SIMULATION AND OPTIMIZATION TECHNIQUES FOR THE MITIGATION OF DISRUPTIONS TO SUPPLY CHAINS

Raj Patel
Abhisekh Rana
Sean Luke
Carlotta Domeniconi
Hamdi Kavak
Jim Jones

George Mason University
4400 University Drive, Fairfax, VA, USA
{arana6, rpatel17, sean,
cdomenic, hkavak, jjonesu}@gmu.edu

Andrew Crooks

University at Buffalo
12 Capen Hall, Buffalo, NY, USA
atcrooks@buffalo.edu

ABSTRACT

The COVID-19 pandemic has clearly highlighted the importance of supply chains to the function of the world economy. Moreover, the global nature of most modern supply chains along with their complexity has left them vulnerable to a wide-ranging set of disruptive scenarios. This increase in complexity has also led to a corresponding increase in disruptions to supply chains from criminal networks. In this paper, we demonstrate how a generic pharmaceutical supply chain network can be successfully modeled using discrete event simulation. We outline how disruptions by criminal networks and mitigation strategies to counter them can be effectively incorporated into the same model. Finally, we show how optimization techniques, such as evolutionary computation, can be used to not only identify worst-case disruptions and find mitigations for them, but also be used to identify mitigation strategies that are effective against a diverse set of damaging disruption scenarios.

Keywords: Simulation, Optimization, Supply Chains, Disruptions, Mitigation.

1 INTRODUCTION AND BACKGROUND

Supply chains and their operations remain critical to the function of the world economy. The global nature of modern supply chains, and specifically pharmaceutical supply chains, has led to an increase in their susceptibility to disruptions, impacting the production of key materials such as personal protective equipment, medical supplies, and vaccines (Chowdhury et al. 2021, Rizou et al. 2020, Kleindorfer and Saad 2005). While the effects and impacts of disruptions to supply chains from natural disasters have been well studied (e.g., Macdonald and Corsi 2013, Kleindorfer and Saad 2005), supply chain disruptions due to the actions of criminal networks and agents remains an open area of research (Basu 2013). These disruptions have proliferated since the COVID-19 pandemic and now represent a real threat to the operation of not only individual supply chains but also the global economy as a whole (Shah 2004, Settanni et al. 2017, Urciuoli et al. 2013).

Traditionally, supply chain operations and logistics have been modeled using simulation techniques, including discrete event simulation (e.g., Tako and Robinson 2012, Kleijnen 2005). However, how to incorporate and study the impact of disruptions to these models, and the efficacy of possible mitigation strategies to ameliorate the damage from them, remains an open question, which this paper addresses.

Supply chain optimization is also another major area of research. For example, optimization has been used to effectively match production capacity to changes in future demand (e.g., Shah 2004, Manopiniwes and Irohara 2014, Zokaee et al. 2017). However, these methods have not been leveraged to study and identify worst-case (optimized) disruption scenarios or to generate efficacious mitigation strategies for them. This is another area of research where this paper makes a contribution. It is also important to note that identifying worst-case disruptions and developing mitigations for them may not accurately reflect real-world scenarios. This is especially the case when considering disruptions by an intelligent criminal network, which may be able to change its disruption strategies to bypass the mitigations that are in place. Thus, implementing an optimized mitigation strategy, which is not only effective against worst-case disruption scenarios but also robust against multiple different damaging ones, is of paramount importance.

The paper is organized as follows. In Section 2, we outline our methodology for modeling a supply chain via Discrete Event Simulation and how to incorporate both disruption and mitigation modeling into the same simulation. We then detail the evolutionary computation optimization techniques that we use to optimize both the disruption and mitigation scenarios. In Section 3, we report the results of our experiments and insights derived from them, and finally finish with a conclusion of our key findings and directions for future research in Section 4.

2 METHODOLOGY

2.1 Simulation of the Supply Chain Model

In this paper we use Discrete Event Simulation (DES) to model a generic pharmaceutical supply chain. DES is one of a set of widely used simulation approaches currently used to model the logistics and operations of global supply chains (e.g., Tako and Robinson 2012, Manuj et al. 2009, Settanni et al. 2017). The topology of this supply chain model was built using input from domain experts from the pharmaceutical and security industry and is shown in Figure 1. The parameters and the distributions used in the supply chain model are derived from extensive consultations with domain experts. However, these parameters and distributions can be changed by the end-user, allowing for the modeling of different supply chains and their operations.

This model includes a focal company (FC), which is the studied pharmaceutical manufacturer. In Figure 1, all icons marked in gray represent the procurement and production facilities of the FC. The other icons represent entities external to the FC, such as external suppliers, Contract Manufacturing Organizations (CMOs), Distributors, and Wholesalers. The black arrows represent information flows between facilities (nodes), while the blue and red arrows represent material flows.

At the start of the model the Hospital/Pharmacy pool receives monthly orders of drugs from the end consumer pool. As the model is stochastic, the value of these orders is determined by a triangular distribution, which is the norm for supply chain models in the pharmaceutical sector (e.g., Toba et al. 2008, Moons et al. 2019). These monthly orders are then conveyed to the Wholesaler, who relays this information to the Distributor. The Distributor then places the order for the required drugs to the FC. In a baseline simulation (i.e. a simulation without disruptions), the FC always has enough capacity to fulfill the whole monthly order from the Wholesaler. However, in the case of disruptions to the operations of the FC, the shortfall in the monthly order is met by the Wholesaler via procurement of drugs from the Untrusted Supplier Pool.

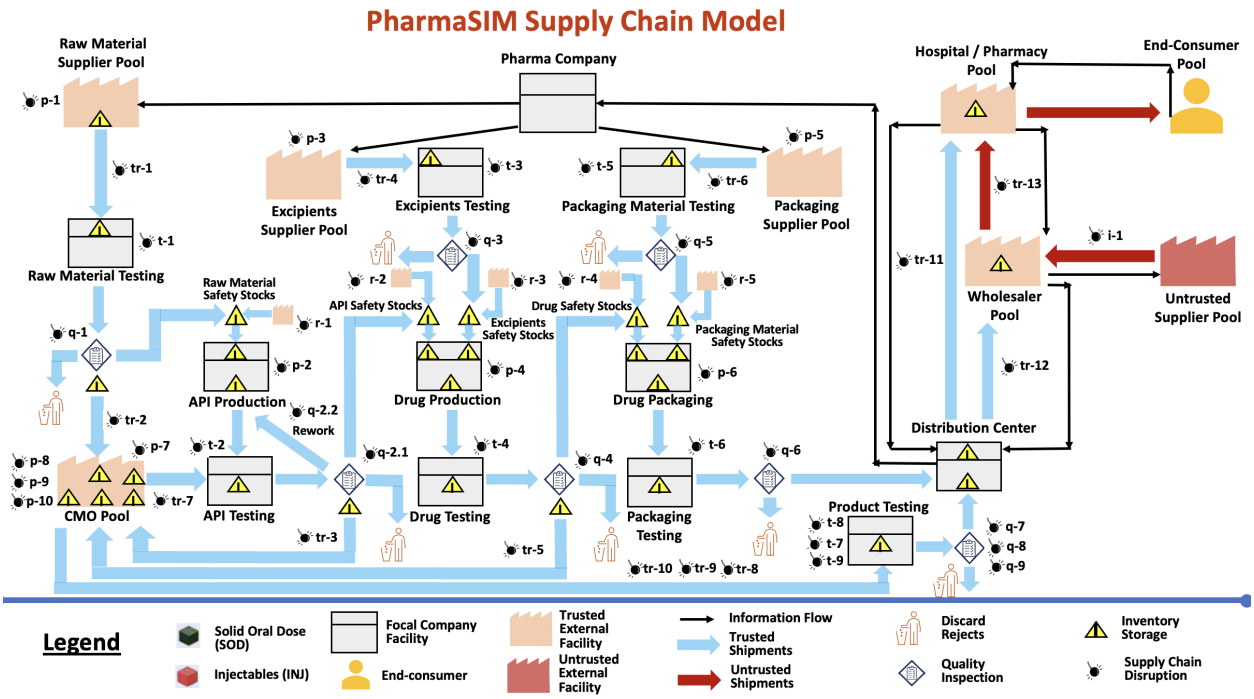

Figure 1: Topology of the generic pharmaceutical supply chain (PharmaSIM) model.

Once the monthly order is received by the FC, the FC places orders for the required materials from its external suppliers. These include supplies for the Raw Materials, Excipients and Packaging. From a DES perspective all three of these suppliers are modeled as infinite resource pools. These orders are fulfilled in batches and the transportation times between order placement and delivery are modeled using delays, which all follow distributions with different parameter settings. Once the batches are received at the corresponding inventory storage centers of the FC, they are stored in a queue. Each batch is then tested. Testing time is represented by a delay, and a batch may fail testing based on a user-defined probability distribution. The batches that pass testing are moved to queues that represent the production centers of the FC. Batches of Raw Materials are used to produce batches of the Active Pharmaceutical Ingredient (API). All production processes are also modeled using delays, after which the API is tested and then combined with the Excipients to create the Bulk Drug. The batches of the Bulk Drug are tested and then packaged, using the Packaging Materials, and tested again to complete production of batches of the finished drug. Batches of these finished drugs are then transported to the distributor, with the delivery time again being determined by a delay. The FC also has access to a CMO to which it can ship a certain percentage of materials in order to assist it in completing the production phases of the drug within the required time. The percentage of different materials the FC outsources to the CMO for production is determined by the end-user and can be different for different supply chains.

2.2 Disruption Modeling

As noted in Section 1, the global nature of modern supply chains leaves them susceptible to a variety of different types of disruptions. In this paper, we have modeled the main types, which are ubiquitous to most supply chains and have been historically carried out by criminal networks (Koh et al. 2003, Urciuoli et al. 2013, Shah 2004). These include the intentional adulteration of materials, which results in an increase in the baseline testing failure rate for batches of materials, either procured from external suppliers or produced

Table 1: Disruption Scenarios.

Scenario	Disruption Code	Disruption Description
1	A2	Theft of Ordered Raw Materials
2	A3	Raw Material Adulteration
3	A4	Raw Material Destruction
4	A5	API Production Destruction
5	A6	Halt in API Production (Cyber or Physical Attack)
6	A7	API Adulteration
7	A8	Theft of Ordered Excipients
8	A9	Excipient Adulteration
9	A10	Excipient Destruction
10	A11	Drug Production Destruction
11	A12	Halt in Drug Production (Cyber or Physical Attack)
12	A13	Drug Adulteration
13	A14	Theft of Ordered Packaging Materials
14	A15	Packaging Material Adulteration
15	A16	Packaging Material Destruction
16	A17	Packaged Drug Destruction
17	A18	Drug Packaging Halting (Cyber or Physical Attack)
18	A19	Packaging Adulteration

internally. A second disruption is the non-receipt of ordered materials, which represents theft of batches of materials while in transit. A third disruption type is the destruction of goods/products, which models a physical attack, and destroys inventory stored at various points in the supply chain network. A final disruption is a halting of, or a decrease in normal production capacity, which models damage to facilities from a physical or cyber-attack. These disruptions can also occur at different points in the chain and thus have varying impact on the supply chain's operations. Each type of disruption and its location represents a different disruption scenario. These are detailed in Table 1.

A simulation in our MASON-based application (which we detail further in Section 2.7) can be run jointly with a *disruption scenario* object, which specifies the list of disruptions that can occur during the simulation. Each disruption is characterized by its type, location, start time, and magnitude. However, since different disruption scenarios can have varying impacts on supply chains operation, and since disruptions may occur simultaneously, how to identify the most damaging or worst-case disruption scenarios remains an open question, which this paper addresses.

2.3 Disruption Detection via Online Anomaly Detection

One of the main challenges in supply chain management is the ability to accurately and rapidly identify disruptions in real-time, which would enable the deployment of effective mitigation strategies to protect the supply chain from damage (Wu et al. 2007, Tang 2006). In this paper we have implemented an online anomaly detector (OAD), which uses sensors at six strategic locations (identified by domain experts) in the supply chain to identify anomalies. These sensors measure the daily flows of materials at all production units of the FC. Though the supply chain model is stochastic in nature, in a baseline simulation (i.e., a simulation without disruptions), these daily production flows are stable, with very low standard deviation from the mean. We leverage this stable nature of the baseline simulation, and at the beginning of each day we calculate the "average normal level" of the flows as the daily flow amount averaged over the most recent 15 days that have not been labeled as "anomalous". At the end of the day, the day's flow amount is compared to the current "average normal level", and if it is less than 80% of that average, the day is labeled

as anomalous. The parameters of this OAD can of course be changed for different use cases. It should also be noted that since the OAD uses daily flow amounts, there is a natural detection delay of up to 24 hours, depending on the exact time the anomaly begins.

2.4 Mitigation Modeling

The primary tool available to supply chains in the pharmaceutical sector, for mitigation of potential disruptions, is the procurement and maintenance of adequate levels of safety stocks of key materials (Amirjabbari and Bhuiyan 2014, Graves and Willems 2000, Graves and Willems 2003). Safety stocks represent an extra inventory of materials, which are available for immediate use, in the case of disruption to normal supply. In our model, and after consultation with industry experts, there are five locations within the FC (outlined in Section 1), where different types of safety stocks are stored. For example, the API production node has a safety stock of raw materials, the drug production node has a safety stock of API and a safety stock of excipients, and so on. In our simulation, a safety stock can only be utilized once the OAD (which is described in more detail in Section 2.3), has identified an anomaly has occurred or is in progress. However, what these safety stock levels should be and how to optimize them to protect the supply chain from worst-case disruption scenarios, while simultaneously making them robust against a variety of other effective disruptions, remains an open question, which we address.

2.5 The Fitness Function

In this paper we utilize optimization to identify the most damaging and worst-case disruption scenarios, and find efficacious mitigation strategies for them (via the appropriate allocation and deployment of safety stocks). In order to accomplish this task we require a global metric that accurately measures and quantifies the overall production performance of the FC. We refer to this metric as the *fitness function*, and utilize it to measure the efficacy of different disruption scenarios and the strategies used to mitigate their impact. The fitness function utilized in this paper is the total amount of finished product supplied by the FC's production network to the distribution center during the simulation period. Every individual disruption scenario contributes to an overall decrease in this metric. Though a baseline simulation (i.e., a simulation without disruptions) is still stochastic, the overall baseline value of the total finished product delivered to the distributor is relatively stable, with a standard deviation after 500 simulations of under 1%. Thus, the magnitude of the global impact of any disruption, and the corresponding mitigation, is accurately reflected by changes in this fitness, when compared to the baseline, during a single simulation.

2.6 Evolutionary and Coevolutionary Optimization

The stochastic optimization techniques we are using, for both disruptions and mitigations, fall in the general category of *evolutionary computation* methods, where a sample of candidate solutions is iteratively tested, then re-sampled based on the performance of the various solutions, to produce a next generation of candidates, and so on (Luke 2013). The main method we use is the well-regarded *Covariance Matrix Adaptation Evolution Strategy* (or CMA-ES) (Hansen and Ostermeier 1996). This method falls in the general area of *estimation of distribution algorithm* (or EDA) methods, part of the evolutionary computation family. Simply stated: an EDA maintains a sample (or *population*) of candidate solutions (*individuals*) to a problem, and tests all of them. It then fits a distribution to this sample, weighted so that the density of the distribution corresponds to the tested quality (or *fitness*) of the individuals. It then re-samples new random individuals generated from this distribution, tests them, and repeats.

In our case, the individuals represent attacks and mitigations and are simply real-valued vectors. Both the attackers and supply chains are allocated a "budget", which determines the amount of resources available for both the attack and mitigation tasks respectively. The budget for the attacks can be allocated to the different disruption scenarios, with the result being that the higher the budget allocated to each scenario the larger the impact of the said scenario. The allocation of the budget is a real-valued attack vector of length 18 (representing the different disruption scenarios detailed in Table 1), which forms a candidate attacker solution. For mitigations, the budget determines the number of safety stocks the model has available to allocate to each of the five chosen safety stock locations in the supply chain (see Section 1). Thus, this results in a real-valued mitigation vector of length five, which forms a candidate mitigation solution.

An optimization technique like this would be very effective at identifying methods of attack against a *specific* supply chain and its mitigation configurations. Similarly, once we have optimized the method of attack against a typical supply chain, we could fix it and in turn optimize the supply chain's mitigations (via safety stocks) against that optimal attack, using the same technique. However, while this might produce a supply chain which defends itself well against a targeted method of attack (in this case optimized for worst-case scenarios), there are many different attack scenarios, which while still being very damaging, are not represented by these optimized attacks. The question then becomes how to implement a technique which is *robust* against a wide range of efficacious attacks.

One widely used method is called *competitive two-population coevolution* (Hillis 1990). Using this method we generate two separate populations: a population of attack approaches, and a population of mitigation configurations. In each iteration, each attack approach is tested against a sample of mitigation configurations from the corresponding population; and likewise each mitigation configuration is tested against a sample of attack approaches. The fitness of the various attack methods, determined by the fitness function detailed in Section 2.5, and mitigation configurations (the allocation of the safety stocks) are based on these tests, after which CMA-ES is applied to each separate population to produce the next generation's population. The idea behind competitive two-population coevolution is that it sets up an arms race: while the supply chains are searching for ways to defend themselves against damaging attacks, the criminal organizations are searching unexplored crevices in the attack space for holes the supply chains had not considered. Ideally this would ultimately produce mitigation strategies which are robust against a wide variety of *effective* attacks, though they may not necessarily be ideal against a *specific* one.

There are many ways to tune this general approach, but we mention one here: how the fitness is determined. An individual is tested against a sample of individuals from the corresponding population: we might assess an individual based on, say, his *best* (maximum) fitness in these tests, his *worst* (minimum) fitness, or his *mean* or *median* fitness. If we used maximum, we would be biasing the system to look for individuals which are as good as possible in a *specific* case: we use this to push populations of attacks to explore for undefended holes to exploit. Using the mean or median, we bias a population to search for mitigations which do well under most attack scenarios, though they may perform very poorly against certain ones. Using minimum, we bias the population to search for mitigations which are modestly effective against *all* attack scenarios, but not poor against a single one. We consider both minimum and mean fitnesses in assessing the performance of our optimized mitigation configurations.

Testing a single attack vector against a single mitigation configuration requires creating a simulation, setting it up to use the provided attack and mitigation configurations, running it, and assessing the outcome. Running a simulation takes time: but stochastic optimization may require thousands of such assessment simulations to home in on high-quality solutions. Thus, it is advantageous to distribute the optimization procedure. Evolutionary computation can do this trivially: because it is testing a population of attack vectors against a population of mitgations configurations, a master evolutionary computation process can ship off the attacks and mitigations to many remote server machines to be tested simultaneously.

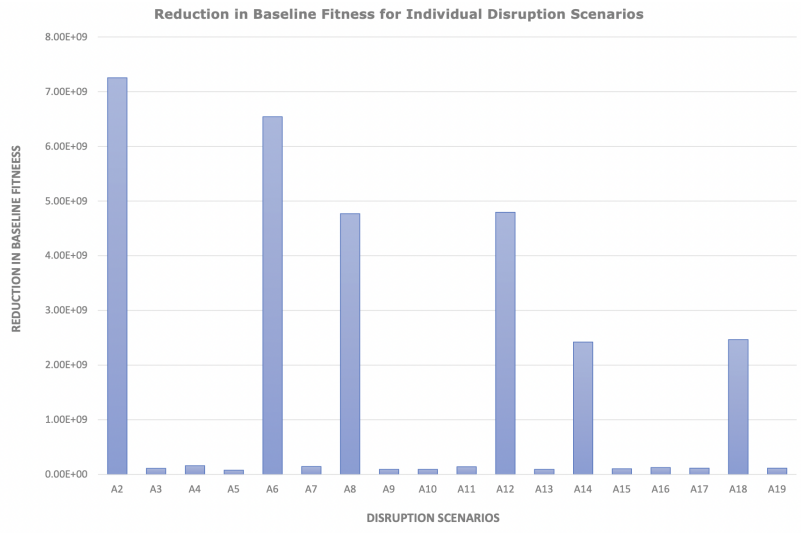

Figure 2: Reduction in fitness from the baseline (i.e. a simulation without disruptions), for each individual disruption scenario.

2.7 Simulation and Optimization Tools

To execute these tasks we use two major tools. First, we use *MASON* to model a supply chain. MASON is an open source, high-performance agent-based modeling (ABM) library written in pure Java (Luke et al. 2018). MASON is designed to be easily modified and combined with other tools. It also has extensive visualization and data reporting facilities, a model completely separable from visualization, guaranteed replicability, and an optional massively distributed version. We are using a new extension to MASON, specifically built to accommodate DES facilities. In this extension, each discrete event process object is implemented as an agent in MASON, and is connected to other process objects to form a non-blocking DES flow graph. This extension interoperates transparently with MASON's standard agent-based model approach in the same simulation.

To optimize both the attack and mitigation configurations, we use *ECJ*, an open source, high-performance evolutionary computation system, again written in pure Java(Scott and Luke 2019). ECJ is massively distributed, has guaranteed replicability, and supports a wide range of evolutionary computation and stochastic optimization methods (including both CMA-ES and two-population competitive coevolution). Importantly, MASON was designed specifically to interoperate with ECJ. MASON models are completely self-contained and can run parallel threads on the same machine. We use ECJ to optimize both the attack and mitigation configurations, and assess them on remote server machines running MASON models.

3 EXPERIMENTS

3.1 Single Disruption Scenarios

For our first set of experiments we ran individual simulations of the 18 different disruption scenarios (outlined in Section 2.2 and detailed in Table 1), in order to determine which of the disruption scenarios significantly affected the baseline operation of the supply chain. Each individual simulation is run for a period of 365 days. The disruptions were all carried out 10 days after the warm-up phase, which represents the time the model takes to reach its steady (baseline) state. This corresponds to the first 65 days of the simulation. When reporting and analyzing results, this warm-up phase is disregarded. The effect of each disruption

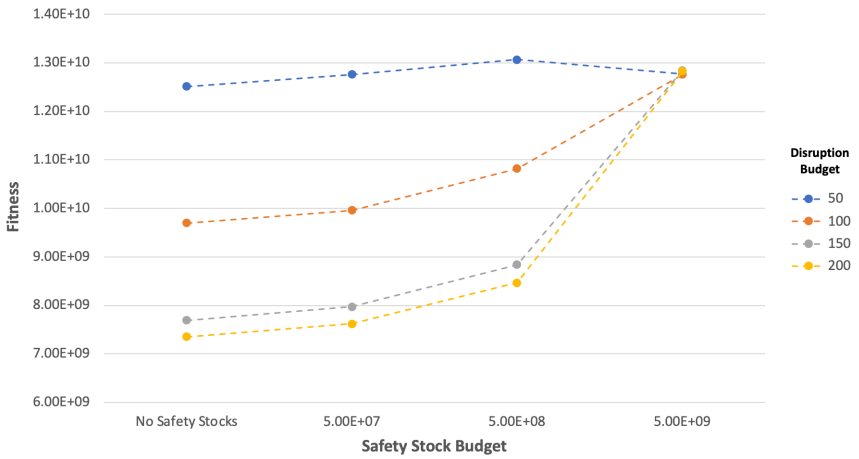

Figure 3: Fitness after evolutionary optimization of attack configurations and corresponding safety stock allocation for different budgets.

scenario, where the total budget for the disruption is fixed to 200, is reported in Figure 2. We report the reduction in fitness from a baseline simulation (where no disruptions have taken place).

The results in Figure 2 clearly demonstrate that some types of disruptions have a much larger impact than others. Specifically, theft of ordered materials (A2, A8, and A14), which are sent in shipments of multiple batches, and damage to production facilities (A6, A12, and A18), which could completely shut down the operation of the facilities, have the most significant impact. In contrast, the other disruption scenarios, which include adulteration of materials found only in single batches and physical inventory damage, have a much more limited scope and a correspondingly smaller effect. This illustrates the point that only a subset of the disruption scenarios has a significant impact on the supply chain's operation, and we leverage this result in constructing a robust evaluation set of disruptions to test the efficacy of the optimization of our mitigations in Section 3.3.

3.2 Evolutionary Optimization of Attacks and Mitigations

In our next set of experiments we take the approach, detailed in Section 2.6, of optimizing attacks to the supply chain without safety stocks, which identifies the worst-case scenarios, and then optimizing the safety stock allocation to counter the effect of these attacks. We ran experiments for different disruption and safety stock budgets, the results of which are detailed in Figure 3.

As the results in Figure 3 clearly demonstrate, a larger attack budget causes more damage to the supply chain. The budget required to efficaciously mitigate this damage also increases as the attack budget is increased. We find that for different attack budgets, ranging from 50-200, a safety stock budget of 5×10^9 is sufficient to completely mitigate the damage from these worst-case attacks to the supply chain's operations, and return the fitness to the baseline. Thus, for our next set of experiments, which compares the efficacy of the evolutionary approach analyzed in this section and the coevolutionary approach, detailed in Section 2.6, we use the maximum attack budget of 200, used in these experiments, and a safety stock budget of 5×10^9, which as noted above, is sufficient to mitigate these worst-case attack scenarios.

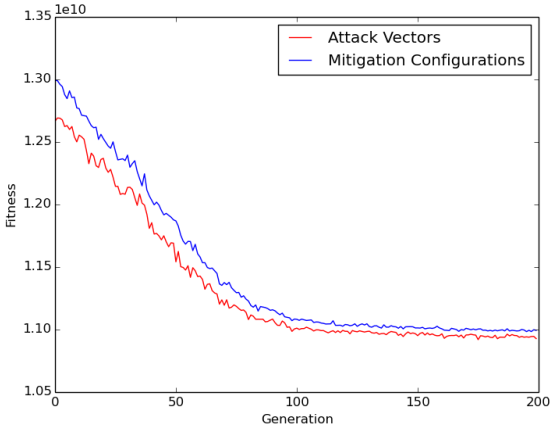

Figure 4: Fitness by generation for the coevolution of attack vectors and mitigation configurations.

3.3 Coevolutionary Optimization of Attacks and Mitigations

Our next experiment focuses on using a coevolutionary approach to optimize mitigations against different attack scenarios. In the first experiment, we use an attack budget of 200 and a safety stock budget of 5×10^9; the reasons for these settings are detailed in Section 3.2. We also carry out experiments with safety stock budgets of 1×10^9 and 1×10^{10}. By applying the coevolutionary optimization approach discussed in Section 3.3, we ran CMA-ES coevolution with a population size of 20 for both attack vectors and safety stock mitigation configurations, for 200 generations. The fitnesses from the evolved attacks and mitigation configurations during the coevolution process is shown in Figure 4. For comparison, we also use the non-coevolution approach, detailed in Section 2.6, where we separately optimize the disruptions and then optimize the mitigations in response to them. We refer to this approach as *Individual* Optimization.

Next, we test our best mitigation candidates against an evaluation set of attacks. To build this set, we created six attack vectors with the entire budget allocated in turn to one of the six effective disruption scenarios discussed in Section 3.1. We then generated 100 attack vectors by randomly splitting the budget only among the six effective disruption scenarios. Out of these 100, we selected the top 34 attack vectors which were most effective in disrupting the supply chain with no safety stock mitigation. Together with the original six, these make up an evaluation set of 40 attack vectors. We ran our simulation with each of these 40 attacks and our optimized safety stock mitigations, and calculate the overall aggregated fitness score using both average and minimum. The average describes how well the mitigations perform under many attack scenarios, while the minimum describes how well the mitigations perform in the worst case, both of which are relevant in judging how well the supply chain is protected.

The above approach was repeated 10 times and the average of the results, along with the corresponding standard deviations, for our chosen safety stock budget value of 5×10^9 are shown in Table 2. The results for the other budget values of 1×10^9 and 1×10^{10} are shown in Tables 3 and 4, respectively. For significance testing, we apply a two-sided Mann Whitney test (Mann and Whitney 1947) with $\alpha = 0.05$. Statistically significant results are given in boldface.

As these results show, the optimized safety stock mitgations are relatively effective against different attack scenarios. For our primary experiment, which uses a budget of 5×10^9, Coevolution outperforms the Individual approach, especially when looking at the worst-case scenario (Minimum). In fact, we expect Coevolution to generally be more robust than the Individual approach, as its solutions are tested against a variety of progressively more effective attacks as the evolution process converges to an optimal solution.

Table 2: Individual vs Coevolution with 5×10^9 Budget.

	Individual	Coevolution
Average	$1.182 \times 10^{10} \pm 1.744 \times 10^8$	$\mathbf{1.194 \times 10^{10}} \pm 9.571 \times 10^7$
Min	$8.862 \times 10^9 \pm 9.367 \times 10^8$	$\mathbf{9.746 \times 10^9} \pm 3.699 \times 10^8$

Table 3: Individual vs Coevolution with 1×10^9 Budget.

	Individual	Coevolution
Average	$1.125 \times 10^{10} \pm 3.217 \times 10^7$	$1.126 \times 10^{10} \pm 3.182 \times 10^7$
Min	$\mathbf{7.595 \times 10^9} \pm 2.392 \times 10^8$	$7.146 \times 10^9 \pm 1.405 \times 10^8$

Table 4: Individual vs Coevolution with 1×10^{10} Budget.

	Individual	Coevolution
Average	$1.206 \times 10^{10} \pm 7.427 \times 10^7$	$1.208 \times 10^{10} \pm 6.435 \times 10^7$
Min	$1.009 \times 10^{10} \pm 7.535 \times 10^8$	$1.025 \times 10^{10} \pm 6.261 \times 10^8$

We also find no significant difference in performance between the Individual and Coevolution approaches for a larger attack budget of 1×10^{10}; and the Individual approach is slightly better for a much smaller budget of 1×10^9. We postulate that the second result is because the budget is too small for the coevolution approach to effectively optimize itself against a variety of attacks during the training process.

4 CONCLUSION AND FUTURE WORK

In this paper we showed how a generic pharmaceutical supply chain can be effectively modeled via Discrete Event Simulation, and how how disruptions to this supply chain — specifically disruptions historically carried out by criminal networks — and mitigation strategies to counter them, can be incorporated into the same model. We next demonstrated how different evolutionary computation techniques could be used to not only identify worst-case disruption scenarios but to also optimize the allocation of the mitigations to counter their effects. We found that coevolution was more robust against a variety of different disruptions scenarios, while individual optimization was more effective against single worst-case ones.

Our results leave a lot of space for future work. The DES simulation toolkit allows for the modeling of many kinds of supply chains. Though we only incorporate and analyze disruptions traditionally carried out by criminal networks, our model also allows for other disruption scenarios, such as natural disasters and pandemics. Furthermore, the model also allows for seamless integration of other mitigation strategies such as the use of alternate suppliers, CMOs, and consumers. We also have facilities available for the utilization of other evolutionary techniques for disruption and mitigation optimization. However, we have demonstrated that the techniques used in this paper are not only effective in mitigating worst-case disruption scenarios, but are also effective against a diverse set of disruptions. This result is of significant importance, especially since supply chains and their operations have been shown to affect the stability of the world economy.

ACKNOWLEDGMENTS

This project has been funded in whole or in part with Federal funds from the Department of Homeland Security under BOA number 70RSAT18G00000001, task order number 70RSAT21FR0000127. The content of this publication does not necessarily reflect the views or policies of the Department of Homeland Security, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

REFERENCES

Amirjabbari, B., and N. Bhuiyan. 2014. "Determining supply chain safety stock level and location". *Journal of Industrial Engineering and Management (JIEM)* vol. 7 (1), pp. 42–71.

Basu, G. 2013. "The Role of Transnational Smuggling Operations in Illicit Supply Chains". *Journal of Transportation Security* vol. 6 (4), pp. 315–328.

Chowdhury, P., S. K. Paul, S. Kaisar, and M. A. Moktadir. 2021. "COVID-19 pandemic related supply chain studies: A systematic review". *Transportation Research Part E: Logistics and Transportation Review* vol. 148, pp. 102271.

Graves, S. C., and S. P. Willems. 2000. "Optimizing strategic safety stock placement in supply chains". *Manufacturing & Service Operations Management* vol. 2 (1), pp. 68–83.

Graves, S. C., and S. P. Willems. 2003. "Supply chain design: safety stock placement and supply chain configuration". *Handbooks in Operations Research and Management Science* vol. 11, pp. 95–132.

Hansen, N., and A. Ostermeier. 1996. "Adapting Arbitrary Normal Mutation Distributions in Evolution Strategies: The Covariance Matrix Adaptation". In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation (CEC)*, pp. 312–317, Institute of Electrical and Electronics Engineers.

Hillis, D. 1990. "Co-evolving Parasites Improve Simulated Evolution as an Optimization Procedure". *Physica D* vol. 42 (1–3).

Kleijnen, J. P. 2005. "Supply chain simulation tools and techniques: a survey". *International Journal of Simulation and Process Modelling* vol. 1 (1-2), pp. 82–89.

Kleindorfer, P. R., and G. H. Saad. 2005. "Managing disruption risks in supply chains". *Production and Operations Management* vol. 14 (1), pp. 53–68.

Koh, R., E. W. Schuster, I. Chackrabarti, and A. Bellman. 2003. "Securing the pharmaceutical supply chain". Technical report, Auto-ID Labs, Massachusetts Institute of Technology, Cambridge, MA.

Luke, S. 2013. *Essentials of Metaheuristics*. second ed. Lulu. Available for free at http://cs.gmu.edu/∼sean/book/metaheuristics/.

Luke, S., R. Simon, A. Crooks, H. Wang, E. Wei, D. Freelan, C. Spagnuolo, V. Scarano, G. Cordasco, and C. Cioffi-Revilla. 2018. "The MASON Simulation Toolkit: Past, Present, and Future". In *International Workshop on Multi-Agent-Based Simulation (MABS)*.

Macdonald, J. R., and T. M. Corsi. 2013. "Supply chain disruption management: Severe events, recovery, and performance". *Journal of Business Logistics* vol. 34 (4), pp. 270–288.

Mann, H. B., and D. R. Whitney. 1947. "On a test of whether one of two random variables is stochastically larger than the other". *The annals of mathematical statistics*, pp. 50–60.

Manopiniwes, W., and T. Irohara. 2014. "A review of relief supply chain optimization". *Industrial Engineering and Management Systems* vol. 13 (1), pp. 1–14.

Manuj, I., J. T. Mentzer, and M. R. Bowers. 2009. "Improving the rigor of discrete-event simulation in logistics and supply chain research". *International Journal of Physical Distribution & Logistics Management*.

Moons, K., G. Waeyenbergh, and L. Pintelon. 2019. "Measuring the logistics performance of internal hospital supply chains–a literature study". *Omega* vol. 82, pp. 205–217.

Rizou, M., I. M. Galanakis, T. M. Aldawoud, and C. M. Galanakis. 2020. "Safety of foods, food supply chain and environment within the COVID-19 pandemic". *Trends in Food Science & Technology* vol. 102, pp. 293–299.

Scott, E., and S. Luke. 2019. "ECJ at 20: Toward a General Metaheuristics Toolkit". In *GECCO '19 Companion*.

Settanni, E., T. S. Harrington, and J. S. Srai. 2017. "Pharmaceutical supply chain models: A synthesis from a systems view of operations research". *Operations Research Perspectives* vol. 4, pp. 74–95.

Shah, N. 2004. "Pharmaceutical supply chains: key issues and strategies for optimisation". *Computers & chemical engineering* vol. 28 (6-7), pp. 929–941.

Tako, A. A., and S. Robinson. 2012. "The application of discrete event simulation and system dynamics in the logistics and supply chain context". *Decision Support Systems* vol. 52 (4), pp. 802–815.

Tang, C. S. 2006. "Robust strategies for mitigating supply chain disruptions". *International Journal of Logistics: Research and Applications* vol. 9 (1), pp. 33–45.

Toba, S., M. Tomasini, and Y. H. Yang. 2008. "Supply chain management in hospital: a case study". *California Journal of Operations Management* vol. 6 (1), pp. 49–55.

Urciuoli, L., T. Männistö, J. Hintsa, and T. Khan. 2013. "Supply chain cyber security—potential threats". *Information & Security: An International Journal* vol. 29 (1).

Wu, T., J. Blackhurst, and P. O'Grady. 2007. "Methodology for Supply Chain Disruption Analysis". *International Journal of Production Research* vol. 45 (7), pp. 1665–1682.

Zokaee, S., A. Jabbarzadeh, B. Fahimnia, and S. J. Sadjadi. 2017. "Robust supply chain network design: an optimization model with real world application". *Annals of Operations Research* vol. 257 (1), pp. 15–44.

AUTHOR BIOGRAPHIES

RAJ PATEL is a PhD student in the Department of Computer Science (CS) at George Mason University (GMU). His research interests include data mining and natural language processing. Raj's email address is rpatel17@gmu.edu.

ABHISEKH RANA is a PhD student in the Department of CS at GMU. His research areas include natural language processing and computational finance. Abhisekh's email address is arana6@gmu.edu.

SEAN LUKE is a Professor in the Department of CS at GMU. He has interests in stochastic optimization, multiagent systems and multiagent learning, agent-based modeling, and swarm robotics. His email address is sean@gmu.edu.

CARLOTTA DOMENICONI is an Associate Professor in the Department of CS at GMU. She has interests in clustering, anomaly detection, text mining, and network analysis. Her email address is cdomenic@gmu.edu.

ANDREW CROOKS is a Professor in the Department of Geography at the University at Buffalo. His research interests include geographical information science and computational social science. His email address is atcrooks@buffalo.edu

HAMDI KAVAK is an Assistant Professor in the Computational and Data Sciences Department and co-director of the Center for Social Complexity at GMU. His research interests lie at the intersection of data science and modeling & simulation. His email address is hkavak@gmu.edu.

JIM JONES is an Associate Professor in the Electrical and Computer Engineering Department at GMU and the Director of the DHS Center of Excellence for Criminal Investigations and Network Analysis. His research interests include cyber-security, digital forensics, criminal operations, and network discovery from data. Jim's email address is jjonesu@gmu.edu.