Delft University of Technology

Advance Modeling

WI4204

Modelling and Optimization of Supply Chain Based on Traffic Flow Models

Authors:
Hangyu Xia (5914779)
Hongrui Liu (6005608)
Wei Fan (5916704)
Yuke Wang (6062555)

June 17, 2024

Delft University of Technology

Abstract

This report focuses on single-chain structure supply chains, derives and analyzes both discrete and continuous models. The discrete model uses discrete event simulation to capture the exact dynamics of supply chains, whereas the continuous model uses the traffic flow model to approximate these dynamics for large-scale systems. By comparing the accuracy of both models, the study demonstrates that the continuous model can achieve results that closely resemble those of the discrete model while maintaining a high level of accuracy. The report further explores optimization techniques to minimize holding costs and enhance supply chain performance. Numerical experiments validate the theoretical findings, showcasing the practical implications of the proposed models in real-world supply chain scenarios.

Keywords: Supply Chain, Traffic Flow Models, Discrete Event Simulation Models, Optimization

Contents

1 Introduction

A supply chain is a network of suppliers that produce goods, both, for one another and for generic customers. Goods travel from origin suppliers to destination customers, possibly visiting intermediate suppliers and being altered or recombined in the process. Conservation rules at each supplier define its outputs as a function of its inputs. The rates at which goods of different types flow over this network depend on the customer demand, the flow of information (orders) across suppliers, and on the algorithms that the suppliers use to place orders and replenish their inventories [12].

This report is concerned with the development and analysis of continuum models for single-chain structure supply chains. We consider a chain of M suppliers or processors $S_0, \ldots, S_{M-1}$. In the generic picture of a supply chain, each supplier processes a certain good (measured in units of parts) and passes it on to the next supplier in the chain. By labeling the parts with the index n, we denote the time at which part number n transitions from supplier number $m-1$ to supplier number m as $\tau(m, n)$. The objective of supply chain modeling and control is to establish rules that govern the evolution of the times $\tau(m, n)$ and subsequently design these rules to optimally manage the supply chain according to a predefined criterion.

For this purpose, there are a range of different models available. If the times $\tau(m, n)$ is used as primary variables, and therefore each part is considered individually, this leads to so-called discrete event simulation models. Banks, Carson and Nelson[7], which studied the most exact, and computationally most expensive, simulation tool. Based on this model, there are several expected benefits from a proper supply chain configuration, which include better coordination of material and capacity, reduced order cycle time, decrease in inventory cost and bullwhip effect [16], transport optimization, and increased customer responsiveness [10]. An analysis of the recent literature shows that the problem of optimizing supply chain design is approached by researchers either with linear programming (operational research) models or through simulation models. Yan et al. [19], Tiwari et al. [18], Bashiri and Tabrizi [8] encompassed determining the number, location, and capacity of distribution centers, minimizing the total cost, or maximizing the profit of the supply chain. Furthermore, problems such as supplier selection or technology management can be included in the model (e.g., Hammami et al. [15]; Goti [14]).

On the other end of the spectrum lie, so-called fluid models, which replace the individual parts with a continuum and use rate equations for the flow of product through a supplier (see [1, 9] for an overview). Armbruster et al. [2] passed to the limit in discrete models, obtained the conservation law:

$$\rho_t + (\min\{\mu(t, x), \rho\})_x = 0,$$

where ρ is the density of objects processed by the supply chain (represented by a real line) and μ is the processing rate. Based on this research, D'Apice and Manzo [13] proposed a mixed continuum-discrete model and discussed possible choices of solutions at nodes that guaranteed the conservation of fluxes. By fixing a rule, they also defined a Riemann solver and proved the existence of solutions to Cauchy problems.

When dealing with a large number of parts, fluid models are considerably less costly but inherently offer only an approximation of the real scenario. Traffic flow models have garnered significant attention as a middle ground between these extremes. This term stems from the analogy of parts moving like vehicles on a highway, leveraging a well-established theoretical framework for modeling traffic flows. This theory utilizes the methodologies from an even older and more refined field, namely gas dynamics. Consequently, discrete event simulation replaces particle-based models (such as Monte Carlo methods), which the equations of gas dynamics can approximate, and so forth. Naturally, this analogy is not exact since the fundamental rules governing the parts in a supply chain, the cars on a highway, and the molecules in a gas differ [4, 5, 6, 11, 17].

This report 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 [3]. The N-curve $U(t)$ at supplier S_m is given by the number of parts that have passed from processor S_{m-1} to processor S_m at time t, i.e., by

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

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.,

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

which 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.,

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

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

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

for the WIP $W(m,t)$ and the flux $F(m,t)$, both given in terms of the transition times $\tau(m,n)$. The traffic flow model replaces the WIP W and the flux F with continuous functions and eliminates the dependence on individual parts. Instead of imposing artificial constraints on the fluxes, we propose a continuum model that incorporates a service rate μ as an input parameter. In this model, the work-in-progress, denoted by $W(m,t)$, is ensured to remain nonnegative. The foundation of our model lies in straightforward assumptions: each supplier operates as an individual processor characterized by a processing time T and is equipped with a buffer queue ahead of it. From these premises, we proceed to establish, in the limit of a continuum, a conservation law of the form

$$\partial_t \rho + \partial_x \min\left\{\mu, \frac{W}{T}\right\} = 0, \tag{5}$$

where the artificial continuous variable x indexes the suppliers and the $\rho(x,t)$ denotes the product density over x, i.e., $W = \int \rho dx$ holds. When considering a large number of parts, solving the conservation law (5) becomes significantly more efficient than directly computing $\tau(m,n)$.

This report is based on the paper [3] by Armbruster, Degond, and Ringhofer. The structure of this report is as follows. In Section 2, we will list the important notations and their meanings used in the following sections. In Section 3, we will introduce the theoretical aspects of the Discrete Model in Section 3.1 and the Continuous Model in Section 3.2. In Section 4, we will present the numerical experiments for these two models. In Section 5, we will optimize the models and provide the corresponding results. In Section 6, we will summarize and reflect on the work.

2 Notation

In this section, we specify notations in our report.

Notation	Meaning
$\tau(m,n)$	Arriving time of n product at m supplier
$\mu(m,n)$	Processing rate of supplier m when processing product n
$T(m)$	Processing time of supplier m, identical to all product
T_0	Average processing time $T_0 = \frac{1}{M}\sum_{m=0}^{M-1} T(m)$
$C(m)$	Processing capacity of supplier m
x_m	Mesh of supplier m
h_m	Interval of mesh $h_m = x_{m+1} - x_m$
Δx_m	Spatial meshsizes when adding virtual nodes
X	Length of degree of completion (DOC) interval $x_M = X$
$f(x_m, t)$	flux of supplier m at time t
$\rho(x_m, t)$	density of supplier m at time t

Table 1: Notations

3 Model

In order to model supply chain networks mentioned in the previous section, in the section, we construct the dynamics of supply chain networks in two different perspectives, namely queuing methods and traffic flow methods. For simplicity, we focus on a single supply chain consisting M heterogeneous suppliers processing N products independently.

3.1 Discrete Model

For a single supply chain consisting m nodes (suppliers), each nodes can be regarded as a queue system with products waiting and being processing in a fixed order. We first present the queuing system model for a single node in the supply chain. We assume a single nodes having two parts: a "processor" to process products at a rate μ with capacity C and "buffer" to accommodate products waiting for the "processor". For a single product n , $n = 1, 2, ..., N$, we denote several important time point relating to supplier m:

- a_n the time point product n arrives at the end of the queue of supplier m.
- b_n the time point product n leaves the queue and is fed into the "processor".
- e_n the time point product n leaves supplier m.

If the queue is full, in other words, product n has already arrived when it is fed into the "processor" as $a_n \le b_{n-1} + \frac{1}{\mu(m,n-1)}$ holds. While if the queue is empty, on the other hands, product n will be fed into the "processor" immediately it arrives as $a_n > b_{n-1} + \frac{1}{\mu(m,n-1)}$ holds. These will derive the relation

$$b_n = \max\left\{a_n, b_{n-1} + \frac{1}{\mu(m,n-1)}\right\}.$$

If we assume the processor m takes time $T(m)$ to process a product, which is identical for all products, we can derive the relation of e_n.

$$e_n = \max\left\{a_n + T(m), e_{n-1} + \frac{1}{\mu(m,n-1)}\right\}.$$

Recalling that $\tau(m,n)$ represents the time product n arrives at supplier m. It is obvious that $a_n \to \tau(m,n)$ and $e_n \to \tau(m+1,n)$. As a result, we can obtain the recursive equation for the queue system

$$\tau(m+1,n) = \max\left\{\tau(m,n) + T(m), \tau(m+1,n-1) + \frac{1}{\mu(m,n-1)}\right\}, \quad m = 0,1,...,M, \quad n \geq 1. \tag{6}$$

with initial value and boundary conditions are specified as

$$\tau(0,n) = \tau^A(n) \quad n \geq 0, \quad \tau(m,0) = \tau^I(m) \quad m = 0,1,...,M.$$

With (6) we can calculate the exact solutions, the time point each product arrives and leaves each supplier.

3.2 Continuous Model

The continuous model assumes that each supplier operates as a single processor with a processing time T and a buffer queue in front of it. Based on this assumption, we derive a conservation law in the continuum limit, which is (5). The artificial continuous variable x represents the suppliers, and $\rho(x,t)$ denotes the product density over x, ensuring that $W = \int \rho\, dx$ holds true. When the number of parts is very large, solving the conservation law (5) is clearly more efficient than directly computing $\tau(m,n)$.

However, Armbruster et al. [3] shows that the conservation law (5) is only satisfied asymptotically in the limit of a large number of suppliers . The main difficulty here is that the conservation law (5) generally has only distributional solutions. Specifically, $\rho(x,t)$ can develop δ-function concentration, which correspond to bottlenecks in the supply chain.

To address this problem, we derive the corresponding hyperbolic equation for the N-function in (1). This approach allows us to numerically compute the distributional solutions of (5) in a more manageable way, under the assumption of a large number of nodes in the supply chain.

3.2.1 Constitutive Law of Asymptotic Density and Flux

In order to asymptotically replace the exact solution of (6) by conservation law with a simple constitutive relation, we first find an appropriate form of the constitutive relation given in (3). At any arbitrary time τ which we assume is continuous, writing as $\tau(x,y)$ instead. The N-function defined previously satisfy the relations

$$\begin{aligned}
\frac{d}{dy}U(x,\tau(x,y)) &= \partial_\tau U(x,\tau(x,y))\partial_y \tau(x,y) = 1, \\
\frac{d}{dx}U(x,\tau(x,y)) &= \partial_x U(x,\tau(x,y)) + \partial_\tau U(x,\tau(x,y))\partial_x \tau(x,y).
\end{aligned}$$

The partial derivation with respect to y equals to a constant implies that $\frac{d}{dy}U(x,\tau(x,y))$ is a function only depends on x. So for continuous arriving time $\tau(x,y)$, it is reasonable to set flux $f(x,\tau(x,y)) = \frac{1}{\partial_y \tau(x,y)}$ and density $\rho(x,\tau(x,y)) = \frac{\partial_x \tau(x,y)}{\partial_y \tau(x,y)}$. It is easy to verify that conservation law $\partial_t \rho + \partial_x f = 0$ holds for the obtained flux and density. Then we model the single supply chain mentioned in the previous section into a mesh $0 = x_0 < x_1 < ... < x_M = X$, where x_m represents the location of supplier m. The continuous $\tau(x,y)$ degraded into discrete $\tau(m,n)$, with

$$\begin{aligned}
\partial_y \tau(x,y) &\approx \Delta_n \tau(m,n) = \tau(m,n+1) - \tau(m,n), \\
\partial_x \tau(x,y) &\approx \frac{\Delta_m \tau(m,n)}{h_{m-1}} = \frac{\tau(m+1,n) - \tau(m,n)}{x_m - x_{m-1}}.
\end{aligned}$$

Therefore, we can define the approximate density ρ and the approximate flux f from any arbitrary time $\tau(m,n)$ by

$$f(x_m,\tau(m,n)) = \frac{1}{\Delta_n\tau(m,n)} = \frac{1}{\tau(m,n+1)-\tau(m,n)}, \tag{7}$$

$$\rho(x_m,\tau(m+1,n)) = \frac{\Delta_m\tau(m,n+1)}{h_m\Delta_n\tau(m+1,n)} = \frac{\tau(m+1,n+1)-\tau(m,n+1)}{(x_{m+1}-x_m)(\tau(m+1,n+1)-\tau(m+1,n))}. \tag{8}$$

The approximate flux f in (7) derived and approximate density ρ in (8) construct an approximate or discretized version of conservation law specified in (5), and also provide a simple constitutive relation of the form $f = f(\rho)$, as illustrated in Theorem 3.1 in [3].

Theorem 3.1. Let the arrival times (m,n) satisfy the recursion (6). Let the approximate density and flux f be defined by (7) and (8). Then the approximate flux can be written in terms of the approximate density via a constitutive relation of the form

$$f(x_m,\tau(m,n)) = \phi(\rho(x_{m-1},\tau(m,n))), \quad m = 1,2,...,M, \quad n \geq 0$$

with the flux function ϕ_m given by

$$\phi_{mn}(\rho) = \left\{\mu(m-1,n), \frac{(x_m-x_{m-1})\rho}{T(m-1)}\right\}. \tag{9}$$

3.2.2 Conservation Law of Asymptotic Density and Flux

Beside the constitutive law, the the approximate density ρ and the approximate flux f also satisfy a conservation law $\partial_t\rho + \partial_x f = 0$ asymptotically when the number of supplier nodes is large.

Using interpolation, we obtain the interpolated density and flux functions as follows.

$$f_1(x_m,t) = f(x_m,\tau(m,n)), \tau(m,n) \leq t < \tau(m,n+1), m = 0,\dots,M-1, n \geq 0, \tag{10}$$

$$\rho_1(x_m,t) = \rho(x_m,\tau(m+1,n-1)), \tau(m+1,n-1) \leq t < \tau(m+1,n), m = 0,\dots,M-1, n \geq 1. \tag{11}$$

$$f^A(t) = \frac{1}{\Delta_n\tau^A(n)}, \tau^A(n) \leq t < \tau^A(n+1). \tag{12}$$

Then the N-function $u(x_m,t)$ representing the number of parts passed from supplier S_{m-1} to S_m at time t is defined recursively by

$$u_1(x_{m+1},t) = u_1(x_m,t) - \frac{h_m}{X}\rho_1(x_m,t), m = 0,\dots,M-1, u_1(x_0,t) = \int_{\tau(0,0)}^{t} f^A(s)ds. \tag{13}$$

Furthermore, we define functions of continuous space and time as

$$f_2(x,t) = f_1(x_{m+1},t), x_m \leq x < x_{m+1}, m = 0,\dots,M-1, \tag{14}$$

$$\tau_2^I(x) = \tau^I(m+1), x_m \leq x < x_{m+1}, m = 0,\dots,M-1, \tag{15}$$

$$u_2(x,t) = u_1(x_{m+1},t), x_m \leq x < x_{m+1}, m = 0,\dots,M-1. \tag{16}$$

Finally, the following theorem holds for u_2 and f_2.

Theorem 3.2. Let the piecewise constant interpolant u_2 and f_2 be defined as in (16), (14), Let the scaled throughput times $T(x_m)$ stay uniformly bounded, i.e., $h_m = O(\epsilon)$ 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 $\rightarrow 0$ except for a certain number of nodes m and a finite number of parts n, which stays bounded as $\rightarrow 0$. Then, for $\rightarrow 0$ and $max \quad h_m \rightarrow 0$ the interpolated N-function and flux u_2, f_2 satisfy the initial boundary value problem

$$\partial_2 u_2 = f_2, t > \tau_2^I(x), 0 < x < X, \tag{17}$$

$$u_2(x, \tau^I(x)) = 0, \lim_{x \to 0_-} u_2(x,t) = \int_{\tau_2(0,0)}^{t} f^A(s)ds \tag{18}$$

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

With the Theorem 3.2, we can assert that the conservation law

$$\partial_t \rho + \partial_x f = 0 \tag{19}$$

holds asymptotically.

3.2.3 Virtual Processors

The assumptions of Theorem 3.2 indicate that the supply chain contains a large number of nodes, the number of bottlenecks is relatively small compared to the number of processors, and each processing time is short in relation to the overall throughput time. This restriction can be removed by bringing in virtual processors. The behind idea is that a processor with a processing time T and service rate μ can be decomposed into K virtual processors with the same service rate μ but smaller processing time $\frac{T}{K}$. With the increase of the number of virtual processors, the bottleneck will occur only in the first virtual processor eventually, and the queues of the additional virtual processor will always remain empty.

Theorem 3.3. Let the first processor S_0 in the chain be governed by (6). If we replace the single processor by K virtual processors with the same processing rates and the same total throughput time, i.e., by

$$\hat{\tau}(m+1,n+1) = \max\left\{\hat{\tau}(m,n+1) + \frac{T(0)}{K}, \hat{\tau}(m+1,n) + \frac{1}{\mu(m,n-1)}\right\}, \quad m = 0,1,...,K-1, \quad n \geq 0. \tag{20}$$

$$\hat{\tau}(0,n) = \tau(0,n) \quad n \geq 0, \quad \hat{\tau}(m,0) = \tau(1,0) - (1 - \frac{m}{K})T(0) \quad m = 1,...,K.$$

then we obtain the same outflux, i.e.,

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

holds.

An view of Theorem 3.3 is that it provide the conditions appropriate for the application of Theorem 3.2. For a supply chain consisting of M suppliers with processing time $T(m)$ respectively, we can make it solvable by Theorem 3.2 as following steps:

- Cut each Supplier S_m into $K(m)$ virtual processors, such that $\frac{T(m)}{K(m)}$, $m = 0,1,...,M-1$ is roughly equidistributed, giving $M_1 = \sum_{m=0}^{M-1} K(m)$ virtual processors.
- If M_1 is too small tfor the symptotic regime in Theorem 3.2 to be valid, cut each virtual processor into additional L subprocessors. Then we arrive at $M_2 = LM_1$ total processors.

However, the number of M_2 can not be as large as we like since we have used the average processing time $T_0 = \frac{1}{M}\sum_{m=0}^{M} T(m)$ to scale the service rates μ in (...). As a result, if $T_0 \rightarrow 0$, the fluxes go to 0 as well because of the scaled service rates μ. Therefore, the virtual nodes skills work well when $C_0 M_2 T_0 = C_0 \sum_{m=0}^{M} T(m) >> 1$.

4 Numerical Results

In this section, we conduct numerical experiments to verify Thoerem 3.2 by comparing the solution of (19) with the direct solution of the recursion (6) for the transition times τ. We first define the influx rate f^A by $f^A = 10 \sin \frac{\pi t}{10}$, which is also shown in Figure 1.

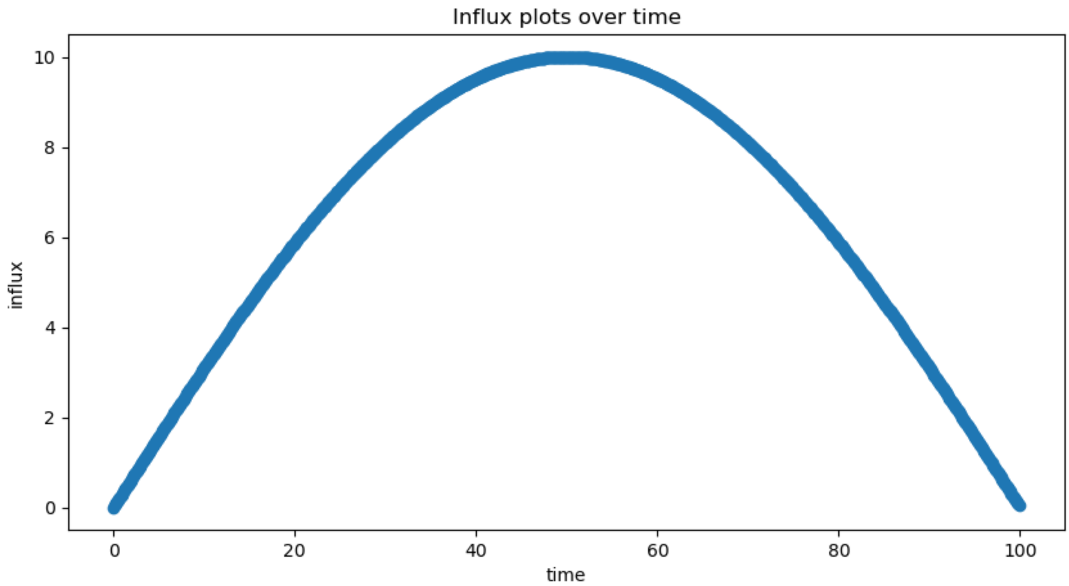


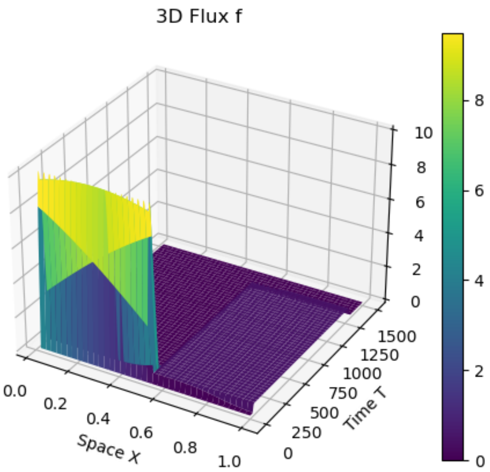
Figure 1: Influx

In the first experiment, we consider a supply chain of 2 suppliers with throughput time $T(0) = 20$, $T(1) = 20$ time units, and service rate $\mu(0) = 10$, $\mu(1) = 0.5$. We have also split the nodes S_0 and S_1 into 20 virtual nodes. Hence we can compute the flux f and density ρ of our two models by using (6). We then conduct experiments with three suppliers and four suppliers and make the same comparisons. Corresponding parameter settings are presented in Table 2.

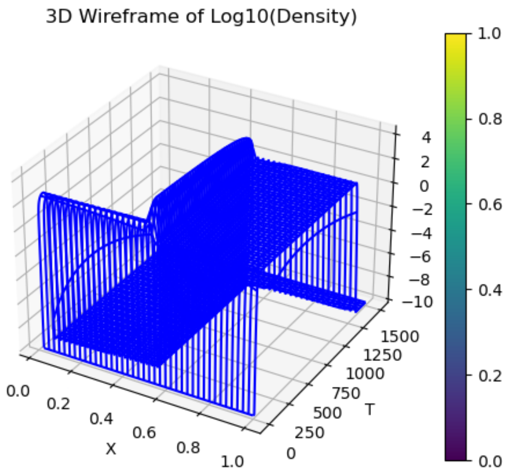
	Two suppliers	Three suppliers	Four suppliers
Processing time T	[20, 20]	[20, 20, 20]	[20, 20, 20, 20]
Service rate μ	[10, 0.5]	[10, 5, 0.5]	[10, 7.5, 5, 0.5]
Virtual nodes	[20, 20]	[20, 20, 20]	[20, 20, 20, 20]

Table 2: Parameter settings

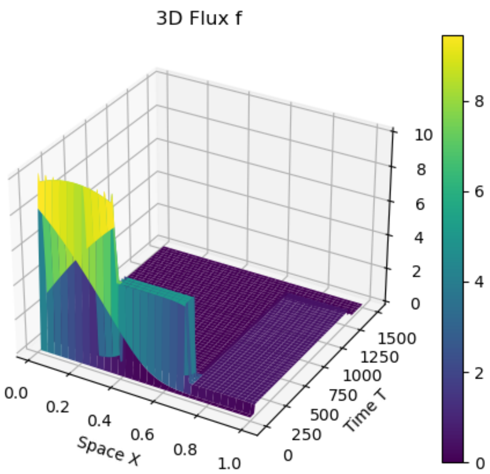
In Figure 2, We draw the flux and density plots of the PDE model for different combinations of parameters.



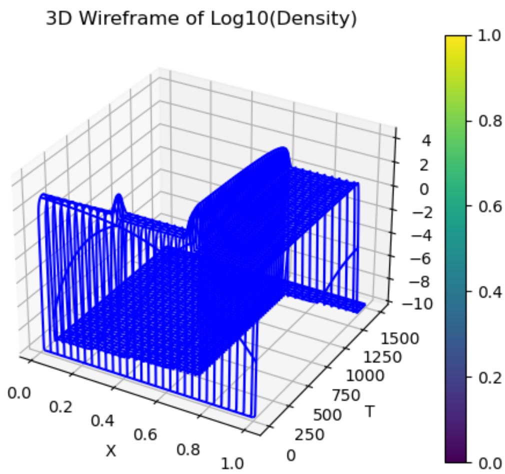
(a) Flux of 2 suppliers



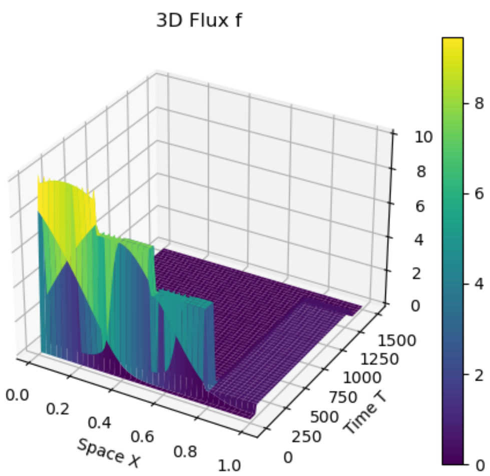
(b) Density of 2 suppliers



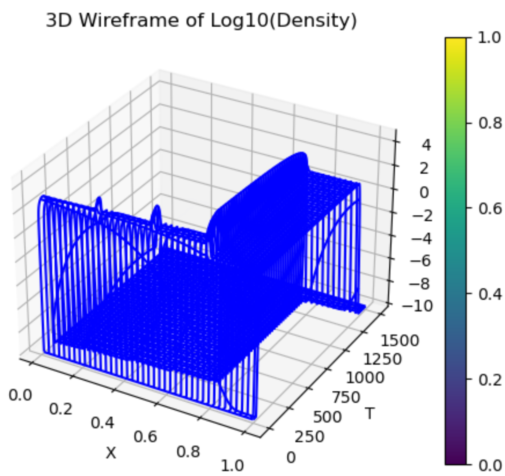
(c) Flux of 3 suppliers



(d) Density of 3 suppliers



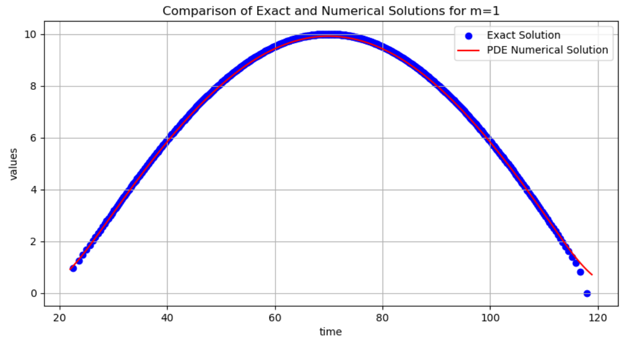
(e) Flux of 4 suppliers



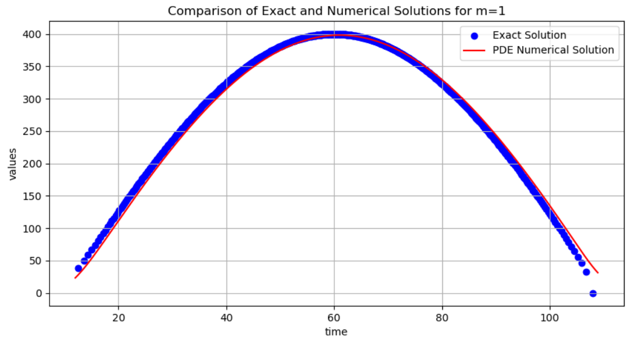
(f) Density of 4 suppliers

Figure 2: 3D flux and density of different parameter combinations

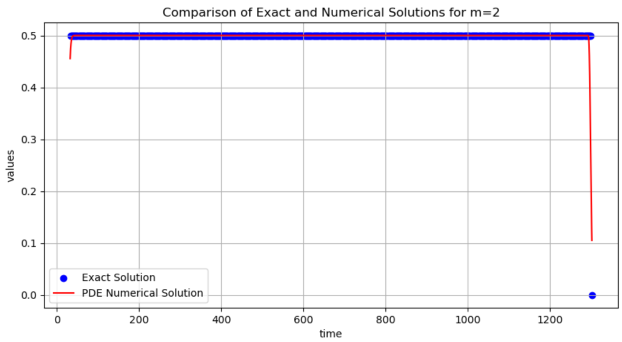
Next, we compare the asymptotic flux and density of these two models according to (7) and (8). Here we only show the case of 2 suppliers, please refer to the Appendix for Figure 7 and Figure 8 under other parameters.



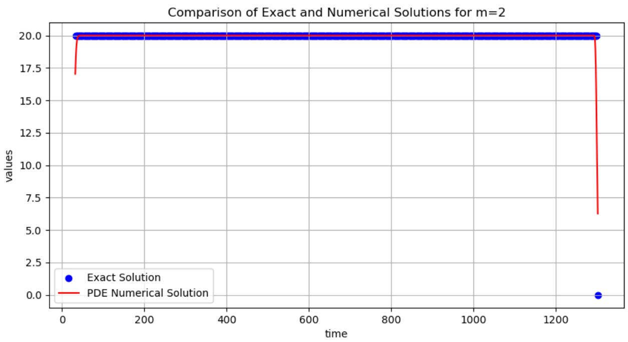
Comparison of flux in the first supplier



Comparison of density in the first supplier



Comparison of flux in the second supplier



Comparison of density in the second supplier

Figure 3: Comparison under 2 suppliers

We can see the PDE numerical solutions of flux and density are very close to these in exact solutions, which means we can use the PDE model to approximate the discrete model (6).

5 Optimization

Having a supply chain consists of suppliers $S_0, S_1, \ldots, S_{M-1}$ with process rate $\mu_0, \mu_1, \ldots, \mu_{M-1}$. For each suppler S_i, there is a holding cost coefficient H_i. Then the holding cost for each supplier S_i throughout the time $0 \to T_{final}$ is computed as $Q_i = H_i \int_0^{T_{final}} \int_0^X \rho(x,t)dxdt$. We list all the holding cost coefficients as $C_h = [H_0, H_1, \ldots, H_{M-1}]$. Assume that we have a fixed sum of all the processing rate as $\Omega = \sum_{i=0}^{M-1} \mu_i$, then we need to determine the allocation of μ_i to minimize the total holding cost.

The optimization problem of minimizing holding cost is defined as

$$\begin{aligned} \underset{\mu_0,\mu_1,\ldots,\mu_{M-1}}{\text{minimize}} \quad & \sum_{i=0}^{M-1} H_i \int_0^{T_{final}} \int_0^X \rho(x,t)dxdt \\ \text{with} \quad & \sum_{i=0}^{M-1} \mu_i = \Omega \\ & 0 \leq \mu_i < \Omega. \end{aligned}$$

Here we do the optimization by using the same parameter settings of processing time T and virtual nodes under two and three suppliers in Table 2. Furthermore, we first test different holding costs with different constraints, then we plot the variation in maintenance costs over a range of processing rate variations with different hold constants C_h, which are shown in the following figures. We only select the case of two suppliers to show, three suppliers of the case please refer to Figure 9 to Figure 15 in the Appendix.

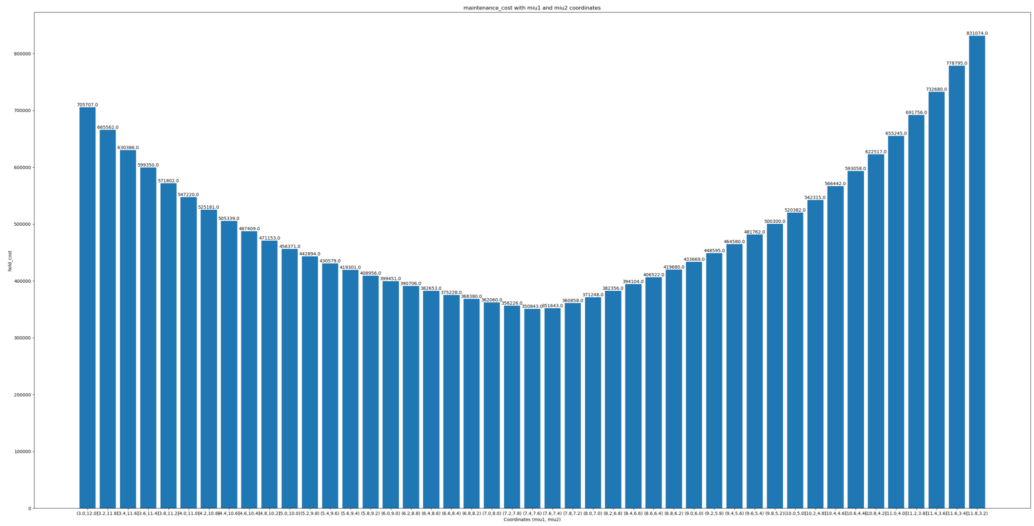


Figure 4: Two suppliers under hold constant $C_h = [10,\ 15]$

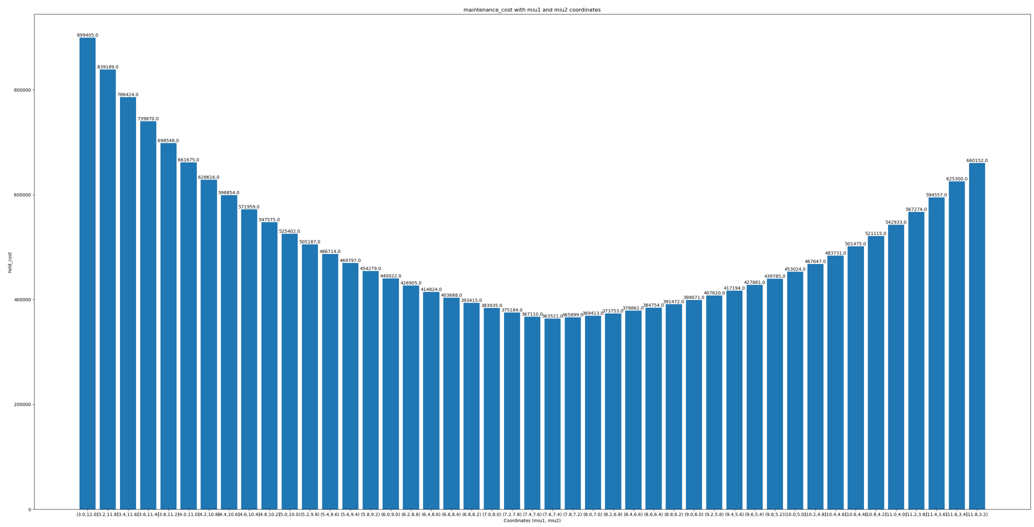


Figure 5: Two suppliers under hold constant $C_h = [15,\ 10]$

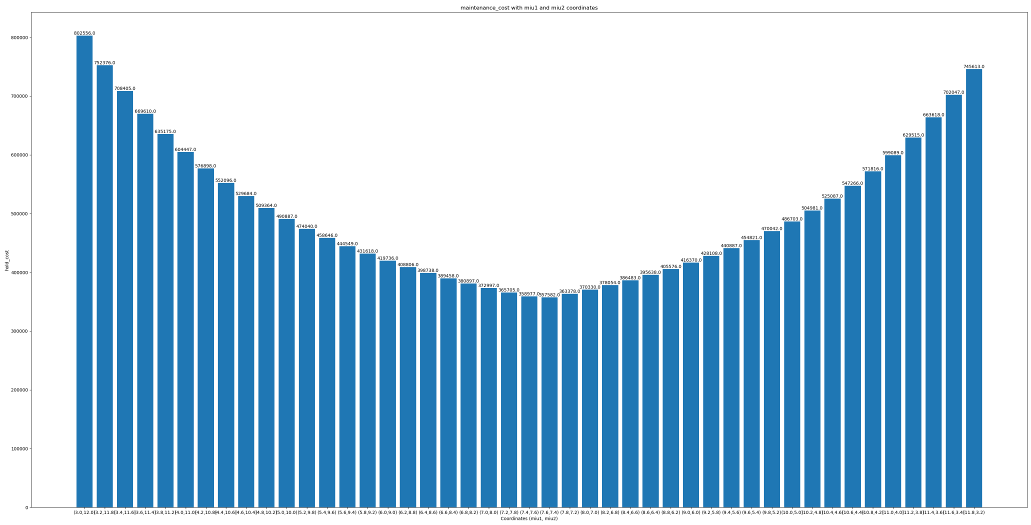


Figure 6: Two suppliers under hold constant $C_h = [12.5,\ 12.5]$

From the above figure, we can find a possible minimum maintenance cost. We then set it as the initial value to the processing rate setting for that case and use Sequential Least Squares Programming (SLSQP) to find the optimal combination of processing rates. The results are shown in Table 3 and Table 4.

Holding constant	Initial μ	Iteration number	Optimal μ	Optimal holding cost
$C_h = [10, 15]$	$[7.4, 7.6]$	12	$[7.500, 7.500]$	348315.7624
$C_h = [15, 10]$	$[7.6, 7.4]$	11	$[7.527, 7.473]$	363202.3159
$C_h = [12.5, 12.5]$	$[7.5, 7.5]$	9	$[7.501, 7.499]$	355813.7770

Table 3: Optimal processing rate μ of two suppliers by using SLSQP

Holding constant	Initial μ	Iteration number	Optimal μ	Optimal holding cost
$C_h = [16, 12, 8]$	$[9.6, 9.4, 5]$	3	$[10.98, 10.81, 2.21]$	305577.2394
$C_h = [16, 8, 12]$	$[9.8, 9.6, 4.6]$	4	$[11.07, 10.83, 2.1]$	305577.2394
$C_h = [12, 16, 8]$	$[9.6, 9.8, 4.6]$	4	$[10.91, 10.96, 2.13]$	305577.2394
$C_h = [12, 8, 16]$	$[9.6, 9.6, 4.8]$	3	$[10.91, 10.90, 2.19]$	305577.2394
$C_h = [8, 12, 16]$	$[9.5, 9.5, 5]$	3	$[10.89, 10.89, 2.22]$	305577.2394
$C_h = [8, 16, 12]$	$[9.8, 9.8, 4.4]$	4	$[10.43, 10.43, 3.14]$	305577.2394
$C_h = [12, 12, 12]$	$[9.8, 9.8, 4.4]$	3	$[10.44, 10.42, 3.14]$	305577.2394

Table 4: Optimal processing rate μ of three suppliers by using SLSQP

6 Conclusion

In this paper, we have developed and analyzed both discrete and continuous models for a 1D supply chain to enhance our understanding and management of supply chain dynamics.

1. we construct a discrete supply chain model focusing on the time $\tau(m, n)$ which item n arrives at supplier m. By establishing recursive relationships between $\tau(m, n)$, we can

effectively simulate the operations of the entire supply chain. This model allows us to trace the movement and timing of individual items through the supply chain, providing detailed insights into the logistics and scheduling aspects.

2. Subsequently, by defining the approximate flux f and density ρ, we develop a continuous supply chain model based on partial differential equations. Solving these equations enables us to compute the density and flow of items at any given time within the supply chain. This continuous approach offers a macroscopic perspective, allowing us to monitor the overall state of the supply chain without the need to track each individual item separately. This is particularly useful for large-scale supply chains where individual tracking becomes impractical.

3. Finally, we apply the continuous supply chain model to address an optimization problem aimed at minimizing holding costs. By leveraging the macroscopic insights provided by our model, we are able to determine optimal strategies for managing inventory levels and reducing costs. This demonstrates the practical utility of our continuous model in solving real-world supply chain management problems.

Overall, our work provides a comprehensive framework for modeling, analyzing, and optimizing supply chains using both discrete and continuous approaches. This enhances our ability to manage supply chains efficiently, ensuring timely deliveries and cost-effective operations.

In the future, 1D supply chain model can be extended in the following aspects.

1. More complicated parameters settings. For example, currently we use the processing rate $\mu = \mu(m)$ which only depends on the supplier S_m in our experiments. By setting $\mu = \mu(m, n)$, we can let the processing rate depend on both m and n, which makes the model more closely aligned with real-world scenarios.

2. Multi-dimensional supply chain models. Our current 1D model can be extended to multi-dimensional supply chains. This would allow for more complex and realistic simulations, capturing the interactions and dependencies across different regions and suppliers.

3. Taking stochastic elements into consideration. Introducing stochastic elements into our models could provide a more accurate representation of real-world supply chains. This would involve incorporating randomness in demand, supply delays, and other uncertainties, thereby making the models more robust and applicable to various scenarios.

References

[1] Anderson, E. J. (1981). A new continuous model for job shop scheduling. Internat. J. Systems Sci., 12:1469–1475.

[2] Armbruster, D., Degond, P., and Ringhofer, C. (2005). A model for the dynamics of large queuing networks and supply chains. submitted to SIAM J. Applied Mathematics.

[3] Armbruster, D., Degond, P., and Ringhofer, C. (2006). A model for the dynamics of large queuing networks and supply chains. SIAM Journal on Applied Mathematics, 66(3):896–920.

[4] Armbruster, D., Marthaler, D., and Ringhofer, C. (2002). A mesoscopic approach to the simulation of semiconductor supply chains. In et al., G. M., editor, Proceedings of the International Conference on Modeling and Analysis of Semiconductor Manufacturing (MASM2002), pages 365–369.

[5] Armbruster, D., Marthaler, D., and Ringhofer, C. (2004). Kinetic and fluid model hierarchies for supply chains. Multiscale Model. Simul., 2(1):43–61.

[6] Aw, A. and Rascle, M. (2000). Resurrection of "second order" models of traffic flow. SIAM J. Appl. Math., 60:916–938.

[7] Banks, J., Carson, J. S., and Nelson, B. L. (1996). Discrete-Event System Simulation. Prentice-Hall, Upper Saddle River, 2 edition.

[8] Bashiri, M. and Tabrizi, E. H. (2010). An improved solution for the capacitated multi-facility location problem using a fuzzy approach. Computers & Industrial Engineering, 59(4):708–715.

[9] Billings, R. and Hasenbein, J. (2001). Applications of fluid models to semiconductor fab operations. preprint.

[10] Chang, Y. S. and Makatsoris, H. (2001). Supply chain modeling using simulation. International Journal of Simulation Systems, Science & Technology, 2(1):24–30.

[11] Daganzo, C. (1995). Requiem for second order fluid approximations of traffic flow. Transport. Res. B, 29:277–286.

[12] Daganzo, C. (2003). A theory of supply chains. Springer Science & Business Media.

[13] D'Apice, C. and Manzo, R. (2006). A fluid dynamic model for supply chains. Networks and Heterogeneous Media, 1(3):379–398.

[14] Goti, A., editor (2010). Discrete Event Simulations. IntechOpen, Rijeka.

[15] Hammami, R., Frein, Y., and Hadj-Alouane, A. B. (2009). Supply chain design in the delocalization context: Relevant features and new modeling tendencies. International Journal of Production Economics, 122(1):183–193.

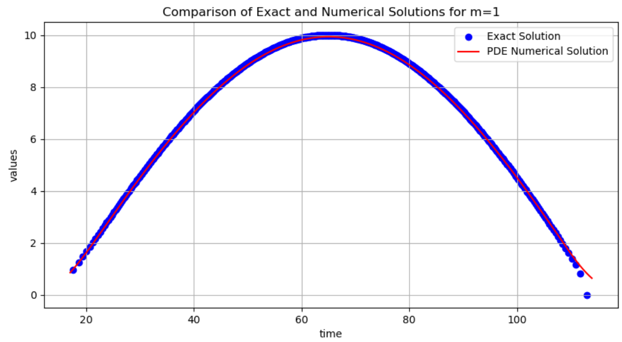
[16] Lee, H. L., Padmanabhan, V., and Whang, S. (2004). Information distortion in a supply chain: The bullwhip effect. Management Science, 50(12):1875–1886.

[17] Lighthill, M. and Whitham, J. (1955). On kinematic waves, i: Flow movement in long rivers, ii: A theory of traffic flow on long crowded roads. Proc. R. Soc. Lond. Ser. A Math. Phys. Engrg. Sci., 229:281–345.

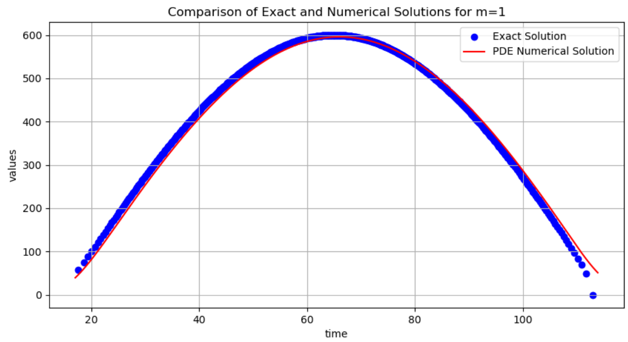
[18] Tiwari, M. K., Changdar, C., and Mishra, P. (2010). A lagrangian heuristic for the capacitated multi-facility location problem with single-source constraint. Computers & Industrial Engineering, 58(1):1–12.

[19] Yan, H., Yu, Z., and Cheng, T. C. E. (2003). A hybrid genetic algorithm for the multi-depot vehicle routing problem. European Journal of Operational Research, 151(1):111–124.

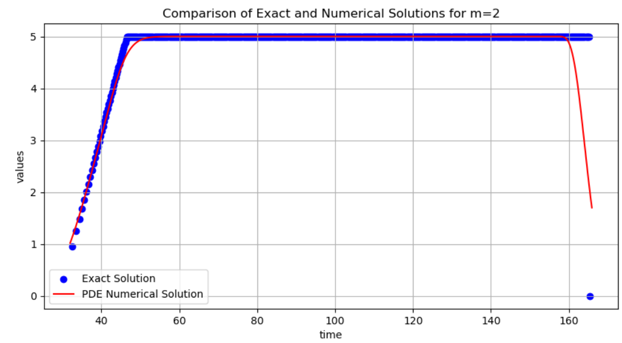
Appendix



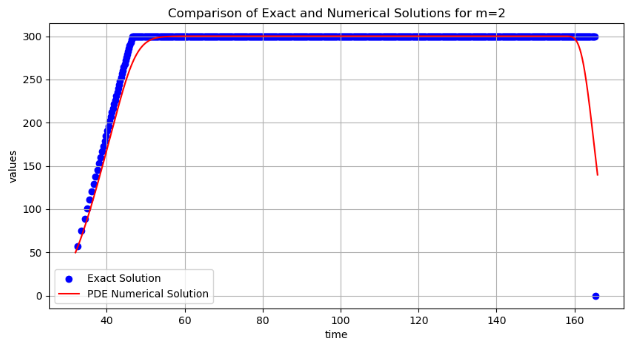
(a) Comparison of flux in the first supplier



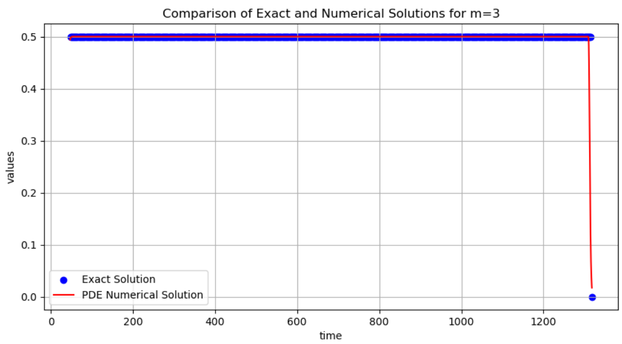
(b) Comparison of density in the first supplier



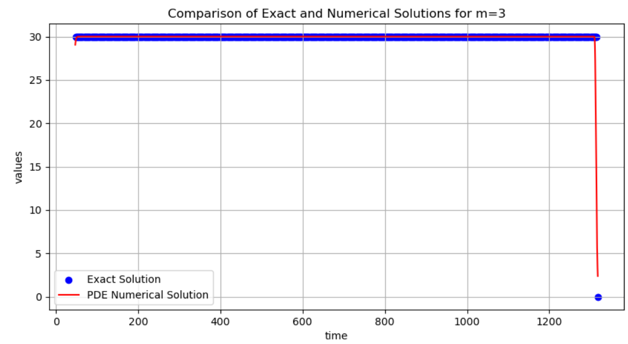
(c) Comparison of flux in the second supplier



(d) Comparison of density in the second supplier

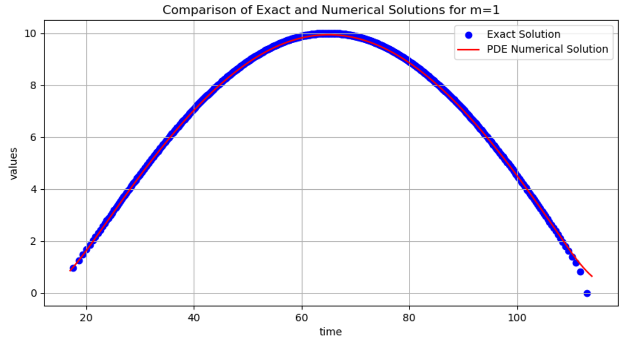


(e) Comparison of flux in the third supplier

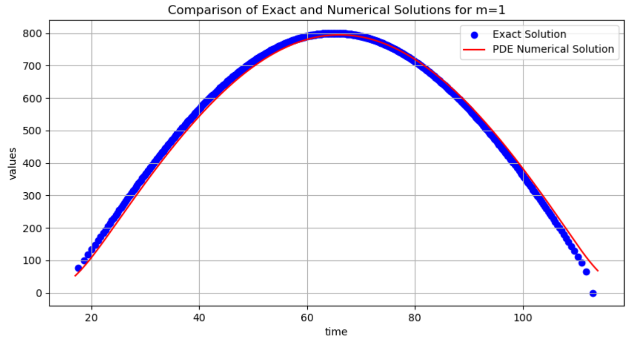


(f) Comparison of density in the third supplier

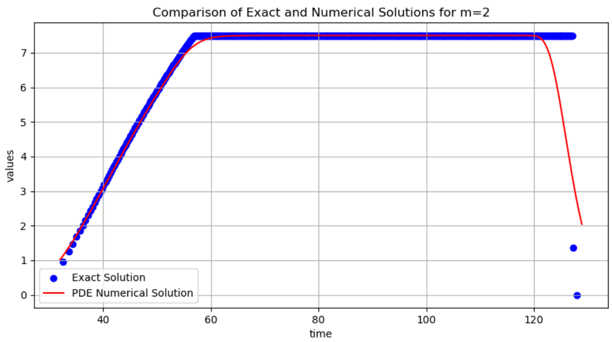
Figure 7: Comparison under 3 suppliers



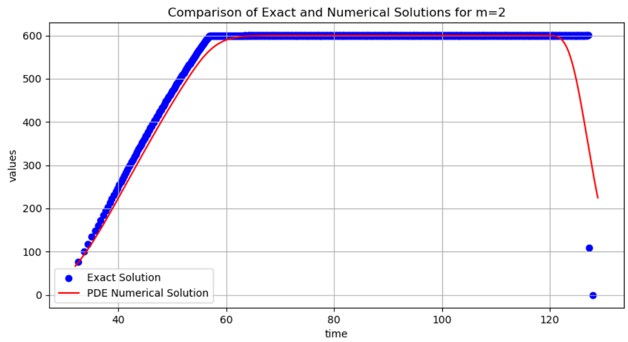
(a) Comparison of flux in the first supplier



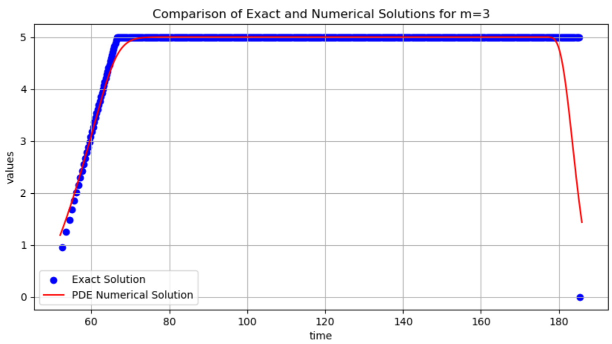
(b) Comparison of density in the first supplier



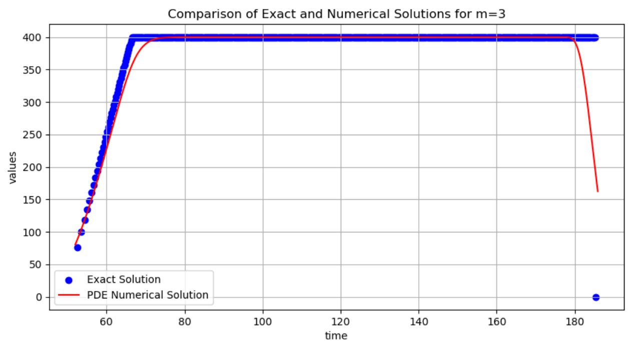
(c) Comparison of flux in the second supplier



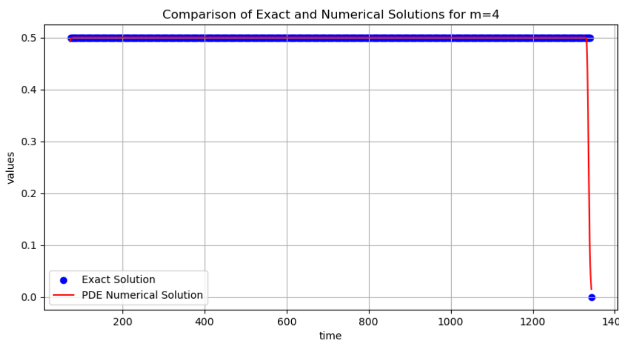
(d) Comparison of density in the second supplier



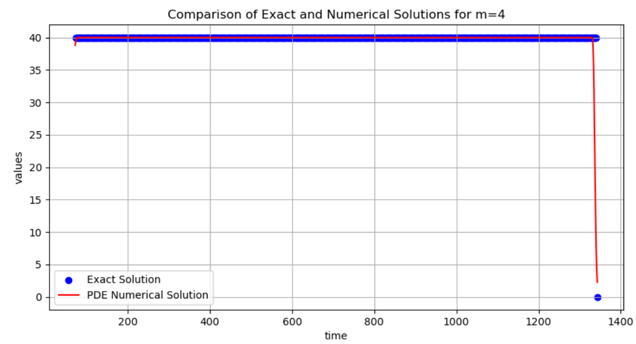
(e) Comparison of flux in the third supplier



(f) Comparison of density in the third supplier



(g) Comparison of flux in the fourth supplier



(h) Comparison of density in the fourth supplier

Figure 8: Comparison under 4 suppliers

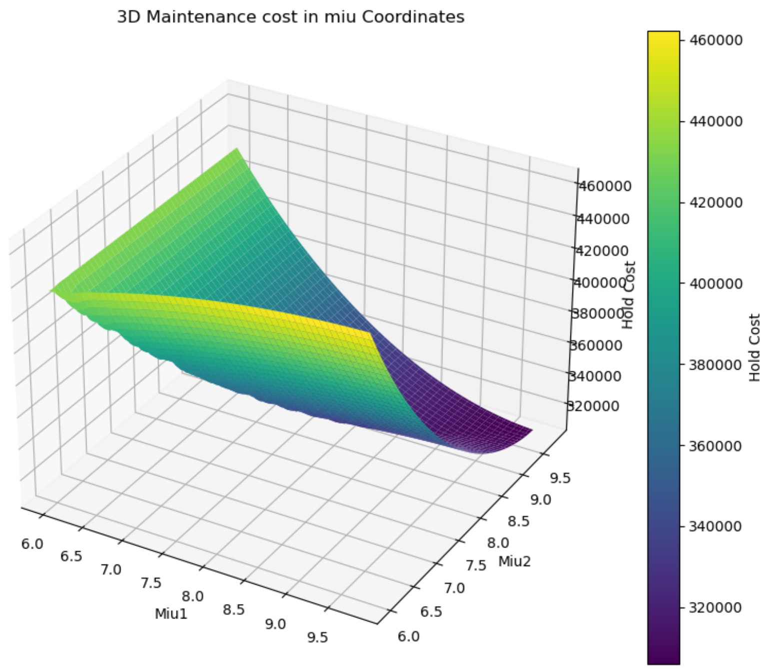


Figure 9: Three suppliers under hold constant C_h = [16, 12, 8]

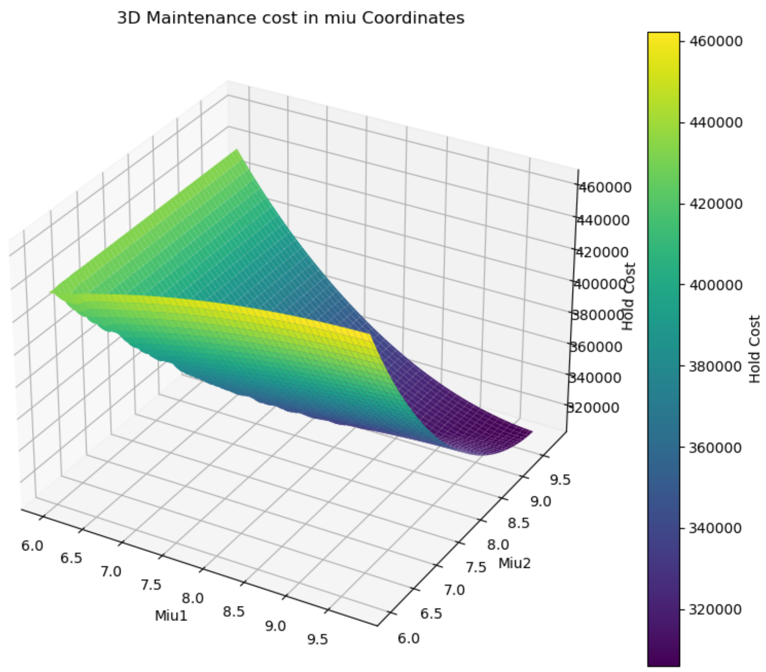


Figure 10: Three suppliers under hold constant C_h = [16, 8, 12]

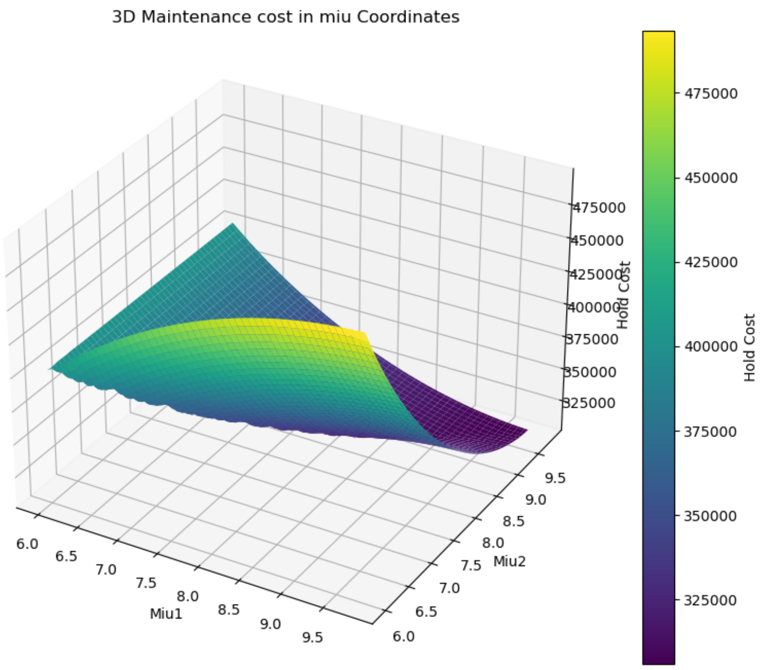


Figure 11: Three suppliers under hold constant C_h = [12, 12, 12]

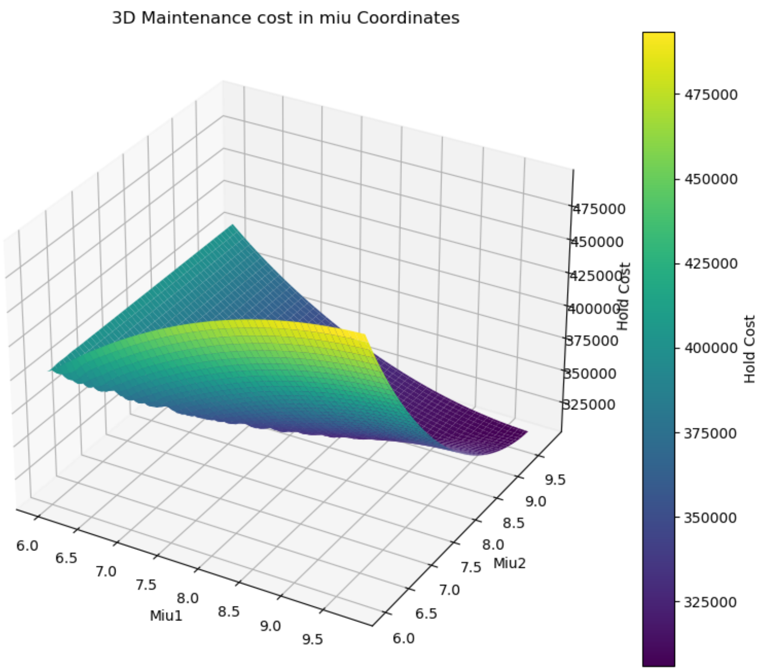


Figure 12: Three suppliers under hold constant C_h = [12, 16, 8]

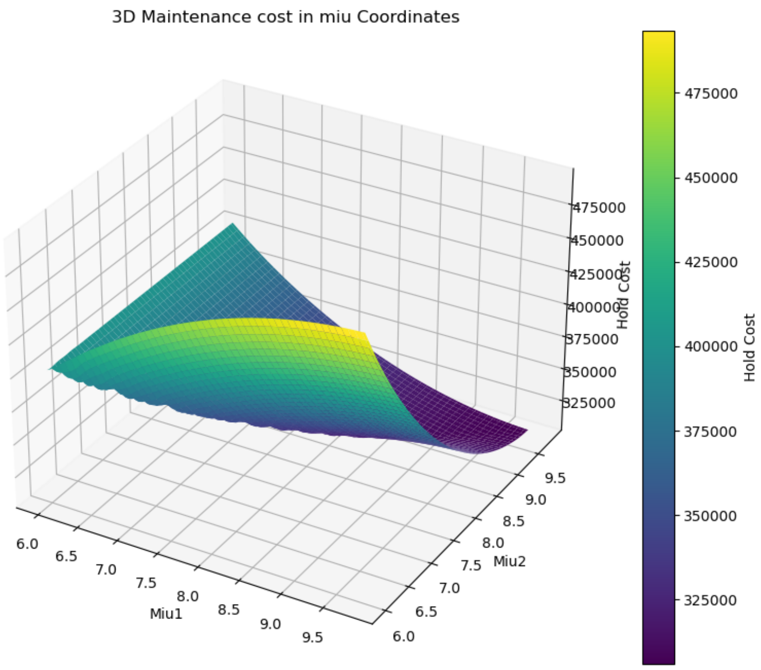


Figure 13: Three suppliers under hold constant C_h = [12, 8, 16]

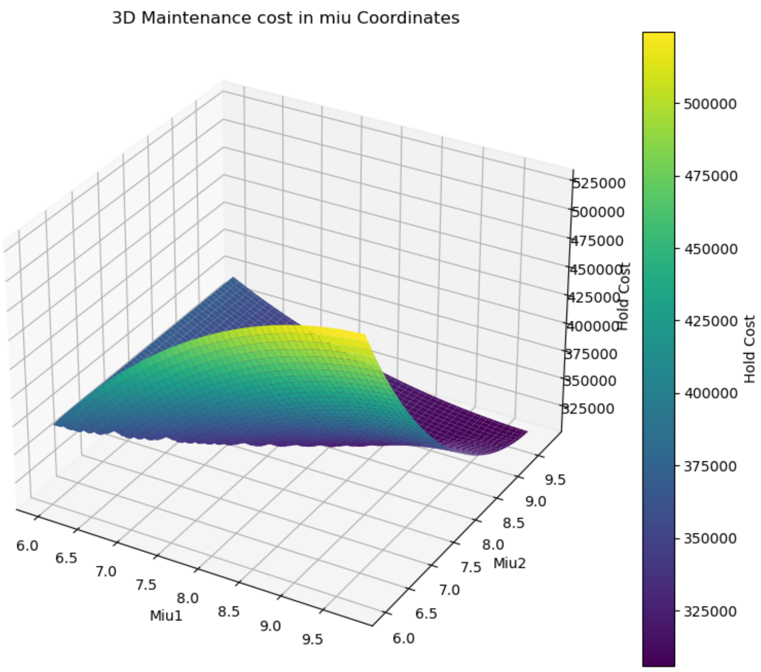


Figure 14: Three suppliers under hold constant C_h = [8, 12, 16]

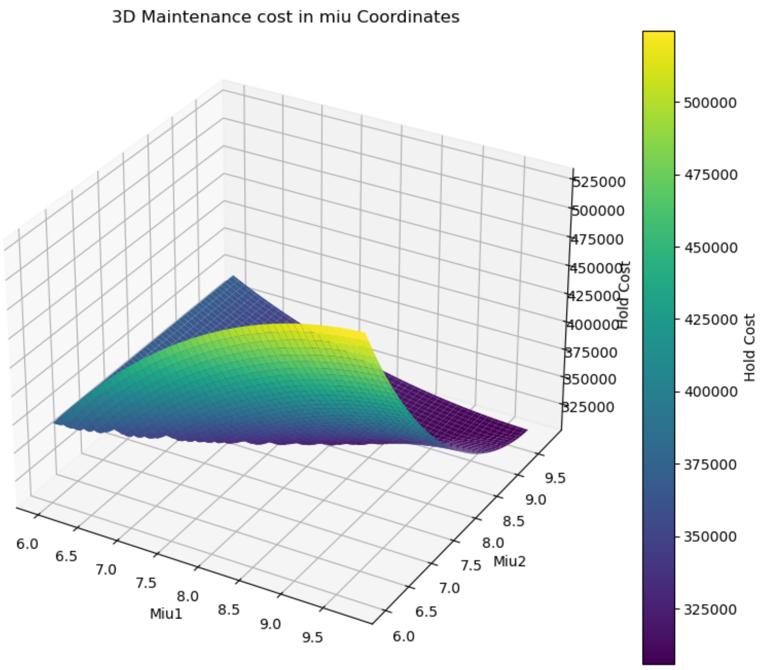


Figure 15: Three suppliers under hold constant C_h = [8, 16, 12]