PARAMETER EXTRACTION OF COMPLEX PRODUCTION SYSTEMS VIA A KINETIC APPROACH

ALI K. UNVER*
Intel Corporation, Assembly & Test Technology Development
5000 W. Chandler Blvd
Chandler, AZ 85226, USA

CHRISTIAN RINGHOFER
Arizona State University
Department of Mathematics and Statistics
Tempe, AZ 85287, USA

M. EMIR KOKSAL
Ondokuz Mayis University
Department of Mathematics
Samsun, 55139, Turkey

(Communicated by Irene M. Gamba)

ABSTRACT. Continuum models of re-entrant production systems are developed that treat the flow of products in analogy to traffic flow. Specifically, the dynamics of material flow through a re-entrant factory via a parabolic conservation law is modeled describing the product density and flux in the factory. The basic idea underlying the approach is to obtain transport coefficients for fluid dynamic models in a multi-scale setting simultaneously from Monte Carlo simulations and actual observations of the physical system, i.e. the factory. Since partial differential equation (PDE) conservation laws are successfully used for modeling the dynamical behavior of product flow in manufacturing systems, a re-entrant manufacturing system is modeled using a diffusive PDE. The specifics of the production process enter into the velocity and diffusion coefficients of the conservation law. The resulting nonlinear parabolic conservation law model allows fast and accurate simulations. With the traffic flow-like PDE model, the transient behavior of the discrete event simulation (DES) model according to the averaged influx, which is obtained out of discrete event experiments, is predicted. The work brings out an almost universally applicable tool to provide rough estimates of the behavior of complex production systems in non-equilibrium regimes.

1. Introduction. In recent years, factories and production systems have become larger and more complicated. For this reason, a research endeavor has been initiated to find time and cost efficient ways of production for such supply chains. Moreover, a wide range of traffic flow theories and models have been developed to find the most effective means of production. While discrete event simulators have been highly

2010 Mathematics Subject Classification. Primary: 65N35, 35K55; Secondary: 62P30.
Key words and phrases. Supply chains, re-entrant factory, numerical analysis, conservation law, nonlinear parabolic PDE, statistics.

This work is supported by NSF grant DMS-0204543.
* Corresponding author: Ali K. Unver.
successful in simulating single factories, they are computationally too expensive
to simulate even a moderately complicated production system [1]. Our goal in this
paper is to present a computationally efficient way to simulate stochastic production
systems.

A supply chain is a network of organizations comprising suppliers, manufactur-
ers, customers, in some cases even distributors and retailers to dispatch any type
of goods, services, and information to customers enabling the members gain com-
petitive advantage in the market place. To gain competitive advantage, a supply
chain should function efficiently and effectively to maximize the profits of the mem-
bers and minimize the delivery time of goods. The performance of the whole chain
depends on how efficient and effective each member functions and in the literature
there has been a host of research on efficient and effective functioning of supply
chains. Higher customer satisfaction and profits, significant determinants of the
performance of supply chains, might be obtained by decreasing conflicts within the
chain and creating superior customer value.

Understanding the behavior of large supply chains under different polices and
scenarios is a major issue for many businesses today. In large factories, no experi-
ments can be done involving whole supply chains. Therefore, simulation models are
developed, which substitute for the real environment. Especially in recent years,
fast scalable simulations of production flows in a supply chain have become a very
important research topic. The long term goal of a supply chain simulation is to
optimize and control the production across the whole supply chain. Since most
production deals with individual parts and the processes that these parts undergo,
discrete event simulators would be the regular method of choice for accurate simu-
lations.

While discrete event simulators have been highly successful in simulating single
factories, they are computationally too expensive to simulate even a moderately
complicated production system. Also, they are not scalable to a full supply chain
[1]. Alternative models that endow supply chain nodes with fixed production ca-
pacitated systems and fixed lead times are not accurate enough since they do not
take into account the fact that capacitated system responds nonlinearly to increases
in demand close to the limit of the production capacity.

The approach followed in this paper is to extract throughput times (TPTs) and
work in progress (WIP) levels from a discrete event simulation, replacing actual
observations of the physical system for the purposes of this work, and then to
transform them into transport coefficients of a macroscopic conservation law. Using
discrete event simulations we estimate the transport coefficients; namely velocity
(convection, $C$) and diffusion ($D$). Velocity coefficient stands for the local velocity
of each particle throughout the system. Diffusion coefficient exists due to the re-
entrance of the particles in the system, which will be similar to going backwards in
the stages. Since we deal with stochastic systems; i.e., sometimes some machines
might be down, this will cause both velocity and diffusion coefficients to be varying
throughout the system. Ultimately, after extracting $C$ and $D$, we are able to solve
the diffusive PDE numerically. The PDE model that we use in this project is the
conservation law given as follows:

$$\partial_t \rho(x, t) + \partial_x F(x, t) = 0, \quad F(x, t) = C \rho(x, t) - D \partial_x \rho(x, t)$$

(1)

where $x$ and $t$ are space and time variables respectively. In the computations, $x$ is
considered as “stage” of the processes, which is actually an artificial “space” (road)
in the simulations [13].
Here, $\rho(x,t)$, which is actually the local work in progress, denotes the density of the particles and $F(x,t)$ denotes the flux, both of which are used to model the dynamics of material flow through the re-entrant factory. Density of products ($\rho(x,t)$) is the conserved variable in the system with units [parts/space]. The flux $F(x,t)$ is the number of parts per time with units [parts/time]. The velocity and diffusion coefficients are extracted from observations of the system (replaced by a discrete event simulation model for the purposes of this paper). Since PDE models are amenable to optimization and control [10], we basically merge the randomness and optimization by using both DES and PDE models.

We base our theory on the assumption that we have the following type of data available. The production process consists of $M$ stages. $a_{mn}^n$ denotes the time at which lot number $n$ arrives at stage $m$. Thus, the basic input of our macroscopic model consists of a table of arrival times $a_{mn}^n$, $n = 1 : N$, $m = 1 : M$. From this table, the throughput times $\tau_{mn}^n$ for each stage as well as WIP in any part of the supply chain can be easily computed. For instance, given the arrival times $a_{mn}^n$ the WIP in the $m$-th supplier $W_m(t)$ is computed as

$$W_m(t) = \sum_n H(t - a_{mn}^n) - H(t - a_{mn}^{n+1})$$

where $H$ denotes the usual Heaviside function, which is also known as the integral of the Dirac delta function. As lots/particles go through the stages, we compute their velocities and variances depending on observed parameters. Since each particle has a different processing time at each stage, it has a varying velocity through out the system, so this causes a variance in velocity for each particle. According to fluid dynamics theory, variance results in diffusion. This results in computing TPTs, means, and variances depending on a chosen set of state parameters. We record the arrival times when lots pass through each stage in the production process. At the same time we record some macroscopic quantity (such as the WIP) of the system at each of the arrival times $a_{mn}^n$, which gives a statistical distribution of throughput times, parameterized by this macroscopic quantity. We extract the transport coefficients; namely, velocity (convection) and diffusion coefficients of the particles in the production system, using DES. Detailed information on all features of DES can be found in [5] and [8]. The velocity and diffusion coefficients in (1) are then related to the mean and variance of this throughput distribution. Here, our goal is to form a compact model which is amenable itself to optimization. PDE models have the important advantage of being fast and that optimization can be applied to obtain accurate results. Our ultimate goal is to predict and optimize the transient behavior of a given system from observation without knowing details about the internal mechanisms of the system; i.e. we always want to know how the system behaves even though the features of the system is changed; for example, if a processor is changed/canceled in the system or if a machine is down for a while...,etc. Our ultimate goal is to predict and optimize the transient behavior of a given system from observation without knowing details about the internal mechanisms of the system; i.e. we always want to know how the system behaves even though the features of the system is changed.

The contents of the rest of the paper are as follows: In Section 2, we present the theoretical background of traffic flow-like PDE models and how to relate TPTs and WIPs to velocity and diffusion coefficients. In Section 3, we explain the model hierarchies that we used to derive the flux function and how we compute the TPT.
distribution and its mean and variance from the given data. Finally, in Section 5, we apply the theories that we cover in this paper on a model problem and give numerical results comparing the discrete event experiments to the PDE model.

2. Traffic flow-like PDE models. Traffic flow-like PDE models are continuum limits of fluid models. These models have several advantages: They are scalable and, most important of all, they are amenable to control and optimization [10]. There are two different types of PDE modeling: First principle modeling and direct modeling from observations. In the first principle modeling; on average there exists a functional relation between the flux and the WIP, which is known as a clearing function. Whereas in direct modeling from observation, which we do in our PDE modeling, the clearing function is replaced by a probability distribution function that is obtained from the observed data (DES in our case). After we obtain the data (randomly generated times) from DES, we get the TPT's and then, we extract the transport coefficients by using mean and variance of TPT’s. We can then solve the PDE-conservation law given in Equation (1).

In our traffic flow-like PDE model, we introduce an artificial variable that represents each product’s percentage of completion, or stage. The stage \( x \) goes from 0 (0%) to 1 (100%). \( x = 0 \) denotes the start of a product into the factory, in other words, the product enters the factory, and \( x = 1 \) denotes the end of a product that means the product leaves the factory. The arrival rates of the products at stages yield a flux function. Both the density and flux functions are very important in PDE modeling such that we use them to model the dynamics of material flow through the whole factory.

The total number of individual products in the system at time \( t \) that are at most \( 100x\% \) of the way through the sequence of stages (machines) that those products must visit before leaving the system can be found at each time \( t \geq 0 \) as follows

\[
W(t) = \int_0^1 \rho(x,t) \, dx.
\]  

(2)

The PDE model that we use in this project is a parabolic conservation law. A conservation law is a relation asserting that a specific quantity is conserved. Basically, for a quantity, to be conserved means that whatever enters to the system has to come out of that system after some time. For instance, in a manufacturing system, all the products that enter the system will exit the system. So, the quantity of the products will be conserved.

In our setting, a conservation law can be defined as a partial differential equation that expresses the fact that some physical quantity is locally conserved in a fluid or other continuous physical system, such as energy, momentum, or the quantity of fluid itself. Conservation law in the integral form is given by

\[
\forall a, b \in [0, 1] \quad \partial_t \int_a^b \rho(x,t) \, dx = \text{influx} - \text{outflux}
\]  

(3)

The integral in Equation (3) gives the number of products per time between \( x = a \) and \( x = b \).

3. Hierarchy of models. We present a model hierarchy to derive the flux function \( F \) given in Equation (1) for a given production system. The model hierarchy consists of three parts, namely; a particle model for the trajectories of individual parts, a kinetic model for the phase space density, and a diffusion convection equation.
 Particle and kinetic models were explained in detail in [3], however brief description of these models are also given in this paper below.

3.1. Particle models. In particle models, the evolution of a large ensemble of lots (particles) is modeled by describing the trajectories of each individual lot in phase space by a set of Newton equations. For a continuous time model, we introduce the trajectories as \( x = \xi(t) \) and \( v = \frac{\partial x}{\partial t} \). The velocity \( v \) is updated periodically from a velocity distribution \( V \), which we get through the TPT’s obtained from DES, so that we obtain a random variable \( v \) from

\[
dP \left\{ \frac{1}{v_n} = v \right\} = V_m(v, a_n^m)dv.
\]

where \( P \left\{ \frac{1}{v_n} = v \right\} \) denotes the probability of a local velocity \( \frac{1}{v_n} \) to be equal to \( v \) and \( V_m \) denotes the velocity distribution at \( m^{th} \) stage. Here, \( n \) and \( m \) stand for the numbers of lots and stages respectively; and \( a_n^m \) represents the time for each of these lots at each stage.

Now, the position \( \xi(t) \) after a small time interval \( dt \) is given by:

\[
\xi(t + dt) = \xi(t) + dt \cdot v \tag{5}
\]

where \( v \) is the random velocity variable.

Similarly, we have a rule for updating the velocity \( v \) according to a random variable \( \kappa \), which we calculate by flipping a coin. The probability of \( \kappa = 1 \) is \( w dt \) and the probability of \( \kappa = 0 \) is \( 1 - w dt \). Here \( w \) is the update frequency. In Section 3.2, we explain how to choose the update frequency \( w \). Now, according to this rule, if \( \kappa = 0 \), we maintain the current velocity, and if \( \kappa = 1 \), we update the velocity \( v \) from the probability distribution (4) given by the formula

\[
v(t + dt) = (1 - \kappa)v(t) + \kappa v'. \tag{6}\]

Here, \( v' \) is also a random velocity variable that we get from the velocity distribution \( V \), which can be seen in Equation (4). Now, we describe in detail how to compute the velocity distribution \( V \) that is obtained from discrete event experiments.

As we run a simulation, or observe an actual system, where \( N \) parts pass through \( M \) stages \( s_1, ..., s_M \), we record the times \( a_n^m \) when part number \( n \) arrives at stage \( s_m \) and finally at the exit, which corresponds to stage \( s_{M+1} \). At the same time, we record some macroscopic state variables, denoted by \( Z_n^m \), at every time \( a_n^m \).

\[
Z_n^m = (Z_1(n,m),...,Z_K(n,m)) \text{ is in general a vector of } K \text{ components. These } Z \text{ components can be variables like the total WIP, downstream WIP (at higher stages), the upstream WIP (at lower stages), or just the station index } m \text{ itself. We eventually form two tables:}
\]

\[
\begin{pmatrix}
  a_1^1 & a_2^2 & \cdots \\
  a_2^1 & a_2^2 & \cdots \\
  \vdots & \ddots & \cdots \\
  \vdots & & \ddots & \cdots \\
  \vdots & & & \ddots & \cdots \\
  \vdots & & & & \ddots & \cdots \\
  \vdots & & & & & \ddots & \cdots \\
  \vdots & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & & & & \ddots \end{pmatrix}
\begin{pmatrix}
  Z_1^1 & Z_2^1 & \cdots \\
  Z_2^1 & Z_2^2 & \cdots \\
  \vdots & \ddots & \cdots \\
  \vdots & & \ddots & \cdots \\
  \vdots & & & \ddots & \cdots \\
  \vdots & & & & \ddots & \cdots \\
  \vdots & & & & & \ddots & \cdots \\
  \vdots & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & & \ddots & \cdots \\
  \vdots & & & & & & & & & & & & & & & & & & \ddots \end{pmatrix}
\]

(7)

where there are different possibilities that the macroscopic state variable \( Z \) might depend on. \( Z_n^m = m \) if we want to make the velocity dependent on space only, or \( Z_n^m = W(a_n^m) \) for dependence on the total WIP only, or \( Z_n^m = (W_{<m}(a_n^m),W_{>m}(a_n^m)) \) for dependence on the WIP in front and behind the part. From one of these possibilities, we compute a distribution for the velocities, dependent on the global variables \( Z \). We map the work stations \( s_1, ..., s_M \) on a stage interval \([0, 1]\) where the stage \( s_m \) corresponds to the interval \((x_m-1, x_m)\) with \( x_m = \frac{m}{M} \), \( m = 0, ..., M \). Then,
the throughput time of lot $n$ entering $s_m$ is given by $\tau^m_n = a^m_n + 1 - a^m_n$, where $m = 1, \ldots, M$ and $n = 1, \ldots, N$. We divide the $K$ dimensional space of the macroscopic variables $Z^m_n$ into cells $C(k)$ where $k = (k_1, \ldots, k_K)$ and then we define the indicator function $\chi_k(Z)$ by

$$\chi_k(Z) = \begin{cases} 1 & \text{for } Z \in C(k) \\ 0 & \text{else.} \end{cases}$$

For the velocity $v$, where $v = \frac{1}{M(\tau^m_n)}$, the collected data give a discrete probability distribution dependent on the macroscopic state variable $Z$ of the form

$$\nu(v, x, Z) = \frac{\sum_n \delta(v - \frac{1}{M(\tau^m_n)})\chi_k(Z^m_n)}{\sum_k \chi_k(Z^m_n)}$$

for $Z \in C(k)$ and $x \in [x_{m-1}, x_m)$, with a mean $\langle v \rangle(x, Z)$ and a variance $\sigma^2(x, Z)$ given by $\langle v \rangle(x, Z) = \frac{\sum_n \frac{1}{M(\tau^m_n)}\chi_k(Z^m_n)}{\sum_k \chi_k(Z^m_n)}$ and $\sigma^2(x, Z) = \frac{\sum_n \frac{1}{M(\tau^m_n)}\chi_k(Z^m_n) - \langle v \rangle(x, Z)^2}{\sum_k \chi_k(Z^m_n)}$ for $Z \in C_k$, $x \in [x_{m-1}, x_m)$.

For the final model used in this work, the transport coefficients needed will be given solely in terms of these means and variances \[12\].

We are dealing with highly re-entrant production systems with many machines and buffers. Throughout the factory, machines with different speeds process the products. This accounts for the fact that in a re-entrant system, later arrivals influence the processing time of a particular product.

Equations (5) and (6) for the position and velocity define a stochastic process for the evolution of the products. So, briefly, the update algorithm for position ($\xi$) and velocity ($v$) is given by

\begin{equation}
(\text{a}) \quad \xi(t + \Delta t) = \xi(t) + \Delta t v(t) , \quad v(t + \Delta t) = \kappa(t) \nu(v) + (1 - \kappa(t))v(t) , \quad (8) \\
(\text{b}) \quad P\{\kappa(t) = 1\} = \omega \Delta t , \quad P\{\kappa(t) = 0\} = 1 - \omega \Delta t \\
(\text{c}) \quad \xi(0) = 0
\end{equation}

As it can be seen in Equation (8), for every $\Delta t$, we choose a random number $\kappa$, which is either zero or one, to decide whether to update the velocity $v$ or not. We update the velocity at every time step of length $\Delta t$ with probability $\omega \Delta t$. In general we let $\omega$ depend on $v(t)$ and $\xi(t)$. So, (8)(b) will be replaced by

$$P\{\kappa(t) = 1\} = \Delta t \omega(v, \xi) , \quad P\{\kappa(t) = 0\} = 1 - \Delta t \omega(v, \xi) , \quad (8)(b')$$

Choosing update frequency $\omega$ is discusses in Section 3.2. For analytic purposes, we would prefer to have a continuous time model, where the function $\xi(t)$ is not given only at discrete time intervals. This could be achieved by the limit $\omega \to \infty$, giving a stochastic process. However, in this limit we would lose the information about the update frequency $\omega$, which is a key parameter for controlling the stochasticity of the model.

3.2. Kinetic models. This section is leading to a kinetic equation for the evolution of the density of the products at stage $x$ with velocity $v$.

The update algorithm (8) is similar to a Monte Carlo method for the solution of a Boltzmann equation (see \[7\] for details). The Boltzmann equation is the kinetic equation for the time ($t$) evolution of the kinetic density function $f(x, v, t)$ in one-particle phase space, where $x$ and $v$ are position and velocity, respectively.

The function $\xi(t)$, defined in Section 3.1, plays the role of a position which is advanced with a randomly changing velocity $v(t)$. It is therefore tempting to derive
a kinetic formulation. These models are often referred to as traffic flow models [2], [9], because of the analogy to lots moving on a freeway from start to completion. A product arrives at the first stage at time \( t = a \) and moves from \( x = 0 \) to \( x = 1 \) along a trajectory with velocity \( v_a(t) \) in such a way that it reaches the end \( x = 1 \) at time \( t = e \), then the velocity has to be chosen in such a way that \( \int_0^e v_a(t) \, dt = 1 \) holds.

The particle model for a product that arrives at the first stage of the system at \( t = a \) together with the phase speed model (8) is given as follows: The trajectory \( x(t) \) at time \( t \) along a trajectory with velocity \( v \) satisfies the same Newton equations as in (8) together with the initial conditions

\[
\xi_a(a) = 0, \quad dP\{v_a(a) = v\} = \mathcal{V}(v,0)dv.
\]  

(9)

The total WIP \( W(t) \), given in terms of the WIP density \( \rho(x,t) \), and the flux \( F(x,t) \) at any point \( x \in [0,1] \) are then given by

\[
\rho(x,t) = \int \delta(x - \xi_a(t)) \lambda_a(a) \, da, \quad W(t) = \int_0^1 \rho(x,t) \, dx,
\]

\[
F(x,t) = \int (\delta(x - \xi_a(t)) \lambda_a(a)v_a(t) \, da.
\]

(10)

Here \( \lambda(t) \), the influx density, is concentrated on the arrival times \( a_n^0 \) of parts in the system, i.e.

\[
\lambda(t) = \sum_n \delta(t - a_n^0)
\]

holds. From the definition given in (10), it is obvious that the WIP density \( \rho \) and the flux \( F \) will satisfy the conservation law

\[
\partial_t \rho + \partial_x F = 0.
\]

(11)

We make a mean field assumption for the rest of this section, that is we will assume that the velocity distribution \( \mathcal{V}(v,x) \) is a given function of velocity and stage. In reality, the velocity distribution will depend on the lot trajectories \( \xi \) through the WIP \( W \), given by (10). The idea underlying this assumption is that, for many lots present in the system, the impact of one individual lot on the distribution \( \mathcal{V} \) is negligible. Therefore, the lots can be treated as approximately statistically independent. This is a standard approach for a system of many molecules in gas dynamics [7]. Making this mean field assumption, we have the following theorem:

**Theorem 3.1.** Let \( f_a(x,v,t) \) denote the probability density for finding \( \xi_a, v_a \), defined by (9), at position \( x \) and velocity \( v \), i.e. \( dP\{\xi_a(t) = x, v_a(t) = v\} = f_a(x,v,t) \, dx \, dv \). Then, \( f_a \) satisfies the initial value problem

\[
(a) \quad \partial_t f_a + v \partial_x f_a = \mathcal{V}(v,x) \int \omega(v',t)f_a(x,v',t) \, dv' - \omega(v,t)f_a(x,v,t), \quad t > a
\]

\[
(b) \quad f_a(x,v,a) = \delta(x)\mathcal{V}(v,x)
\]

in the limit \( \Delta t \to 0 \), weakly in \( x, v \) and \( t \).

Using Theorem 3.1, we can derive the corresponding initial boundary value problem for the kinetic density \( f(x,v,t) \) of the number of lots at position \( x \) with velocity \( v \).

**Proof.** The proof consists of essentially summing up all possibilities for the values of \( \kappa(t), \xi_a(t) \) and \( v_a(t) \) and weighting them with their respective probabilities. This takes the form

\[
f_a(x,v,t + \Delta t) =
\]
\[
\int \delta(x' + \Delta v' - x)\delta(ku + (1 - k)v' - v)dP\{\kappa(t) = k\}dP\{v(t) = u\}
\]

\[
dP\{\xi_a(t) = x', v_a(t) = v'\}
\]

\[
= \int \delta(x' + \Delta v' - x)\delta(ku + (1 - k)v' - v)[\Delta \omega(v', t)\delta(k - 1) + (1 - \Delta \omega(v', t))\delta(k)]
\]

\[
\mathcal{V}(u, x) f_a(x', v', t) \, dv' du' dk .
\]

Integrating the \(\delta\)– functions with respect to \(k\) gives

\[
f_a(x, v, t + \Delta t) = \int \Delta \omega(v', t)\delta(x' + v'\Delta t - x)\delta(u - v)\mathcal{V}(u, x) f_a(x', v', t) \, dv' du'
\]

\[
+ \int (1 - \Delta \omega(v', t))\delta(x' + v'\Delta t - x)\delta(v' - v)\mathcal{V}(u, x) f_a(x', v', t) \, dv' du' .
\]

Integrating with respect to \(x', v'\) and \(u\) in the above two integrals gives

\[
f_a(x, v, t + \Delta t) = \int \Delta \omega(v', t)\mathcal{V}(v, x) f_a(x - v\Delta t, v', t) \, dr' + (1 - \Delta \omega(v', t)) f_a(x - v\Delta t, v, t) ,
\]

where we have used the fact that \(\int \mathcal{V}(u, x) \, du = 1\) holds. We now multiply (13) by a test function \(\psi(x, v, t)\) and integrate over \(t > a\) giving

\[
\int H(t-a)\psi(x, v, t)f_a(x, v, t + \Delta t) \, dx dv dt
\]

\[
= \int H(t-a)\psi(x, v, t)\Delta \omega(v', t)\mathcal{V}(v, x) f_a(x - v'\Delta t, v', t) \, dv' dx dv dt
\]

\[
+ \int H(t-a)\psi(x, v, t)(1 - \Delta \omega(v, t)) f_a(x - v\Delta t, v, t) \, dx dv dt ,
\]

where \(H\) denotes the Heaviside function. Shifting the arguments in the integrals gives

\[
\int H(t - \Delta t - a)\psi(x, v, t - \Delta t)f_a(x, v, t) \, dx dv dt
\]

\[
= \int H(t-a)\psi(x + v\Delta t, v, t)\Delta \omega(v', t)\mathcal{V}(v, x) f_a(x, v', t) \, dr' dx dv dt
\]

\[
+ \int H(t-a)\psi(x + v\Delta t, v, t)(1 - \Delta \omega(v, t)) f_a(x, v, t) \, dx dv dt .
\]

The left hand side of the above equation can be rewritten as

\[
\int H(t-a)\psi(x, v, t - \Delta t)f_a(x, v, t) \, dx dv dt - \int_a^{a+\Delta t} \left[ \int \psi(x, v, t - \Delta t)f_a(x, v, t) \, dx dv \right] dt
\]

\[
= \int H(t-a)[\psi - \Delta t \partial_t \psi]f_a(x, v, t) \, dx dv dt - \Delta t \int \psi(x, v, t) f_a(x, v, t) \, dx dv + O[(\Delta t)^2]
\]

\[
= \int H(t-a)[\psi - \Delta t \partial_t \psi]f_a(x, v, t) \, dx dv dt - \Delta t \int \psi(0, v, t) \mathcal{V}(v, a) \, dv + O[(\Delta t)^2] .
\]

(15)

Expanding the right hand side of (14) gives

\[
\int H(t-a)\psi(x, v, t)\Delta \omega(v', t)\mathcal{V}(v, x) f_a(x, v', t) \, dv' dx dv dt
\]

\[
+ \int H(t-a)(\psi + v\Delta \omega \psi) f_a(x, v, t) \, dx dv dt
\]
\[ - \int H(t-a) \psi(x,v,t) \Delta t \omega(v,t) f_a(x,v,t) \, dx dv dt + O[(\Delta t)^2]. \]

Setting this equal to (15) and dividing by \( \Delta t \) gives

\[ - \int H(t-a) \partial_t \psi + v \partial_x \psi \, f_a(x,v,t) \, dx dv dt - \int \psi(0,v,a) V(v,a) \, dv \]

\[ = \int H(t-a) \psi(x,v,t) \omega(v',t) V(v,x) f_a(x,v',t) \, dv' \, dx dv dt \]

\[ - \int H(t-a) \psi(x,v,t) \omega(v,t) f_a(x,v,v) \, dx dv dt + O(\Delta t), \]

which for \( \Delta t \to 0 \) is the weak formulation of (12).

Now, we derive the corresponding initial boundary value problem for the kinetic density \( f(x,v,t) \). We define

\[ f(x,v,t) = \int H(t-a) f_a(x,v,t) \lambda(a) \, da. \] (16)

\( f \) is not a probability density since the trajectory \( \xi_a \) will only exist for \( t > a \) and the number of lots will not be constant over time. The lot density \( f \) will satisfy the same Boltzmann equation as the probability density \( f_a \) with a more natural influx boundary condition replacing the initial condition (12)(b).

Now, we have the following theorem:

**Theorem 3.2.** The lot density \( f \), given in (16), satisfies the initial boundary value problem (IBVP)

\[ (a) \ \partial_t f + v \partial_x f = V(v,x) \int \omega(v',t) f(x,v',t) \, dv' - \omega(v,t) f(x,v,t), \quad x > 0, \ t > 0 \] (17)

\[ (b) \ f(0,v,t) = \frac{\lambda(t)}{v} V(v,x), \quad f(x,v,0) = 0 \]

weakly in \( x, v \) and \( t \). The Boltzmann equation (17) is a generalization of the kinetic model derived in [2]. The random phase approach has introduced the concept of collisions, i.e. the integral operator on the right hand side of (17), into the particle model.

**Proof.** The weak form of the initial value problem (12) implies that for any test function \( \psi(x,v,t) \) the relation

\[ - \int H(t-a) f_a[\partial_t \psi + v \partial_x \psi](x,v,t) \, dx dv dt = \int \psi(0,v,a) V(v,x) \, dv + \] (18)

\[ \int \{H(t-a) \psi(x,v,t)[V(v,x) \int \omega(v',t) f_a(x,v',t) \, dv' - \omega(v,t) f_a(x,v,t)] \} \, dx dv dt \]

holds, where all integrals range over \( \mathbb{R} \) and \( H \) denotes the usual Heaviside function. Multiplying (18) by \( \lambda(a) \) and integrating with respect to \( a \) gives

\[ - \int f[\partial_t \psi + v \partial_x \psi](x,v,t) \, dx dv dt = \int \psi(0,v,a) V(v,x) \lambda(a) \, dv + \] (19)

\[ \int \{\psi(x,v,t)[V(v,x) \int \omega(v',t) f(x,v',t) \, dv' - \omega(v,t) f(x,v,t)] \} \, dx dv dt . \]
Now, \( f_a(x,v,t) \) and \( \lambda(a) \) vanish identically for \( x < 0 \) and \( a < 0 \), respectively. Therefore, \( f(x,v,t) \) vanishes identically for \( x < 0, t < 0 \). Thus, (19) is the same as

\[
- \int H(x)H(t)f[\partial_t \psi + v \partial_x \psi](x,v,t) \, dx dv dt = \int H(t)\psi(0,v,t)\mathcal{T}(v,t)\lambda(t) \, dv dt +
\]

\[
\int \{ H(x)H(t)\psi(x,v,t)[\mathcal{V}(v,x) \int \omega(v',t)f(x,v',t) \, dv' - \omega(v,t)f(x,v,t)] \} \, dx dv dt .
\]

(20) is the weak formulation of the boundary value problem (17).

### 3.2.1. Choosing the update frequency.

Now, it remains to choose the update frequency \( \omega(v,t) \). We remark that there necessarily is some arbitrariness in this choice. The discrete event simulation model yields one particular realization of a stochastic process. To compute meaningful answers, one therefore will run a series of discrete event experiments and compute averages. The more experiments, the closer the result will be to the average, where a degenerate throughput time distribution (a \( \delta \) function concentrated at the mean) is used. So, the size of \( \omega \) has to be chosen in the same way as the number of realizations of the discrete event simulation model is chosen. The guiding principle here has to be that the overall number of random numbers generated is the same.

The basic idea of the approach presented so far is to modify the discrete event simulation model in order to take into account the transient changes in throughput times and influx, which will influence the outflux in the case of re-entrant nodes. So, on the other hand, if there are no changes in influx and throughput time, the discrete event model should correspond to a solution of the Boltzmann equation (17). Translating the discrete event simulation into kinetic model along the lines of the approach in [2] gives the following picture. For an arrival time \( a \), we pick a velocity \( v \) according to the distribution \( \mathcal{V}(v,x) \) and move with the constant velocity \( v = \frac{1}{2} \) until reaching \( x = 1 \) at time \( t = a + \tau \). This gives the relations for the lot trajectory in phase space by

\[
\xi_a(t) = v_a(t-a), \quad d\mathcal{P}\{v_a = v\} = \mathcal{V}(v,0)dv .
\]

We denote by \( f_a^d(x,v,t) \) the corresponding probability density in phase space, i.e. \( f_a^d(x,v,t)\, dx dv = d\mathcal{P}\{\xi_a(t) = x, v_a(t) = v\} \). So, \( f_a^d \) can be computed exactly as

\[
f_a^d(x,v,t) = \delta(v(t-a) - x)\mathcal{V}(v,x) .
\]

The probability density \( f_d(x,v,t) \) corresponding to (16) is then given by

\[
f_d(x,v,t) = \int H(t-a)\delta(v(t-a) - x)\mathcal{V}(v,x)\lambda(a) \, da = \frac{1}{v}H(x)\mathcal{V}(v,x)\lambda(t - \frac{x}{v}) .
\]

For the particle model to be equivalent to the discrete event simulation model, \( f_d \) has to be a solution of the IBVP (17). Since \( f_d \) is a function of \( t - \frac{x}{v} \), it satisfies \( \partial_t f_d + v \partial_x f_d = 0 \). \( f_d \) also satisfies the boundary condition (17)(b). Therefore, the residual obtained from inserting \( f_d \) into (17) is given by the collision term

\[
\mathcal{Q}[f^d](x,v,t) := \mathcal{V}(v,x) \int \frac{\omega(v',t)}{v'} \mathcal{V}(v',x')\lambda(t - \frac{x'}{v'}) \, dv' - \frac{\omega(v,t)}{v} \mathcal{V}(v,x)\lambda(t - \frac{x}{v}) ,
\]

and \( \mathcal{Q}[f^d] = 0 \) has to hold in the case that neither \( \mathcal{V} \) nor \( \lambda \) depend on time, giving

\[
0 = \mathcal{V}(v) \int \frac{\omega(v',t)}{v'} \mathcal{V}(v') \, dv' - \frac{\omega(v,t)}{v} \mathcal{V}(v) .
\]
which implies $\omega(v, t) = v \gamma(t)$ for some arbitrary function $\gamma(t)$. This makes sense since we update the phase speed more frequently for shorter current throughput times, i.e. for a faster transient response of the system. So, with this choice, the discrete event simulation and the random phase model (8) are equivalent for constant $V$ and $\lambda$ and any function $\gamma(t)$. In the general case, we choose $\gamma(t)$ such that we roughly compute the same number of random numbers for either model. Using the discrete event simulation, we pick $\Delta t \lambda(t)$ random numbers for the velocity in the time interval $\Delta t$. Using the random phase model, we pick $\Delta t \int v V(v, x) dv$ in the same time interval.

3.3. Diffusion convection equation. In this section, we explain how we obtain the velocity (convection) and diffusion coefficients. We express the flux $F$ in terms of the density $\rho$ via a functional expansion, namely the Chapman - Enskog expansion [6]. The Chapman - Enskog expansion is an essential asymptotic tool of the Boltzmann equation (17) for a regime where average throughput time and the time intervals between velocity updates in the particle model are small compared to the overall time scale. Fluid equations for the simulation of throughput times have been studied in [2] and [4], leading to simple convection equations for the density of lots as a function of degree of completion (DOC). We essentially recover these results but obtain as a generalization, an additional diffusive term arising from the variance of the velocity distribution $V$.

A fluid dynamic approximation to the Boltzmann equation (17) is obtained by taking integrals with respect to the velocity $v$. Defining the density $\rho(x, t)$ and the flux $F(x, t)$ by

$$
\rho(x, t) = \int f(x, v, t) \, dv, \quad F(x, t) = \int v f(x, v, t) \, dv, \quad (22)
$$

we obtain the conservation law

$$(a) \partial_t \rho + \partial_x F = 0, \quad (b) \rho(x, 0) = 0, \quad (c) F(0, t) = \lambda(t). \quad (23)$$

Here, $\rho$ and $F$, defined by (22), represent expectation values of the position and flux densities, respectively. The goal is now to find an expression for the flux $F$ in terms of the density $\rho$ using an asymptotic solution of the Boltzmann equation (17) large time scales.

We start by scaling the involved probability distributions and by bringing the Boltzmann equation (17) into an appropriate dimensionless form. From (17), we have that the IVBP for the unscaled Boltzmann equation for the part density is of the form

$$
\partial_t f + v \partial_x f = V(v, x) \int \omega(v', x) f(x, v', t) \, dv' - \omega(v, x) f(x, v, t), \quad x > 0, \ t > 0
$$

$$
f(0, v, t) = \lambda(t) V(v, 0), \quad f(x, v, 0) = 0, \ \theta \omega(v, x) = \frac{v \lambda_0}{V_1(x)}, \ \theta V_1(x) = \int_v v V(v, x) \, dv.
$$

The appropriate scale for the velocity (and therefore also the time scale) are given by the probability distribution $V(v, x)$. We choose the velocity scale $v_0 = \int_0^1 V_1(x) \, dx$, where $V_0$ is a characteristic value of the influx density $\lambda(t)$.
i.e. as the spatial average of the mean of the velocities, and rescale the probability distribution \( \mathcal{V} \) correspondingly:

\[
\mathcal{V}(v, x) = \frac{1}{v_0} \mathcal{V}_s(v_s, x), \quad v = v_0 v_s
\]

(We denote the scaled and dimensionless variables with subscript \( s \) here.) We choose the over all time scale to be \( \frac{1}{v_0} \) and define the scaled time as \( t = \frac{t_0}{v_0} \). We scale the influx density \( \lambda(t) \) by its characteristic value \( \lambda_0 \) and scale the scattering frequency \( \omega \) accordingly, setting

\[
\lambda(t) = \lambda_0 \lambda_s(t_s), \quad \omega(v, x) = \lambda_0 \omega_s(v_s, x) \Rightarrow \omega_s(v_s, x) = \frac{v_0 v_s}{V_1(x)}
\]

making \( \lambda_s \) and \( \omega_s \) dimensionless \( O(1) \) quantities. Finally, we scale the kinetic density \( f(x, v, t) \) itself by \( f(x, v, t) = \frac{\lambda_0}{v_0} f_s(x, v_s, t_s) \). The scaled initial boundary value problem for the Boltzmann equation (17) now reads:

\[
(a) \quad \partial_{t_s} f_s(x, v_s, t_s) + v_s \partial_v f_s = -\frac{1}{\varepsilon} \mathcal{V}_s(v_s, x) \int \omega_s(v'_s, x) f_s(x, v'_s, t_s) dv'_s - \omega_s(v_s, x) f_s(x, v_s, t_s)
\]

\[
(b) \quad f_s(x, 0, t_s) = \frac{\lambda_0(t_s)}{v_0} \mathcal{V}_s(v_s, 0), \quad f_s(x, v_s, 0) = 0,
\]

\[
(c) \quad \omega_s(v_s, x) = \frac{v_s}{V_{1s}(x)}, \quad V_{1s}(x) = \int v_s \mathcal{V}_s(v_s, x) dv_s, \quad \varepsilon = \frac{v_0}{\lambda_0}.
\]

The diffusion convection equation will be derived in the limit \( \varepsilon \to 0 \). Note that, according to Section 3.1 the average expectation \( v_0 \) is of order \( O(\frac{1}{M\tau}) \). Thus, the timescale \( t_0 = \frac{1}{v_0} = O(M\tau) \) corresponds to the total average throughput time through the whole chain, and \( \varepsilon = \frac{v_0}{\lambda_0} << 1 \) implies that the intervals between arrivals in the chain are small compared to the whole TPT, i.e. there are many parts in the whole system. Now, using the Chapman - Enskog expansion, the flux \( F \) is expressed for \( \varepsilon << 1 \) in terms of the density \( \rho \) [6]. For notational simplicity, we drop the subscript \( s \) from here on.

In the limit \( \varepsilon \to 0 \), equation (24) will be dominated by the right hand side. We therefore define the collision operator \( Q[f] \) as

\[
Q[f](x, v, t) = \mathcal{V}(v, x) \int \omega(v', x) f(x, v', t) dv' - \omega(v, x) f(x, v, t), \quad (25)
\]

and write the Boltzmann equation (24) as

\[
\partial_t f + v \partial_v f = \frac{1}{\varepsilon} Q[f]. \quad (26)
\]

The relation between the density \( \rho \) and the flux \( F \) in (22) is now obtained via asymptotics for \( \varepsilon \to 0 \), in other words by a Chapman - Enskog procedure. In general, the Chapman - Enskog expansion consists of splitting the dynamics of the Boltzmann equation (26) into a slow evolution on the kernel of the operator \( Q \) and a fast evolution on its orthogonal complement. While the Chapman - Enskog expansion is in general a nonlinear procedure [6], its complexity reduces significantly in the case of linear collision operators, and it can be written in terms of projection operators. To this end, we first define the kernel of the operator \( Q \) in (25).

The definition of \( Q \) is the following: When we apply \( Q \) on a function, we take the integral of that function and multiply by probability distribution \( \mathcal{V}(v) \), and then subtract that function from the multiplication obtained. The operator \( Q \) given in
the form \((24)(c)\) of the scattering frequency \(\omega\) element as

\[ W(v, x) = \frac{\mathcal{V}(v, x)}{vV_{-1}}, \quad V_{-1}(x) = \int \frac{1}{v} \mathcal{V}(v, x) \, dv. \]  

(27)

A projection on the kernel is then given by the operator

\[ P f(x, v, t) = W(v, x) \rho(x, t), \quad \rho(x, t) = \int f(x, v', t) \, dv'. \]  

(28)

\(P\) is a projection operator since the kernel density \(W(v, x)\) is normalized, i.e. \(\int W(v, x) \, dv = 1 \forall x \in W\), and therefore \(P^2 = P\) holds.

For the following, it will be notationally convenient to define symbols for the moments of the distributions \(\mathcal{V}\) and \(W\). We define

\[ V_j(x) = \int v^j \mathcal{V}(v, x) \, dv, \quad W_j(x) = \int v^j W(v, x) \, dv, \quad j \in \mathbb{Z}. \]  

(29)

Because of (27), the moments \(V_j\) and \(W_j\) are obviously related. They satisfy the relations

\[ W_j = \frac{V_{j-1}}{V_{-1}}, \quad V_j = \frac{W_{j+1}}{W_1}, \quad j \in \mathbb{Z}. \]  

(30)

We have the following theorem.

**Theorem 3.3.** For the dimensionless parameter \(\varepsilon \ll 1\), the flux \(F\) in (23) is given in terms of the density \(\rho\) as

\[ F(x, t) = C(x) \rho(x, t) - D(x) \partial_x \rho(x, t) + O(\varepsilon^2) \]  

(31)

with the velocity (convection) coefficient \(C\) and the diffusion coefficient \(D\) given by

\[ C = W_1 - \varepsilon W_2 W_{-1} \partial_x W_1, \quad D = \varepsilon W_2 (W_{-1} W_1 - 1), \]  

(32)

where the moments \(W_j\) of the kernel density \(W\) are defined as in (29). The diffusion coefficient \(D(x)\) satisfies \(D \geq 0\).

**Proof.** We split the function space \(\mathcal{B}\) for the kinetic density function \(f\) into the linear hull of the kernel and its complement:

\[ \mathcal{B} = \mathcal{B}_0 \oplus \mathcal{B}_1, \quad \mathcal{B}_0 = \mathcal{L}[\mathcal{W}], \quad \psi \in \mathcal{B}_1 \iff \int \psi \, dv = 0, \]

and define the projection onto the kernel space \(\mathcal{B}_0\) as in (28). The definition (28) of the projection operator \(P\) implies immediately that \(PQ = QP = 0\) holds. In other words, the range of \(Q\) is contained in the null space of \(P\).

Now, we split the evolution equation (26) into the evolution in the kernel manifold and the evolution in its complement \(\mathcal{B}_1\) by writing \(f = \phi + \psi, \phi = P[f], \psi = f - P[f]\), and by applying the projections \(P\) and \(I - P\) to equation (26), giving

\[ (a) \quad \partial_t \phi + P[v \partial_x \phi + v \partial_x \psi] = 0, \]

\[ (b) \quad (I - P) \partial_t (\phi + \psi) + (I - P)[v \partial_x \phi + v \partial_x \psi] = \frac{1}{\varepsilon} Q[\psi]. \]  

(33)
Equation (33)(a) gives the conservation law for the density $\rho$, whereas equation (33)(b) has to be solved asymptotically for $\psi$, giving the closure for the conservation law (33)(a). In detail, we have

\begin{align}
(a) \quad & \partial_t \rho + \partial_x F = 0, \quad (b) \quad F = W_1 \rho + \int v \psi \, dv, \\
(c) \quad & W_1(x) = \int v W(v, x) \, dv = \frac{1}{\epsilon_{-1}(x)}, \\
(d) \quad & \partial_t \psi + (I - P) \partial_x [v W \rho + v \psi] = \frac{1}{\epsilon} Q[\psi].
\end{align}

In order to solve equation (34)(d) asymptotically for $\epsilon\rightarrow 0$, we define the pseudo-inverse $Q^+$ of the operator $Q$, i.e. the inverse of $Q$ restricted to $\Omega_1$. The pseudo-inverse of $Q$ is obtained by solving the problem

$$
f = Q^+ g \iff Qf = g, \quad \int g \, dv = 0, \quad \int f \, dv = 0.
$$

Using the form (25) of $Q$, this gives

$$
f(x, v, t) = -\frac{g(x, v, t)}{\omega(x, \nu)} + c(x, t) W(v, x), \quad c(x, t) = \int \frac{g(x, v, t)}{\omega(v, x)} \, dv,
$$
or $Q^+ g = (P - I) \frac{g}{\epsilon}$, and (34)(d) can be written as

$$
\psi = -\epsilon(I - P) \left\{ \frac{1}{\epsilon} (I - P) \partial_x [v W \rho + v \psi] \right\}, \quad \omega(v, x) = \frac{v}{V_1(x)}
$$

which yields, in first order

$$
\psi = -\epsilon(I - P) \left\{ \frac{1}{\epsilon} (I - P) \partial_x [v W \rho] \right\} + O(\epsilon^2).
$$

Inserting this into the closure in (34)(b) gives for the closure term

$$
\int v \psi \, dv = -\epsilon \int \left[ v(I - P) \left\{ \frac{1}{\epsilon} (I - P) \partial_x [v W \rho] \right\} \right] \, dv.
$$

It remains to calculate the action of the projection operator $I - P$ on each term. We have

$$
\frac{1}{\epsilon} (I - P) \partial_x (v W \rho) = \frac{1}{\epsilon} \partial_x (v W \rho) - \frac{W}{\epsilon} \partial_x (W_1 \rho) = V_1 \partial_x (W \rho) - \frac{V_1}{v} \partial_x (W_1 \rho),
$$

and

$$
v(I - P) \left\{ \frac{1}{\epsilon} (I - P) \partial_x (v W \rho) \right\}
$$

$$
= V_1 \partial_x (v W \rho) - V_1 W \partial_x (W_1 \rho) - v W[V_1 \partial_x \rho - V_1 W \partial_x (W_1 \rho)].
$$

Note, that the first two terms on the right hand side cancel, when integrated of $v$. We obtain

$$
\int v \psi(x, v, t) \, dv = -\epsilon \int \left[ v(I - P) \left\{ \frac{1}{\epsilon} (I - P) \partial_x [v W \rho + v \psi] \right\} \right] \, dv
$$

$$
= \epsilon W_1 [V_1 \partial_x \rho - V_1 W \partial_x (W_1 \rho)].
$$

Because of the relations (30) between the moments $V_j$ and $W_j$, we obtain

$$
\int v \psi(x, v, t) \, dv = \epsilon W_2 [\partial_x \rho - W \partial_x (W_1 \rho)] = \epsilon W_2 [(1 - W_1 W_1) \partial_x \rho - \rho W \partial_x W_1].
$$

Comparing this to

$$
F = C \rho - D \partial_x \rho = W_1 \rho + \int v \psi \, dv
$$
PDE MODELLING VIA A KINETIC APPROACH

gives
\[ C\rho - D\partial_x\rho = W_1\rho + \varepsilon W_2(1 - W_{-1}W_1)\partial_x\rho - \rho W_{-1}\partial_x W_1. \]

Comparing coefficients gives
\[ C = W_1 - \varepsilon W_2 W_{-1} \partial_x W_1, \quad D = \varepsilon W_2(W_{-1}W_1 - 1). \]

The diffusion coefficient \( D \) is always non-negative, since it can be written as a variance, i.e. we have
\[ D(x) = \varepsilon W_2 W_1 \int \left[ \frac{1}{v} - \frac{1}{W_1} \right]^2 v W(v,x) \, dv \geq 0. \]

Hence, we obtain the convection and diffusion coefficients and the asymptotic conservation law given by
\[ \partial_t \rho + \partial_x F = 0, \quad F = C\rho - D\partial_x \rho, \quad C = W_1 - \varepsilon W_2 W_{-1} \partial_x W_1, \quad D = \varepsilon W_2(W_{-1}W_1 - 1). \]

(36)

The conservation law (36) can be used for modeling complex re-entrant production systems successfully.

4. Parameter extraction. In this section, we describe how to compute the transport coefficients, i.e. the convection and the diffusion coefficients from the velocity distribution \( V \) (see Section 3.1 for the derivation) for the macroscopic model dependent on stage \( x \) and time \( t \).

According to the fluid dynamics theory, the mean and the variance of the distribution \( V \) depending on the macroscopic state variable \( Z \), obtained from the observations (DES), are related respectively to the average velocity coefficient \( C(x) \) and the diffusion coefficient \( D(x) \) in a fluid dynamics approximation of the form
\[ \partial_t \rho + \partial_x F = 0, \quad F = C\rho - D\partial_x \rho \]
where \( \rho = \rho(x,t) \) represents density of the products dependent on stage and time, and \( F(\rho(x,t)) \) represents the product flow (flux).

Using the theory developed in Section 3.3, we relate the velocity \( C \) and the diffusion \( D \) coefficients to the mean and variance of the distribution \( V \). After normalization, formulas for the velocity \( C \) and the diffusion \( D \) are respectively given by
\[ C = \frac{1}{M} \langle v \rangle_m, \quad D = \frac{1}{M} \langle v \rangle^2_m \Omega \]
where \( \frac{1}{M} \) denotes the length of space interval from one stage to the other and \( \Omega = \langle v \rangle^2_m \sigma_m^2 \) denotes the variation coefficient, which is a measure of stochasticity. As \( \Omega \) gets large, the stochasticity increases; and as \( \Omega \) gets smaller, the system becomes more deterministic. The derivation of velocity coefficient is obviously obtained from the division of average distance by average time; whereas diffusion coefficient is derived from the particle model by long time averaging, which is known as “Chapman - Enskog expansion”. The proof and the details of the derivations of the formulas can be found in Section 3.3.

Note that the PDE given in (1) represents in general a nonlinear problem since the transport coefficients \( C \) and \( D \) depend on the state variable \( Z \) which, in turn, will depend on the density \( \rho \).
5. A model problem. We carry out a numerical experiment in order to compare a complex system to the PDE model. The purpose of this comparison is to substantiate the agreement in terms of WIP levels and fluxes as functions of time at each machine stage.

As a semi-realistic example, we use a simplified model of a semiconductor fab. This model consists of nine machine clusters, each performing tasks such as etching, lithography and deposition. Each cluster consists of multiple machines. So, there is a total of 200 machines in the system. The simplified production process consists of 26 stages, i.e. each part has to visit an individual cluster multiple times, and we are dealing with a highly re-entrant production system. The re-entrant structure in semiconductor manufacturing arises from the fact that chips are produced in layers. So, each time a layer is deposited, the same wafer has to go through the sequence of lithography - deposition - etching again.

For a detailed explanation of a simplified semiconductor fab model, we refer the reader to [11].

For the purpose of demonstration, we replace the observed system by a discrete event simulation. We take an influx profile that we get out of the averaged DES models and then we try to predict the non-equilibrium behavior using the PDE model. In the simulations, we compute the transport coefficients depending on a macroscopic state variable \( Z \), which is the total WIP since we use FIFO (first in first out) as the service rule in the system. One can also use PULL or PUSH policies as service rules; then it would be wise to use upstream WIP or downstream WIP in the simulations.

We execute the DES model by generating a total of 900 lots with a varying influx. In Figure 1, the averaged influx computed out of 200 discrete event simulations is seen. We compute the data for the observations from a steady state situation and then predict the system in a non-steady state regime as can be seen by the transients in Figures 4 and 5.

Velocity coefficients for the model of semiconductor fab is shown in Figure 2 both in contour and log scale plots as functions of WIP for each stage. Moreover,
After estimating the velocity and the diffusion coefficients, which are extracted from the ‘observed data’ as outlined in Section 3, we numerically solve the PDE given in (1) using an upwinding scheme where the flux is given as $F = C\rho - D\partial_x \rho$.

With the PDE model, we predict the transient behavior of the DES model according to the influx given in Figure 1 that we get out of the average of 100 DES model runs. The total time to execute this DES model takes almost one day, whereas the time to simulate the corresponding PDE model takes around one hour. The execution in this example is carried out by MATLAB 5.01 and obtained by a PC Pentium 4, 2.00 GHz, 1 GB of RAM.

The comparisons of all DES and PDE fluxes as functions of time at each machine stage can be seen in 3D surface plots in Figure 4. Similarly, the WIP levels of DES and PDE models are compared as functions of time at each machine stage in 3D surface plots in Figure 5. In the simulations, each time step corresponds to 120 mesh units depending on what time unit is chosen; i.e. minutes, hours, or days, etc.

As it can be seen in Figures 4 and 5, we substantiate the agreement of PDE and DES models by comparing the WIP levels and flux values of the two models at each stage. In addition, at two randomly chosen sample stages (8 and 23), we compare three averaged discrete event experiments to the PDE model in 2D plots in terms of WIP levels as functions of time. It is seen in Figures 6 and 7 that, in each plot, all the curves fall in the same imaginary envelope. Therefore, using the PDE model we are able to predict the transient behavior of the given system with the macroscopic model under conditions different than those used to collect the data.
Conclusion. In this paper, we have presented a computationally efficient way to simulate stochastic production systems. Our goal was to solve the diffusive PDE model (1) after extracting the transport coefficients ($C$ and $D$) from the observed data which we obtained out of a discrete event experiments for the purpose of this paper. With the PDE model, using a MATLAB program, we predicted the transient behavior of the DES model according to the averaged influx obtained out of 100 discrete event experiments. Finally, we compared the PDE and DES models
in terms of WIP levels and fluxes at each machine stage as functions of time and substantiated the agreement of two models. PDE models are preferable to simulate large and complex production systems, because they are accurate, approximately twenty times faster than DES models in execution of simulations, and they can be optimized via constraint optimization, which is a very important advantage for a mathematical model as compared to other models.
Figure 7. WIP Level Comparison of PDE and DES models at Machine Stage: 23

REFERENCES

[1] D. Armbruster, D. Marthaler and 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) (eds. G. Mackulak et al.), 365–369.

[2] D. Armbruster, D. Marthaler and C. Ringhofer, Kinetic and fluid model hierarchies for supply chains, SIAM J. on Multiscale Modeling, 2 (2003), 43–61.

[3] D. Armbruster and C. Ringhofer, Thermalized kinetic and fluid models for re-entrant supply chains, SIAM J. Multiscale Modeling and Simulation, 3 (2005), 782–800.

[4] D. Armbruster, C. Ringhofer and T. Jo, Continuous models for production flows, in Proceedings of the 2004 American Control Conference, Boston, 4589–4594.

[5] J. Banks, J. Carson and B. Nelson, Discrete Event System Simulation, Prentice Hall, 1999.

[6] S. Brush, Kinetic Theory, Pergamon Press, 1972.

[7] C. Cercignani, Boltzmann Equation and its Applications, Applied Mathematical Sciences, 67, 1988.

[8] G. Fishman, Discrete-Event Simulation, Springer-Verlag, New York, 2001.

[9] D. Helbing, Traffic and related self-driven many-particle systems, Reviews of Modern Physics, 73 (2001), 1067–1141.

[10] J. Nocedal and S. Wright, Numerical Optimization, 2nd edition, Springer-Verlag, New York, 2006.

[11] A. Unver and C. Ringhofer, Estimation of transport coefficients in re-entrant factory models, 15th IFAC Symposium on System Identification, SYSID France, (2009), 705–710.

[12] A. Unver, C. Ringhofer and D. Armbruster, A hyperbolic relaxation model for product flow in complex production networks, AIMS Proceedings, (2009), 790–799.
A. Unver, Observation Based PDE Models for Stochastic Production Systems, Thesis (Ph.D.), Arizona State University, 2008.

Received May 2015; revised November 2015.

E-mail address: ali.k.unver@intel.com
E-mail address: ringhofer@asu.edu
E-mail address: mekoksal@omu.edu.tr