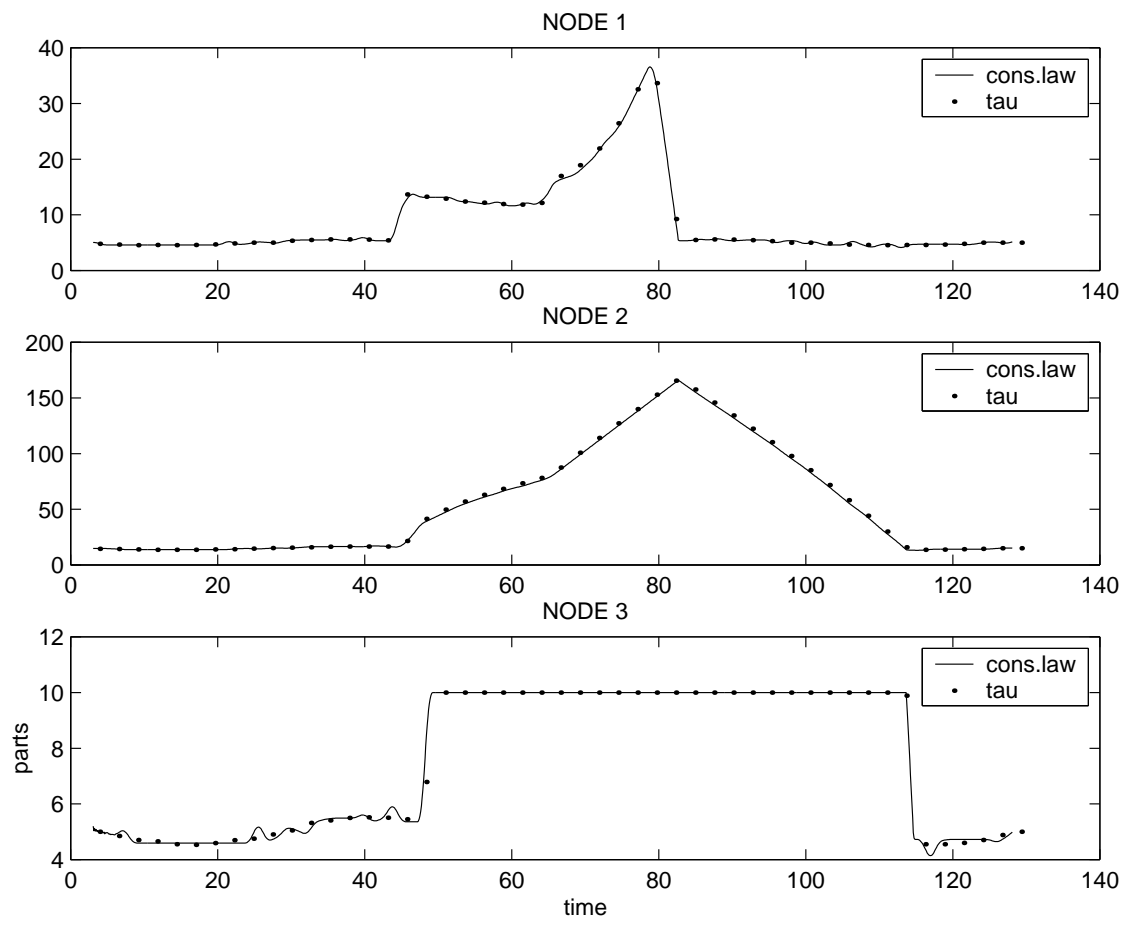


Figure 2: Work in Progress

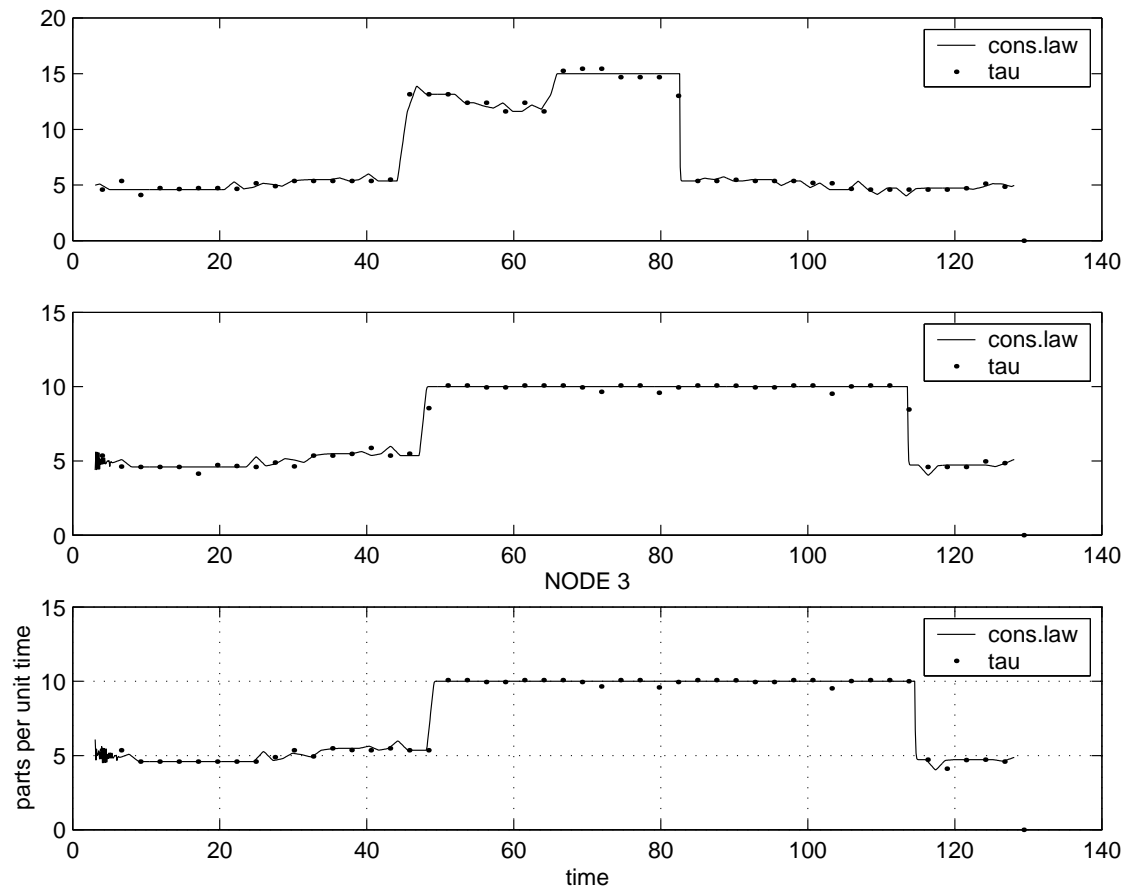


Figure 3: Outflux.

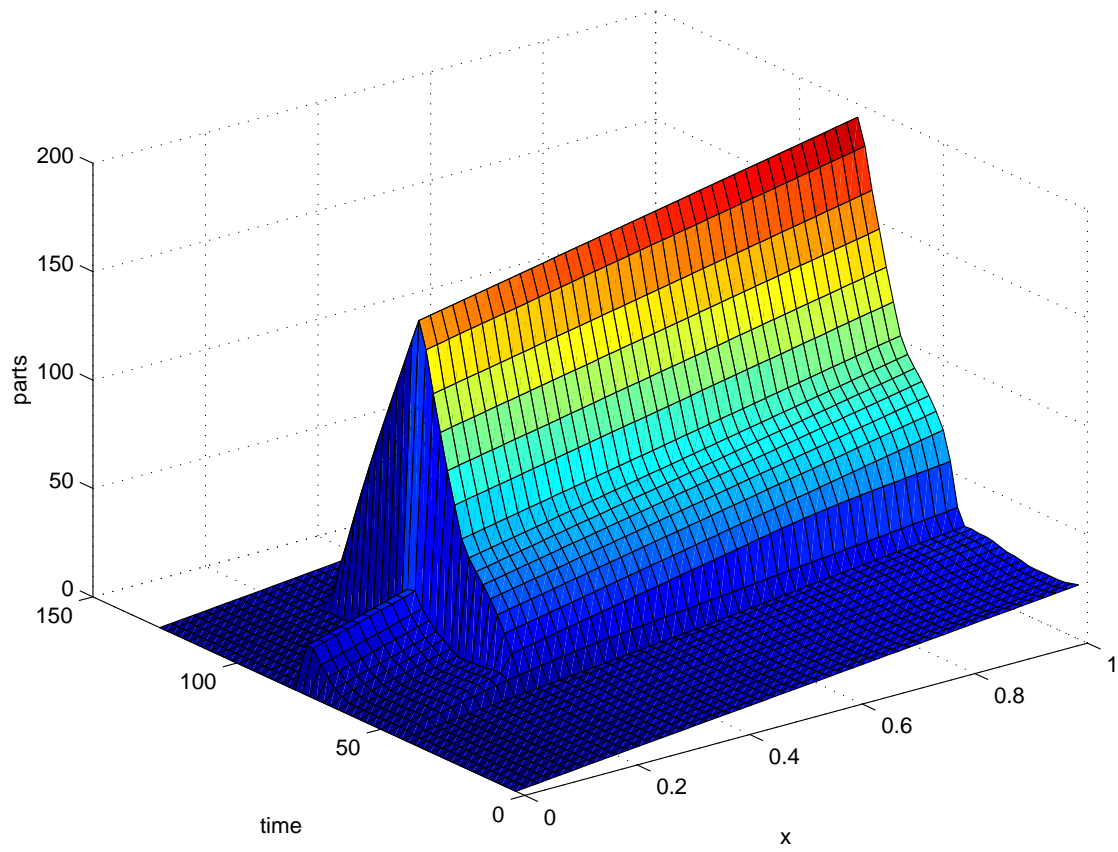


Figure 4: N-Curve: $u(0,t)-u(x,t)$.

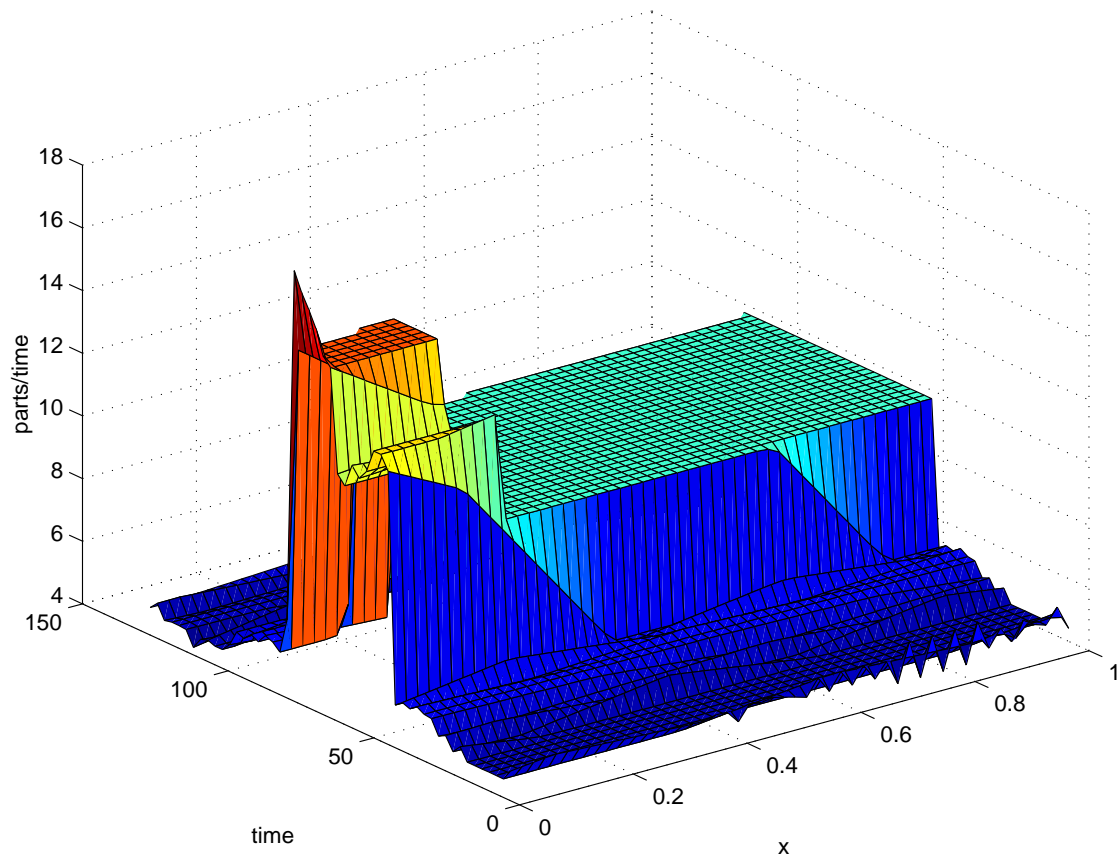


Figure 5: Flux f .

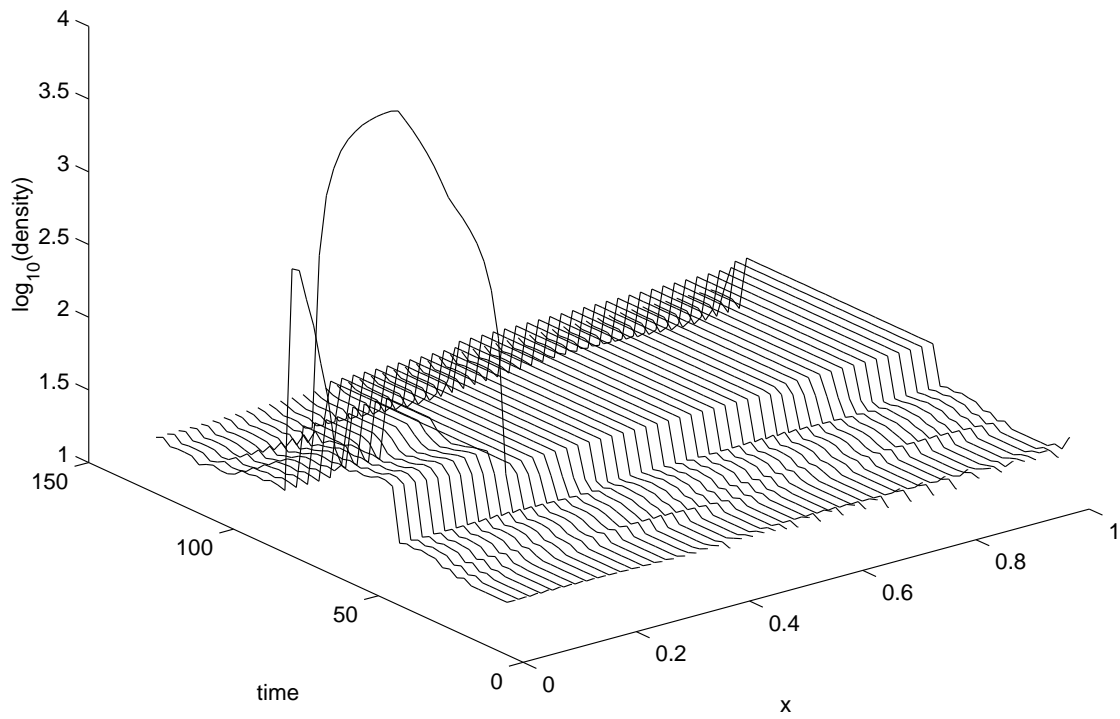


Figure 6: Density ρ .

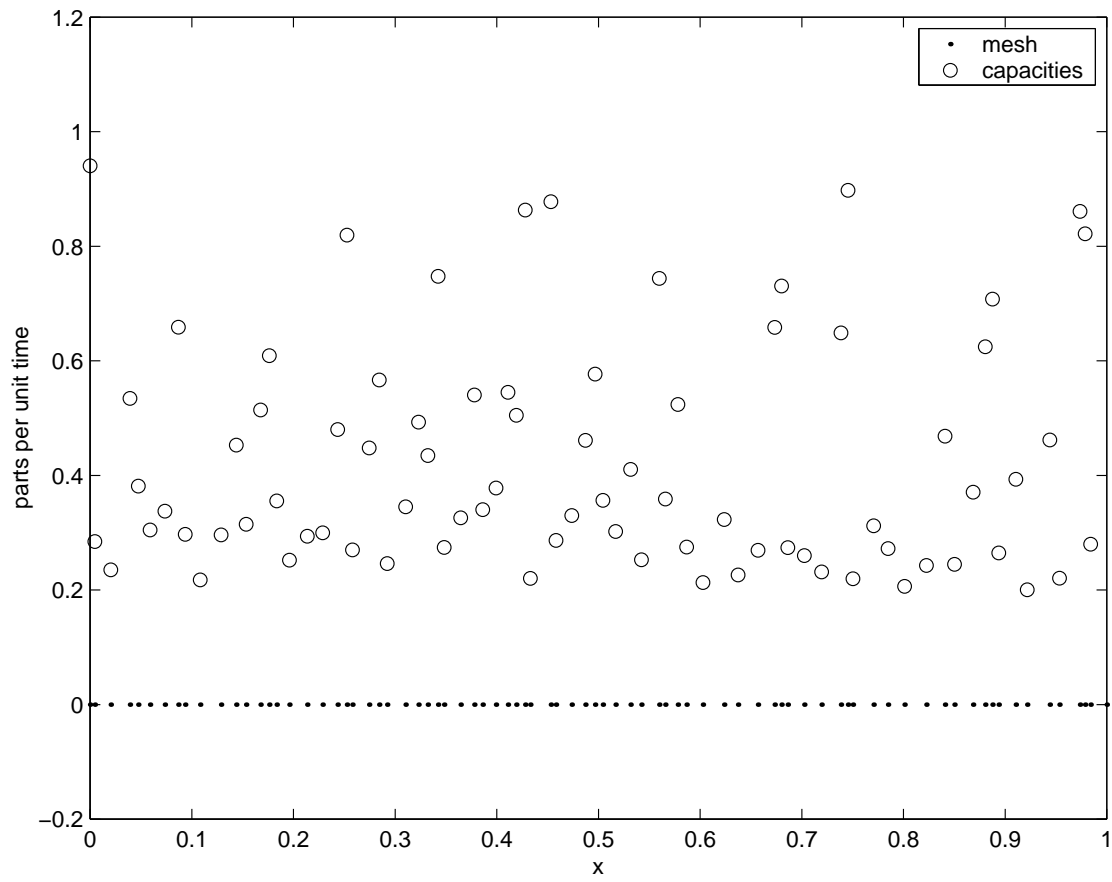


Figure 7: Mesh and Capacities.

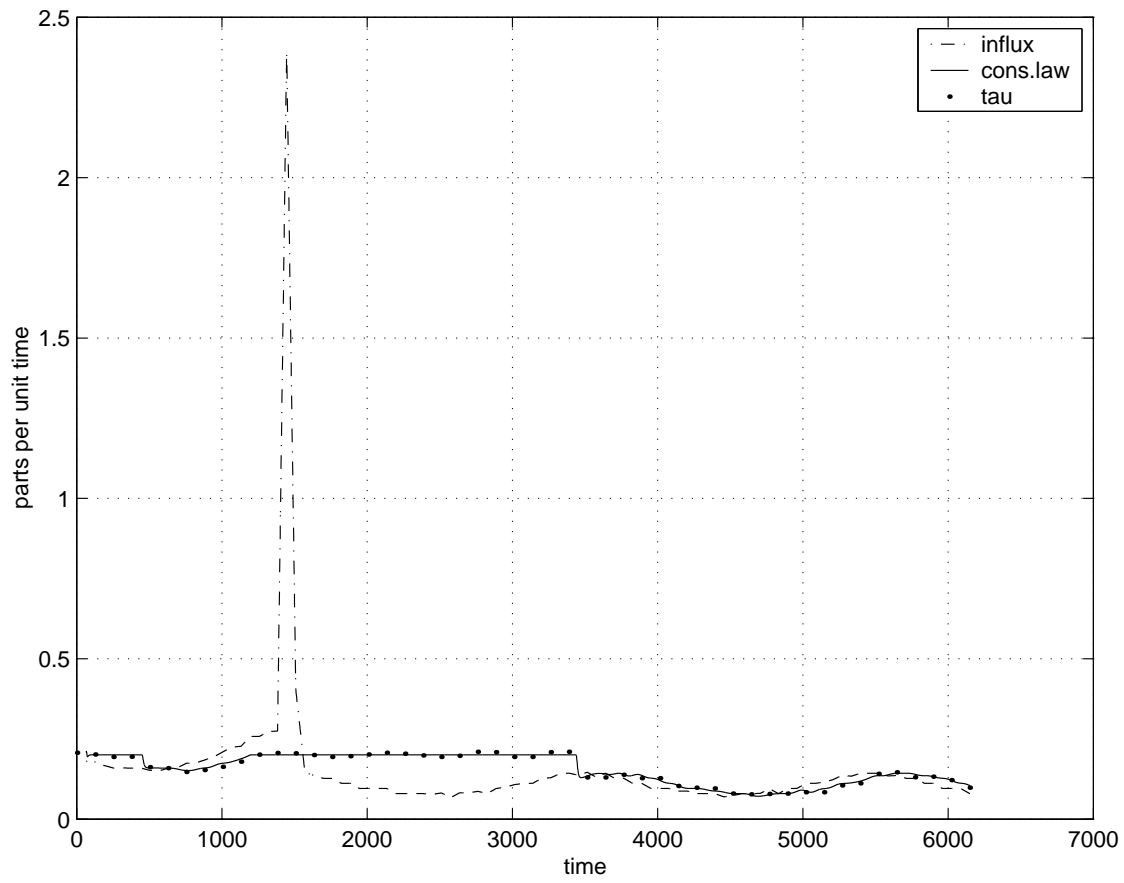


Figure 8: Influx and Outflux.

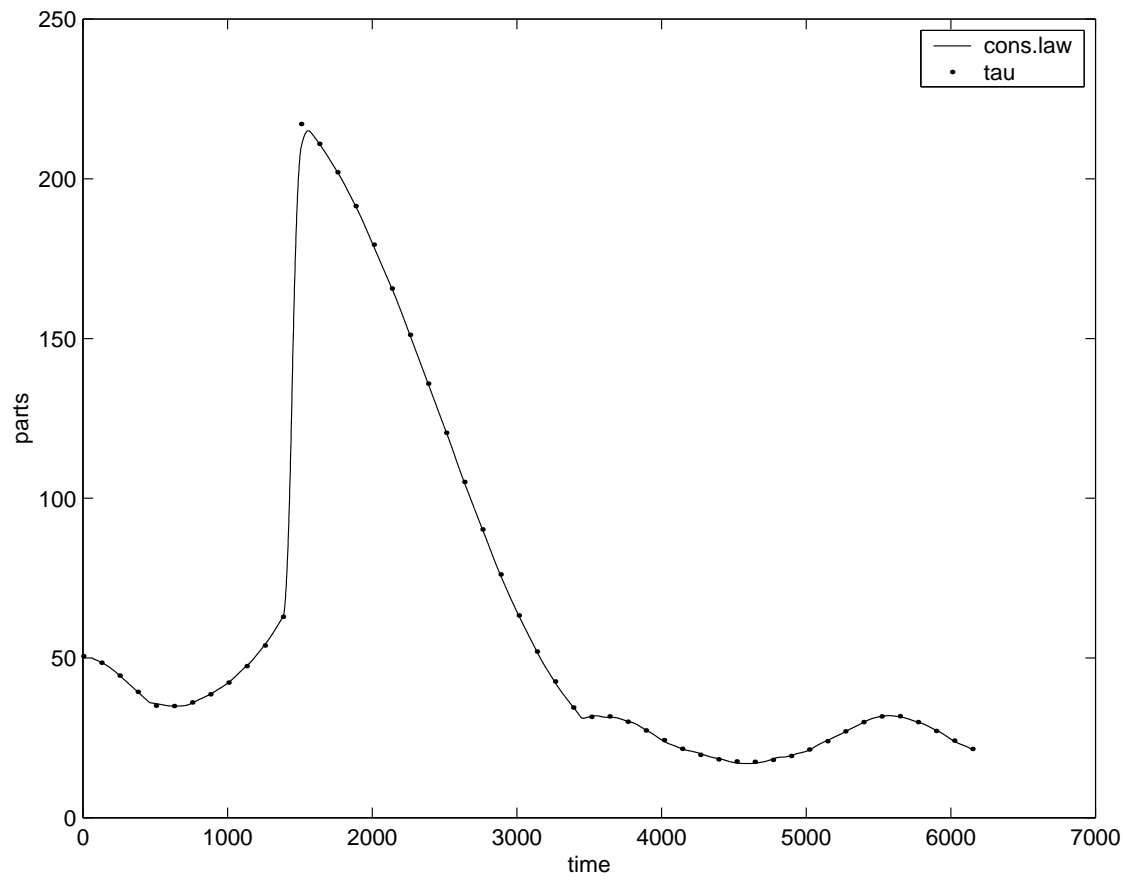


Figure 9: Total Work in Progress.

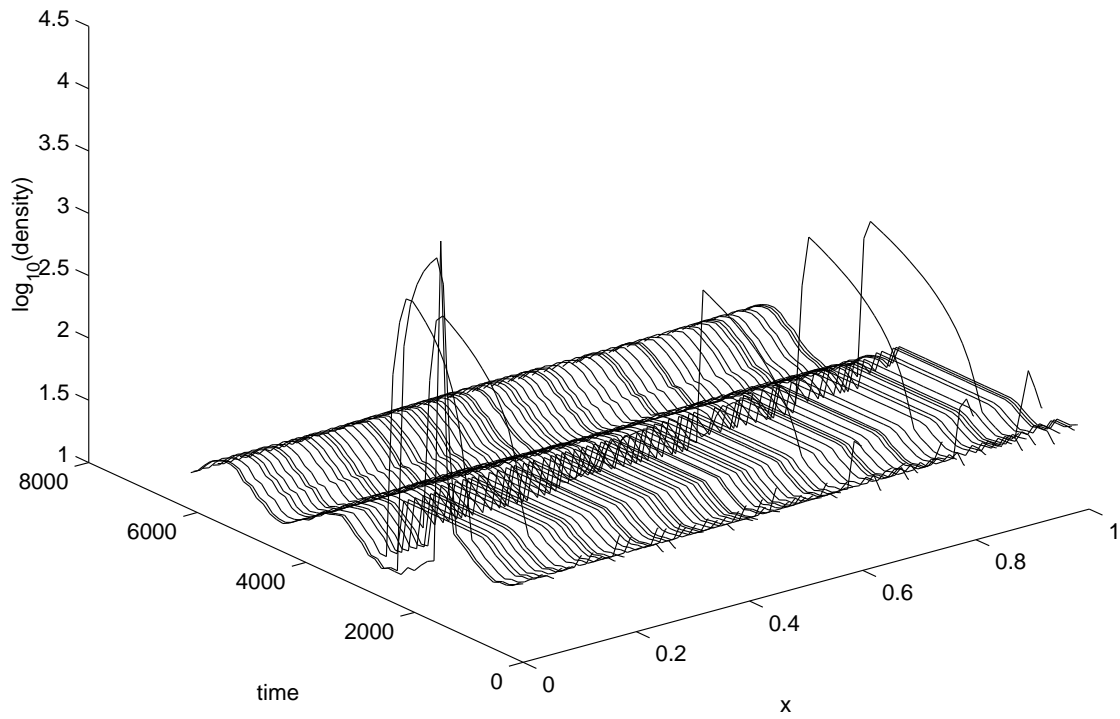


Figure 10: Part density ρ .

7 APPENDIX

Proof of Theorem T1:

We first re- write (8). Defining

$$\Delta_n \tau(m, n) := \tau(m, n + 1) - \tau(m, n), \quad \Delta_m \tau(m, n) := \tau(m + 1, n) - \tau(m, n),$$

(8) can be written as

$$0 = \min\left\{\Delta_m \tau(m - 1, n + 1) - T(m - 1), \Delta_n \tau(m, n) - \frac{1}{\mu(m - 1, n)}\right\}.$$

Using the definition (13) of ρ and f , this is equivalent to

$$0 = \min\left\{\frac{h_{m-1}\rho(x_{m-1}, \tau(m, n))}{f(x_m, \tau(m, n))} - T(m - 1), \frac{1}{f(x_m, \tau(m, n))} - \frac{1}{\mu(m - 1, n)}\right\}.$$

To simplify the notation, we will drop the indices m and n and write the above as $\min\{\frac{h\rho}{f} - T, \frac{1}{f} - \frac{1}{\mu}\} = 0$. Furthermore, we will write this relation in the variable $z = \frac{1}{f}$. So we have to invert the function $\alpha(z, \rho)$, given by

$$y = \alpha(z, \rho) = \min\left\{h\rho z - T, z - \frac{1}{\mu}\right\}$$

as a function of z for any given parameter ρ , i.e. find a function $\beta(y, \rho)$ satisfying

$$y = \alpha(z, \rho) \quad \iff \quad z = \beta(y, \rho).$$

If this is possible, then f is given in terms of ρ as $f = \frac{1}{\beta(0, \rho)} = \phi(\rho)$. There are two different cases to consider, namely the case $0 < h\rho < 1$ and the case $h\rho \geq 1$:

Case 1: $h\rho \geq 1$. In this case α is piecewise defined as

$$y = \alpha(z, \rho) = \begin{cases} zh\rho - T & \text{for } z < z_0 = \frac{T - \frac{1}{\mu}}{h\rho - 1} \\ z - \frac{1}{\mu} & \text{for } z > z_0 \end{cases}. \quad (35)$$

(In the case $h\rho = 1$ the second case in (35) simply never occurs.) This monotonically increasing and piecewise linear function of z can be inverted as

$$z = \beta(y, \rho) = \begin{cases} \frac{y+T}{h\rho} & \text{for } y < y_0 := z_0 - \frac{1}{\mu} \\ y + \frac{1}{\mu} & \text{for } y > y_0 \end{cases}.$$

Evaluating β at $y = 0$ gives

$$\beta(0, \rho) = \begin{cases} \frac{T}{h\rho} & \text{for } 0 < y_0 = \frac{T - \frac{1}{\mu}}{h\rho - 1} \\ \frac{1}{\mu} & \text{for } 0 > y_0 \end{cases} = \begin{cases} \frac{T}{h\rho} & \text{for } \frac{1}{\mu} < \frac{T}{h\rho} \\ \frac{1}{\mu} & \text{for } \frac{1}{\mu} > \frac{T}{h\rho} \end{cases} = \max\left\{\frac{1}{\mu}, \frac{T}{h\rho}\right\}$$

Case 2: $0 < h\rho < 1$. In this case, we proceed in the same way obtaining the piecewise linear definition

$$y = \alpha(z, \rho) = \begin{cases} z - \frac{1}{\mu} & \text{for } z < z_0 = \frac{T - \frac{1}{\mu}}{h\rho - 1} \\ zh\rho - T & \text{for } z > z_0 \end{cases}$$

for the function α . Note, that the ranges for the linear pieces of α are now switched. Inverting α gives

$$z = \beta(y, \rho) = \begin{cases} y + \frac{1}{\mu} & \text{for } y < y_0 := z_0 - \frac{1}{\mu} \\ \frac{y+T}{h\rho} & \text{for } y > y_0 \end{cases}$$

and evaluating β at $y = 0$ gives

$$\beta(0, \rho) = \begin{pmatrix} \frac{1}{\mu} & \text{for } 0 < y_0 = \frac{T - Dh\rho}{h\rho - 1} \\ \frac{T}{h\rho} & \text{for } 0 > y_0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\mu} & \text{for } \frac{1}{\mu} > \frac{T}{h\rho} \\ \frac{T}{h\rho} & \text{for } \frac{1}{\mu} < \frac{T}{h\rho} \end{pmatrix} = \max\left\{\frac{1}{\mu}, \frac{T}{h\rho}\right\}$$

So, in both cases we obtain the same value for the inverse β evaluated at $y = 0$. Setting $\phi(\rho) = \frac{1}{\beta(0, \rho)}$ gives (14). ■

Proof of Theorem T2

In order to avoid dealing with the boundary conditions we extend the definition of the variables ρ, f, u, τ onto the whole real line $x \in \mathbb{R}$. We define

$$x_m = -mh_0 \text{ for } m \leq -1, \quad x_m = X + (m - M)h_{M-1} \text{ for } m \geq M + 1 .$$

Next we extend the values of the arrival times τ for $m < 0$ and $m > M$. We set

$$\begin{aligned} \tau^I(m) &= \tau^I(0) + mh_0, \quad m < 0 \quad \tau^I(m) = \tau^I(M) + (m - M)h_{M-1}, \quad m > M , \\ \tau(m, n + 1) &= \tau(m, n) + \varepsilon \Delta_n \tau^A(n), \quad m \leq 0, \quad n \geq 0, \\ \tau(m, n + 1) &= \tau(m, n) + \varepsilon \Delta_n \tau(M, n), \quad m > M, \quad n \geq 0 . \end{aligned}$$

With this definition the values of the flux function f_1 in (22) satisfy

$$f(x_m, \tau(m, n)) = \frac{1}{\Delta_n \tau(m, n)} = f^A(\tau(m, n)), \quad m < 0, \quad n \geq 0 .$$

and the corresponding interpolant f_2 in the continuous x - variable satisfies

$$\lim_{x \rightarrow 0-0} f_2(x, t) = f^A(t) . \tag{36}$$

Now we consider a compactly supported test function $\psi(x, t)$ and its discrete antiderivative Ψ given by

$$\Psi(x_m, t) = \sum_{m'=-\infty}^{m-1} h_{m'} \psi(x_{m'}, t), \quad \Psi(x_{m+1}, t) - \Psi(x_m, t) = h_m \psi(x_m, t) . \tag{37}$$

Since ψ and Ψ are compactly supported in the time direction we have for any fixed index n the trivial equality

$$\sum_{m=-\infty}^{\infty} [\Psi(x_{m+1}, \tau(m+1, n)) - \Psi(x_m, \tau(m, n))] = 0, \quad \forall n,$$

which we sum over the index n and multiply by ε , giving

$$0 = \varepsilon \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} [\Psi(x_{m+1}, \tau(m+1, n)) - \Psi(x_m, \tau(m, n))] = A - B,$$

and A and B are defined by

$$(a) \quad A = \varepsilon \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} [\Psi(x_{m+1}, \tau(m+1, n)) - \Psi(x_m, \tau(m+1, n))] = \varepsilon \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} h_m \psi(x_m, \tau(m+1, n)) \quad (38)$$

$$(b) \quad B = \varepsilon \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} [\Psi(x_m, \tau(m, n)) - \Psi(x_m, \tau(m+1, n))].$$

We first estimate the spatial difference A . From the definition of the interpolant f_1 and the definition (17) of f , we have

$$\int_{\tau(m, n)}^{\tau(m, n+1)} f_1(x_m, t) = \varepsilon \Delta_n \tau(m, n) f(x_m, \tau(m, n)) = \varepsilon.$$

Inserting this into (38)(a) gives

$$\begin{aligned} A &= \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} h_m \psi(x_m, \tau(m+1, n)) \int_{\tau(m+1, n)}^{\tau(m+1, n+1)} f_1(x_{m+1}, t) dt = \\ &= \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} h_m \int_{\tau(m+1, n)}^{\tau(m+1, n+1)} \psi(x_m, t) f_1(x_{m+1}, t) dt + O(\varepsilon) = \\ &= \sum_{m=-\infty}^{\infty} h_m \int_{\tau^I(m+1)}^{\infty} \psi(x_m, t) f_1(x_{m+1}, t) dt + O(\varepsilon), \end{aligned}$$

where we have committed an $O(\varepsilon)$ error by taking the test function ψ inside the integral. Because of the definition (24) of the interpolant f_2 in the spatial direction, the term $h_m f_1(x_{m+1}, t)$ can be written as an integral with respect to x giving

$$\begin{aligned} A &= \sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} \psi(x_m, t) \left[\int_{x_m}^{x_{m+1}} H(t - \tau_2^I(x)) f_2(x, t) dx \right] dt + O(\varepsilon) \\ A &= \int \int H(t - \tau_2^I(x)) \psi(x, t) f_2(x, t) dx dt + O(\varepsilon), \quad (39) \end{aligned}$$

where we have committed another $O(\varepsilon)$ error by taking the test function ψ inside the x -integral. Here H denotes the usual Heaviside function.

Now, we turn to the term B in (38)(b). We replace the difference in the time direction by a partial derivative, giving

$$B = -\varepsilon^2 \sum_{n=0}^{\infty} \sum_{m=-\infty}^{\infty} \partial_t \Psi(x_m, \tau(m+1, n)) \Delta_m \tau(m, n) + O(\varepsilon).$$

Again, we commit only an error of order $O(\varepsilon)$ in doing so, even if, according to the assumptions, a bounded number of the $\Delta_m \tau(m, n)$ are of order $O(\frac{1}{\varepsilon})$, since the test function Ψ will be bounded. We split of the $n = 0$ term in the sum and write

$$B = -\varepsilon^2 \sum_{n=1}^{\infty} \sum_{m=-\infty}^{\infty} \partial_t \Psi(x_m, \tau(m+1, n)) \Delta_m \tau(m, n) - \varepsilon^2 \sum_{m=-\infty}^{\infty} \partial_t \Psi(x_m, \tau^I(m+1)) \Delta_m \tau^I(m) + O(\varepsilon).$$

Clearly, the second term is of order $O(\varepsilon)$ again and can be neglected. Using the definition (17)(b) of ρ , we obtain

$$B = -\varepsilon \sum_{n=1}^{\infty} \sum_{m=-\infty}^{\infty} \frac{h_m}{X} \partial_t \Psi(x_m, \tau(m+1, n)) \rho(x_m, \tau(m+1, n-1)) \Delta_n \tau(m+1, n-1) + O(\varepsilon)$$

Now, we repeat essentially the same procedure used for the term A . From (17) and the definition of ρ_1 we obtain

$$\begin{aligned} \int_{\tau(m+1, n-1)}^{\tau(m+1, n)} \rho_1(x_m, t) dt &= \varepsilon \Delta_n \tau(m+1, n-1) \rho(x_m, \tau(m+1, n-1)) \\ B &= - \sum_{n=1}^{\infty} \sum_{m=-\infty}^{\infty} \frac{h_m}{X} \partial_t \Psi(x_m, \tau(m+1, n)) \int_{\tau(m+1, n-1)}^{\tau(m+1, n)} \rho_1(x_m, t) dt + O(\varepsilon) = \\ &\quad - \sum_{m=-\infty}^{\infty} \frac{h_m}{X} \int_{\tau^I(m+1)}^{\infty} \partial_t \Psi(x_m, t) \rho_1(x_m, t) dt + O(\varepsilon) \end{aligned}$$

Using the definition of ρ_1 as the spatial difference of $-u_1$ gives

$$\sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} H(t - \tau^I(m+1)) \partial_t \Psi(x_m, t) [u_1(x_{m+1}, t) - u_1(x_m, t)] dt + O(\varepsilon)$$

Re- grouping the terms in the above expression yields

$$B = - \sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} H(t - \tau^I(m)) \partial_t [\Psi(x_m, t) - \Psi(x_{m-1}, t)] u_1(x_m, t) dt + O(\varepsilon) =$$

$$\begin{aligned}
& - \sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} H(t - \tau^I(m)) h_{m-1} \partial_t \psi(x_{m-1}, t) u_1(x_m, t) dt + O(\varepsilon) = \\
& - \sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} \partial_t \psi(x_{m-1}, t) \left[\int_{x_{m-1}}^{x_m} H(t - \tau_2^I(x)) u_2(x, t) dx \right] dt + O(\varepsilon) ,
\end{aligned}$$

giving altogether

$$B = - \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \partial_t \psi(x, t) H(t - \tau_2^I(x)) u_2(x, t) dx \right] dt + O(\varepsilon) . \quad (40)$$

Combining (39) and (40) gives

$$\int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} H(t - \tau_2^I(x)) [\psi(x, t) f_2(x, t) + \partial_t \psi(x, t) u_2(x, t)] dx \right] dt = O(\varepsilon) ,$$

which, in the limit $\varepsilon \rightarrow 0$, is the weak formulation of

$$\partial_t u_2 = f_2, \quad x \in \mathbb{R}, \quad t > \tau_2^I(x), \quad u_2(x, \tau^I(x)) = 0$$

Because of the definition of $u_1(x_0, t)$ in (23) this is the solution of (25) on the interval $[0, X]$. ■

REFERENCES

- [1] E.J. Anderson: A new continuous model for job shop scheduling, *International J Systems Science* 12, pp.: 1469-1475 (1981).
- [2] D. Armbruster, D. Marthaler, C. Ringhofer: Modeling a re-entrant factory, *submitted, Operations Research*, preprint available at URL: <http://math.la.asu.edu/~chris> (2002).
- [3] D. Armbruster, D. Marthaler, C. Ringhofer: A mesoscopic approach to the simulation of semiconductor supply chains, *Proceedings of the International Conference on Modeling and Analysis of Semiconductor Manufacturing (MASM2002)*, G. Mackulak et al, eds., pp. 365 - 369 (2002).
- [4] D. Armbruster, D. Marthaler, C. Ringhofer: Kinetic and Fluid Model Hierarchies for Supply Chains, *SIAM J. on Multiscale Modeling and Simulation*, Vol.2 Nr.1, pp. 43-61 (2004).
- [5] D. Armbruster, C. Ringhofer: Thermalized kinetic and fluid models for re-entrant supply chains, submitted, *SIAM J. Multiscale Modeling and Simulation*, preprint available at URL: <http://math.la.asu.edu/~chris> (2003).
- [6] A.Aw, M.Rasle: Resurrection of "second order" models of traffic flow, *SIAM J. Appl. Math.* 60(3), 916-938, (2000).

- [7] J. Banks, J. Carson II, B. Nelson: Discrete event system simulation, *Prentice Hall* (1999).
- [8] R. Billings, J. Hasenbein: Applications of fluid models to semiconductor fab operations, *preprint*, (2001).
- [9] P. Brandt, A. Franken, B. Lisek: Stationary stochastic models, *Wiley, New York*(1990).
- [10] C. Cercignani: The Boltzmann equation and its applications, *Applied Mathematical Sciences, vol. 67, Springer Verlag* (1988).
- [11] C. Daganzo: Requiem for second order fluid approximations of traffic flow, *Transportation Research B, 29, 277-286*, (1995).
- [12] C. Daganzo: A Theory of supply chains, *ITS Research reportUCB-ITS-RR2001-7, URL: <http://www.ce.berkeley.edu/~daganzo/>* (2001).
- [13] M. el-Taha, S. Stidham: Sample path analysis of queuing systems, *Kluwer's international series* (1999).
- [14] D. Helbing: Gas kinetic derivation of Navier Stokes like traffic equations, *Phys. Rev. E 53, 2366-2381* (1996).
- [15] D. Helbing: Traffic and related self-driven many particle systems, *Reviews of modern physics 73, 1067-1141* (2001).
- [16] M. Lighthill, J. Whitham: On kinematic waves, I: Flow movement in long rivers, II: A theory of traffic flow on long crowded roads, *Proc. Royal. Soc., A229, 281-345* (1955).
- [17] G.F. Newell: A simplified theory of kinematic waves in highway traffic, *Transportation Research 27B*, pp. 281-313 (1993).
- [18] I. Prigogine, R. Herman: Kinetic theory of vehicular traffic, *New York: Elsevier* (1971).
- [19] M. C. Pullan: An algorithm for a class of continuous linear programs, *SIAM J. Control and Optimization 31, pp.1558-1577* (1993).