**Noname manuscript No.**
(will be inserted by the editor)

# Bicriteria approaches for an optimal balance between resilience and cost-effectiveness of supply chains

**Erik Diessel · Heiner Ackermann**

the date of receipt and acceptance should be inserted later

**Abstract** In supply chain optimization multiple objectives are considered simultaneously, for example to increase resilience and reduce costs. In this paper we discuss the corresponding bicriteria problems to find a good balance between these two objectives. We give a general model for supply chain resilience that integrates strategic decisions with the operational level. This modular model allows the flexible combination of various constraints, objectives and different types of uncertainties. We treat uncertainties with a scenario-based approach.

The resulting models have continuous and discrete decision variables. The computation of the Pareto frontier for these large-scale bicriteria mixed-integer problems creates a computational challenge. We illustrate and use the novel Adaptive Patch Approximation algorithm to efficiently compute approximations of the Pareto frontier.

After presenting the theoretical advantages of the algorithm we compute the Pareto frontier of several supply chain problems. Our analysis of the different shapes of Pareto frontiers illustrates the characteristics of different types of bicriteria problems as well as the relevant impact on decision makers.

**Keywords** Bicriteria optimization · mixed-integer programming · supply chains · resilience

## Declarations

Funding: Fraunhofer ITWM, Institute for Industrial Mathematics
Conflicts of interest/Competing interests: None to declare
Availability of data and material: None available
Code availability: None available

## 1 Challenges of optimizing supply chains

Due to changes in the economic environment there is a constant need for adapting and optimizing the supply chain to new circumstances. A reconsideration of existing supply chains can for example be triggered by the ending of contracts, uncovered problems relating to quality and security in the supply chain or changes in demand. Supply chain optimization based on mathematical models is essential to improve the performance of the supply chain in such situations while fulfilling the constraints imposed by the previous configuration, guidelines or other limits, The most prominent goals are the improvement of the resilience of the supply chain and the reduction of costs.

The classical objective of minimizing the cost of operating a supply chain can be modeled as a network flow problem. Apart from procurement costs that correspond to continuous variables, in many cases also the organizational efforts have to be considered. For example activating a supplier might incur a fixed cost required for its setup. In the mathematical model,

Erik Diessel (Corresponding author)
Fraunhofer ITWM, Institute for Industrial Mathematics, Fraunhofer Platz 1, 67663 Kaiserslautern, Germany
ORCiD: 0000-0002-0671-9169 E-mail: erik.diessel@itwm.fraunhofer.de

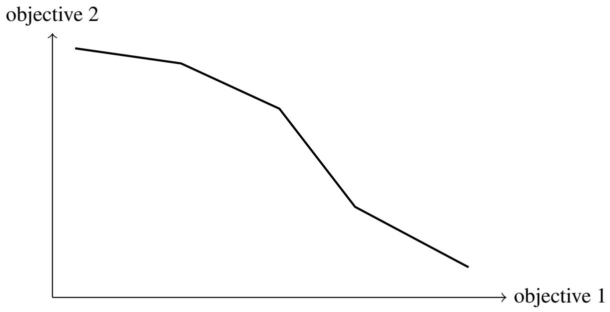
Heiner Ackermann
Fraunhofer ITWM, Institute for Industrial Mathematics, Fraunhofer Platz 1, 67663 Kaiserslautern, Germany
ORCiD: 0000-0003-4091-5558 E-mail: heiner.ackermann@itwm.fraunhofer.de

the decision to activate a supplier is represented with a discrete variable. In total, we are thus dealing with a mixed-integer problem (MIP).

Improving the resilience of a supply chain, i.e. the ability of a supply chain to adapt to major changes, is a second important objective. We model resilience by considering a set of possible scenarios in which a major deviation from the expected situation occurs, for example the long-term disruption of a supplier. For each such scenario we allow a reaction in form of a mitigation flow in the supply chain that decreases the impact of the deviation. The resulting impact in each scenario can be aggregated to a single quantity that measures the resilience of the supply chain: its ability to react to changes. We integrate two possible choices for this modeling of risks. Following the robust approach, we consider the worst-case impact among all scenarios. In the stochastic approach an average of the scenarios weighted according to the estimates probabilities is taken. In this way we can optimize the decision on a strategic level such that appropriate reactions on the operational level are enabled in case of a significant deviation.

As discussed above there are several, often conflicting objectives like costs and resilience in supply chain optimization. Thus it is important to identify all possible trade-offs. Since there exist often different views on the relative importance of these different objectives due for example to differing risk-taking attitudes, the whole range of possible trade-offs should be available to a decision maker. This gives rise to a bicriteria optimization problem that aims to compute a Pareto frontier of reachable optimal trade-offs, as shown in Figure 1. While eliminating solutions that are clearly worse with respect to both objectives, the Pareto frontier does not assume any priorities between the objectives. This provides an unbiased basis for decisions by multiple stakeholders in a supply chain design decision whose views on the relative importance of the various objectives are not aligned.

**Figure 1** Example of a non-convex Pareto frontier of a bicriteria problem.




Since realistic supply chain models correspond to large-scale mixed-integer problems, the computation of the corresponding Pareto frontiers poses a significant challenge as they are non-convex and contain potentially infinitely many Pareto-efficient points. Particularly in the context of supply chains where the number of discrete variables can get significantly large, methods based on enumeration of all possible assignments or branch-and-bound can have impractical running times. In this paper we thus use the novel Adaptive Patch Approximation algorithm for the approximation of Pareto frontiers of such mixed-integer problems (Diessel 2020a). The algorithm is based on iteratively combining adaptively chosen patches to create a patchwork that approximates the Pareto frontier with increasing accuracy. The algorithm obtains a provably almost-optimal convergence rate. Due to the inherent noise in supply chain data, the provided approximations with small error are sufficient whereas an exact computation of the Pareto frontier by existing algorithms becomes impractical.

For modeling a wide range of supply chain optimization problems we provide a framework for optimizing the resilience of supply chains by combining the strategic level with the operational level. The flexible nature of the framework allows to incorporate a wide range of different objectives and constraints on the strategic and operational level. This allows us to present a large list of such model components that can be combined to models for a large range of supply chain design tasks. Using several of such models we examine the different shapes of the corresponding Pareto frontiers, their relation to the types of used constraints and their impact for decision makers.

## 1.1 Related work

### *1.1.1 Supply chain models*

For supply chain optimization, a wide range of different models focusing on various aspects are available. For sustainable supply chain design Eskandarpour et al. (2015) give a survey on optimization approaches that also include multicriteria models.

Since supply chains are inevitably affected by future events that are not yet fully known during design, uncertainties are often connected to the optimization of supply chains. Govindan, Fattahi, and Keyvanshokooh (2017) give a survey on supply chain optimization models to cope with these uncertainties. As a specific type of uncertainty, optimal reactions to disruptions in the supply chain in form of a mitigation are also an important topic in supply chain optimization (Ali and Nakade 2015; Ivanov et al. 2017; Rajagopal, Prasanna Venkatesan, and Goh 2017). Typically, individual models are however focused on a particular part of the supply chain and type of uncertainty. For example, Cui, Ouyang, and Shen (2010) consider a facility location problem faced with supplier disruptions. For managing such risks a large range of different types of actions is possible (Tang 2006). The analysis of supply chain risks is often done through simulation which prevents the systematic consideration of these risks during optimization (Klibi and Martel 2012).

Due to the large variety of supply chain risks (Heckmann, Comes, and Nickel 2015) for which a large set of specialized models (Fahimnia et al. 2015) has been developed, flexible models that can quickly be adapted to different situations are better suited for practical usage. As the full elimination of risks is often not feasible, the resilience of a supply chain, i.e. the possibility of a supply chain to react to a disrupting scenario with a sufficient reaction in form of a mitigation, is a frequent goal for decision makers (Tomlin 2006; Kamalahmadi and Parast 2016). This paper provides a flexible model of resilience that can incorporate different types of objectives, constraints, sources of uncertainty and types of mitigation. By sharing a common mathematical framework this enables supply chain optimization in a large range of different industries and situations.

An analysis of the trade-offs between costs and risks is often done on a qualitative level while quantitative studies are less common and are instead focused on a particular type of model (Huang and Goetschalckx 2014; Mori et al. 2017). In this paper we aim to give a more general illustration of effects on the trade-offs by various types of constraints. We achieve this by comparing the Pareto frontiers between various types of models in a shared framework.

### *1.1.2 Bicriteria optimization*

A bicriteria optimization problem has the form

$$\begin{aligned} \min\ & (g_1(x), g_2(x)) \\ \text{s.t. } & x \in X \end{aligned}$$

where $X$ is the feasible set in objective space, and $g_1$ and $g_2$ are objective functions. Typically the set $X$ is represented through inequalities. An objective vector $(g_1(x), g_2(x))$ is called nondominated in the case of minimization if there does *not* exist another solution $y \in X$ which is better with respect to both objectives, by fulfilling the inequalities $g_1(y) \leq g_1(x)$ and $g_2(y) \leq g_2(x)$ while $(g_1(x), g_2(x)) \neq (g_1(y), g_2(y))$.

The Pareto frontier consists of all nondominated solutions. The task of non-interactive multicriteria optimization is the computation or approximation of Pareto frontiers. For a general introduction to the theory of multicriteria optimization we refer to Ehrgott (2005).

For problems with only continuous variables and linear constraints and objectives, the approximation of the Pareto frontier can be done with the Sandwiching algorithm which obtains an optimal convergence rate (Rote 1992). With additional discrete variables, the approximation of the Pareto frontier becomes significantly more challenging. In this paper we consider bicriteria convex mixed-integer problems which have the following structure

$$\begin{aligned} & \min\ (g_1(c,d), g_2(c,d)) \\ \text{s.t.}\quad & h_i(c,d) \leq 0 \qquad \text{for all } i \in I \\ & c \in \mathbb{R}^k \\ & d \in \mathbb{Z}^m \end{aligned}$$

where the objective functions $g_1$ and $g_2$ are linear and the functions $h_i$, $i \in I$ describing the constraints are convex or linear. Due to the presence of the integral variables $d$ the resulting Pareto frontier is non-convex. The continuous part of variables $c$ can cause the existence of an infinitely large set of nondominated solutions. Thus, for solving multicriteria mixed-integer programs algorithms designed for pure integer or continuous problems cannot be used. Instead, specific algorithms have to be created. For bicriteria mixed-integer programs two major classes of algorithms are available:

– Branch-and-bound algorithms that perform recursive branching on some variables (Mavrotas and Diakoulaki 1998; Stidsen, Andersen, and Dammann 2014; Belotti, Soylu, and Wiecek 2016; Cacchiani and D'Ambrosio 2017).
– Scalarization algorithms that combine objective functions to create single-objective problems that can be solved to obtain more information on the Pareto frontier (Vassilvitskii and Yannakakis 2005; Kim and Weck 2006; Boland, Charkhgard, and Savelsbergh 2014; Fattahi and Turkay 2018; Perini et al. 2019).

We use in this paper the new Adaptive Patch Approximation algorithm (Diessel 2020a) that differs in many aspects from the previously used algorithms since it focuses directly on obtaining approximations instead of the prevalent approach of the exact computation of the whole Pareto frontier. The algorithm adaptively computes optimal patches to improve the approximation to the Pareto frontier in each iteration as much as possible. This enables theoretically-guaranteed almost-optimal convergence rates.

### 1.2 Our results and organization of this paper

Our general framework for modeling the optimization of resilience in supply chains is given in section 2. The components for concrete models with a selection of possible constraints and objectives are developed in section 3. In section 4 we describe the novel Adaptive Patch Approximation algorithm for bicriteria mixed-integer programming and the underlying principles. We illustrate the algorithm and its advantages along an example. We use this algorithm to compute in the subsequent section 5 the Pareto frontiers of several supply chain models with different choices for constraints and objectives. We discuss the corresponding Pareto frontiers and their relation to the model type. We conclude with an outlook in section 6.

## 2 Modeling resilience in supply chains

As decisions have to be taken often ahead of time with limited information, supply chains are faced with many risks. These risks itself cannot be prevented by companies as recent examples of epidemics and other *force majeure* scenarios have shown. The supply chain however can be designed to increase its resilience: the ability of a supply chain to react to significant disruptions (Kamalahmadi and Parast 2016). We provide a flexible mathematical model for supply chain resilience. By taking a bicriteria approach we can ensure that this resilience is also achieved in a cost-effective way. The model is able to work with different types of uncertainty like the following:

– force-majeure events that lead to the disruption of some suppliers or supply lanes,
– future actions by competitors, regulating agencies or clients,
– severe misprediction of demand,
– prizes of goods in the future.

To ensure that a supply chain is resilient its design needs to enable a good use of mitigation in disruption scenarios (Tang 2006). We provide a general model that unifies all these types of uncertainties and corresponding mitigation into a shared framework, using a three-stage model. This flexible framework enables to quickly adapt the model to new types of uncertainties. By sharing a common language for often separately treated uncertainties, also new synergies and insights can be found. The presented stochastic or robust modeling allows for the quantification of the resilience which enables a rigorous optimization.

Dealing with uncertainties also typically involves a trade-off between different objectives. For example the trade-off between the usual costs of operating a supply chain and the risks which it faces. Although the ideal goal for each of the decisions (e.g. lowering both costs and risks) is shared by all stakeholders, the importance assigned to these different objectives is often varying.

To give an accurate view on the possible trade-offs we compute Pareto frontiers. This allows to eliminate solutions that are worse with respect to every objective while not making any assumptions on the relative importance of objectives. The Pareto frontier provides all reasonable options for the design of the supply chain while not ruling out any meaningful possibilities due to fixed priorities.

## 2.1 Scenario-based modeling of uncertainties

To model uncertainties, we make use of scenarios. Each scenario consists of a set of significant changes to parts of the supply chain. For example, a scenario can consist of a force majeure event leading to the shutdown of a certain supplier. Similarly, a significantly increased interest in a market for a product, increasing the demand, can be modeled as a scenario.

By creating a set of possible scenarios that could impact the decision and the objectives, the decision maker models the uncertainties. The set of scenarios should be large enough to cover all events that have a sufficient likelihood to occur, while ruling out extremely unlikely events. By different choices for the set of scenarios, uncertainties on several scales can be modeled. For operational planning, small scenario sets might be sufficient as the range of possible events can be restricted. For strategic planning however, larger set of scenarios considering more extreme events can ensure that also large-scale uncertainties are handled appropriately.

*Scenario based flow modeling* A supply chain design can typically be described by flows of goods or assets. We distinguish between two types of flows: The nominal flow corresponds to the original situation not affected by any unplanned influence. For each scenario, a mitigation flow can be chosen, that represents the reaction taken to the scenario, reducing its impact. For example, when a supplier is disrupted additional material might be sent from alternative suppliers.

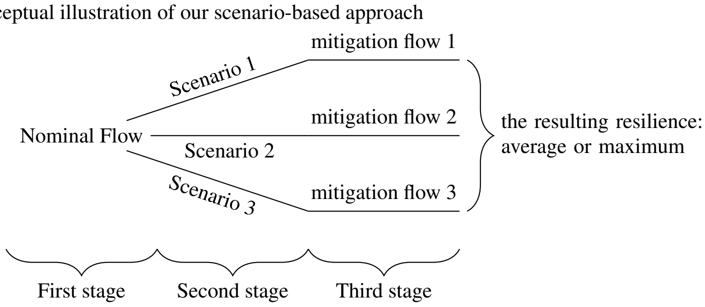
## 2.2 A three-stage model for resilience

Our general model consists of three stages. In the first stage a nominal flow is chosen. Then in the second stage some scenario arises. The exact scenario was not known before, but it is known that the scenario is a member of a known set of possible scenarios. Depending on the scenario, a reaction can be taken in form of the mitigation flow in the third stage. Each of these mitigation flows can be chosen independently of each other to optimize the objective.

Constraints can link the mitigation flow to the nominal flow. Thus, the choice of the nominal flow influences the options available for the mitigation flow. Hence, the nominal flow should be chosen in such a way that preferable options are available for the mitigation flow in most scenarios.

The objective function value obtained in a scenario depends on the nominal flow, the scenario itself and the mitigation flow. As this creates a separate objective value for each scenario, we use a way to scalarize this into a single value of resilience that quantifies the impact of the uncertainty taking into account the mitigation. In Figure 2 our scenario-based three-stage framework is illustrated.

**Figure 2** Conceptual illustration of our scenario-based approach



*Quantifying the effect of the uncertainties* To combine the information on the outcome in several scenarios into one measure of resilience, we allow for two approaches.

In the robust approach, we consider the worst case of the several scenario objective values $g(f,x,f^{(x)})$. The aim is then to improve this worst case, i.e. making sure that none of the scenarios $x$ has an extremely bad impact on the objective $g$ after taking into account the nominal flow $f$ and the mitigation flow $f^{(x)}$. Mathematically this corresponds to taking a maximum-operator. The optimization task can thus be summarized as the following three-stage expression

$$\min_{f} \max_{x \in X} \min_{f^{(x)}} g(f,x,f^{(x)}).$$

The corresponding approach can be seen as a special case of adjustable robust optimization (Ben-Tal et al. 2004). A detailed discussion of adjustable robust optimization models in supply chains is provided by Diessel (2019).

Alternatively, in stochastic optimization models we measure the resilience as an average of the objective values over the scenarios. The scenarios can be weighted according to the estimated probability $w(x)$ of each scenario $x$. In this way we model the expectation value of the objective value obtained for a random scenario. In a mathematical model this can be easily represented by a weighted sum, leading to an optimization problem of the form

$$\min_{f} \sum_{x \in X} \min_{f^{(x)}} w(x) \cdot g(f,x,f^{(x)})$$

This type of model corresponds to stochastic programming which aims to optimize the expected value of an optimization problem with random coefficients (Birge and Louveaux 2011).

## 3 A flexible supply chain modeling framework

### 3.1 Classification scheme

We use a field-notation similar to the notation for scheduling problems by Graham et al. (1979) to describe biobjective supply chain problems following a scenario-based approach. This allows a short and unambiguous notation for a wide range of supply chain problems.

The notation consists of

- constraints for the nominal flow,
- constraints for the mitigation flow,
- terms for the first objective function,
- terms for the second objective function.

A bicriteria supply chain problem is then specified in the form

nominal constraints | mitigation constraints | objective function 1 | objective function 2.

The constraints for the nominal flow are independent of scenario and mitigation flow. The constraints for the mitigation flow can integrate the corresponding scenario and mitigation flow as well as the chosen nominal flow. The objective functions can depend on all of these values however separately for each scenario and corresponding mitigation flow. The resulting objective values in each of the separate scenarios need to be combined into a single number. As discussed in subsection 2.1, there are two possibilities for this. A robust modeling corresponding to the objective $g$ is written as $\max g$, while a stochastic modeling is denoted as $\sum g$.

### 3.2 General model

Our supply chain models are based on network flows with additional constraints. In our base model, we assume that there is only one commodity and that the nominal flow is steady over time. It is possible to extend the model to include also a nominal flow varying over time or the use of multiple commodities.

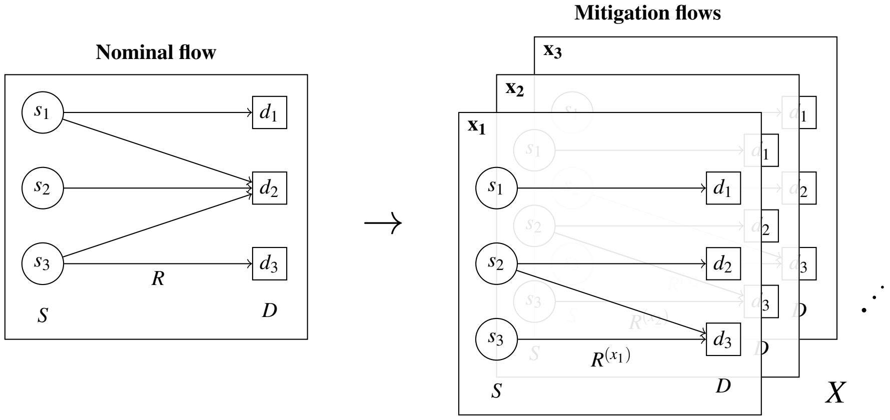
Our model is based on the following components:

- a set of suppliers $S$,
- a set of demand nodes $D$,

- a set of nominal arcs $R$,
- a set of scenarios $X$,
- a set of mitigation arcs $R^{(x)}$ for each scenario $x \in X$.

This basic model is illustrated in Figure 3. We denote for some node $v$ (which can be either a supplier or a demand node) the set of outgoing nominal arcs by $\delta^+(v) := \{(v,u) \mid (v,u) \in R\}$ and the set of ingoing arcs as $\delta^-(v) := \{(u,v) \mid (u,v) \in R\}$. The set of outgoing mitigation arcs for scenario $x$ is denoted by $\delta_x^+(v)$, the set of ingoing arcs by $\delta_x^-(v)$.

**Figure 3** Our scenario-based model for a supply chain: a nominal flow determines the constraints for each of the mitigation flows corresponding to a set of scenarios $X = \{x_1, x_2, x_3, \dots\}$.




For each nominal arc $r \in R$ there exists a variable $f_r \in \mathbb{R}_{\geq 0}$ for the nominal flow on this arc. For each mitigation arc $r \in R^{(x)}$ there is for every scenario $x \in X$ a variable $f_r^{(x)} \in \mathbb{R}_{\geq 0}$ for the mitigation flow over this arc in the scenario $x$.

## 3.3 Creating a corresponding MIP-formulation

The flexible nature of our framework allows the direct translation of a supply chain problem given in the field notation

nominal constraints | mitigation constraints | objective function 1 | objective function 2

into a corresponding bicriteria MIP that can be directly solved. Each of the constraints corresponds to a set of linear inequalities, with possible additional variables which are listed in Sections 3.4, 3.5 and 3.6. These inequalities and variables can then be combined in the desired way to yield the full formulation of the bicriteria program corresponding to the supply chain model. For a scenario set $X$ we obtain the following formulation by combining all parts:

$$
\begin{aligned}
&\min\ (g_1, g_2) \\
\text{s.t.}\quad & \text{linear inequalities corresponding to nominal constraints} \\
& \text{linear inequalities corresponding to mitigation constraints for each } x \in X, r \in R^{(x)} \\
& \text{additional variables required for the nominimal constraints} \\
& \text{additional variables required for the mitigation constraints for each } x \in X \\
& \text{additional variables required for the two objective functions} \\
& f_r \in \mathbb{R}_{\geq 0} \text{ for each } r \in R \\
& f_r^{(x)} \in \mathbb{R}_{\geq 0} \text{ for each } x \in X, r \in R^{(x)}
\end{aligned}
\tag{1}
$$

In the case that the objective function $g_i$ has the robust form $\max g$, the objective is represented as a maximum over the scenarios, e.g. by the linear inequalities

$$g_i \geq g^{(x)} \text{ for each } x \in X$$

where $g^{(x)}$ is the objective value in scenario $x \in X$.

If $g_i$ is a stochastic objective $\sum g$, we can represent it as

$$g_i = \sum_{x \in X} w^{(x)} g^{(x)}$$

where $w^{(x)}$ is the weight given to the scenario $x$, or 1 if it is not specified.

## 3.4 List of nominal constraints

We now describe a number of constraints regarding the nominal flow. For each of the constraints we describe the required attributes (for example capacity values), a mathematical formulation of the constraint as well as a corresponding linear MIP-formulation with additional variables when the nonlinearity of the constraint requires this. For each constraint we give an abbreviation that we will use in our field notation. By combining the MIP-formulations for each of the constraint types corresponding to a supply chain problem, a corresponding MIP can be built. Due to space limitations we only list a selection of these constraints to show the range of possibilities, however this list can be easily extended.

### *3.4.1 Demand fullfillment* (DF)

**Description:** The constraint ensures that each demand node receives exactly the required amount.
**Required attribute:** Nominal demand $b_d \in \mathbb{R}_{\geq 0}$ for each demand node $d \in D$
**Mathematical formulation:**

$$\sum_{r \in \delta^-(d)} f_r = b_d \quad \text{for each } d \in D.$$

### *3.4.2 Nominal arc capacity* (NAC)

**Description:** This constraint limits the amount of flow over individual arcs.
**Required attribute:** Upper bound $u_r \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ for each nominal arc $r \in \mathbb{R}$
**Mathematical formulation:**

$$f_r \leq u_r \text{ for each } r \in R$$

### *3.4.3 Supplier capacity* (SC)

**Description:** The constraint limits the total flow from each supplier. This corresponds to the limited production capacities of suppliers.
**Required attribute:** Limit $C_s \in \mathbb{R}_{\geq 0}$ on the capacity for each supplier $s \in S$
**Mathematical formulation:**

$$\sum_{r \in \delta^+(s)} f_r \leq C_s \quad \text{for each } s \in S$$

### *3.4.4 Number of active suppliers* (NAS)

**Description:** The constraint ensures that only a limited number of suppliers are active sources. This constraint is motivated by the desire to reduce the organizational overhead. For this purpose it is often important to restrict the number of suppliers that have to be operationally managed in a supply chain.
**Required attribute:** Limit $K \in \mathbb{N}$ on the number of active suppliers
**Mathematical formulation:**

$$\left| \{ s \in S : \sum_{r \in \delta^+(s)} f_r > 0 \} \right| \leq K$$

**MIP formulation:** Introduce for each supplier $s \in S$ a binary variable $a_s \in \{0,1\}$ as an indicator for the activation of supplier $s$. We add the constraints

$$\sum_{r \in \delta^+(s)} f_r \leq M_s a_s \qquad \text{for each } s \in S,$$

$$\sum_{s\in S} a_s \leq K,$$

where $M_s$ is a constant that should be chosen as an upper bound on the sum $\sum_{r\in\delta^+(s)} f_r$ flowing out from $s$ in a feasible solution. If a demand fulfillment (DF) constraint is present we can choose $M_s = \sum_{d\in D} f_d$. If a supplier capacity (SC) constraint is present, it can be chosen as $M_s = C_s$.

## 3.5 List of mitigation constraints

In a similar format as the nominal constraints, we list possible mitigation constraints that limit the mitigation flow, depending on the nominal flow and the scenario. Such a constraint can only include mitigation flow variables from its corresponding scenario. The mitigation flows are thus independent of each other.

### *3.5.1 Remaining arc capacity* (RAC)

**Description:** This constraint limits the mitigation flow to the remaining arc capacity given by the nominal flow.
**Required attribute:** Upper bound $u_r \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ for each nominal arc $r \in R$
**Mathematical formulation:** For each scenario $x \in X$ and mitigation arc $r \in R^{(x)}$ (assuming this arc $r$ is also present in the sets of nominal arcs) it should hold

$$f_r + f_r^{(x)} \leq u_r.$$

### *3.5.2 Remaining supplier capacity* (RSC)

**Description:** This constraint ensures that the mitigation flow is using only the remaining capacity of the suppliers. If a supplier is disrupted, this also limits the mitigation flow.
**Required attributes:** Nominal capacity $C_s$ for each supplier $s \in S$; Scenario disruption value $D'^{(x)}_s \in [0,1]$ for each supplier $s \in S$ and scenario $x \in X$ denoting the fraction of which the capacity of the supplier is reduced in the scenario
**Mathematical formulation:**

$$\sum_{r\in\delta_x^+(s)} f_r^{(x)} \leq \left(1 - D'^{(x)}_s\right)\left(C_s - \sum_{r\in\delta^+(s)} f_r\right) \quad \text{for each } s \in S, x \in X$$

### *3.5.3 Mitigation Arc Capacity* (MAC)

**Description:** This constraint limits the amount of mitigation flow over each individual mitigation arc.
**Required attribute:** Capacity $u_r^{(x)} \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ for mitigation arc $r \in R^{(x)}$
**Mathematical formulation:**

$$f_r^{(x)} \leq u_r^{(x)} \quad \text{for each } x \in X,\ r \in R^{(x)}$$

### *3.5.4 Further constraints*

A large set of possible further constraints which we do not describe in detail here can be modeled with a MIP:

– mitigation from depots that store limited amounts of material,
– delays incurred by ramping up production,
– transportation delays that limit the amount of mitigation,
– cardinality constraints on the number of active suppliers or actively used arcs in mitigation.

### 3.6 Objective functions

The objective functions can depend on both the nominal flow and the mitigation flow. For each of the scenarios this results in a value of the impact after taking into account the mitigation. The values are then combined into one objective value, taking a sum or maximum over all scenario values as discussed in subsection 3.3. We assume that the optimization direction is to minimize the objective function. For quantities that should be maximized the negative of the value can therefore be taken.

#### *3.6.1 Flow costs* (FC)

**Description:** This objective measures the cost for the nominal and mitigation flow, proportional to the flow units sent. In this cost purchasing prices, transportation costs and other costs that are proportional to the amount of flow can be aggregated.

**Required attributes:** Cost $c_r \in \mathbb{R}_{\geq 0}$ for each nominal arc $r \in R$ and cost $c_r^{(x)}$ for each mitigation arc $r \in R^{(x)} \in \mathbb{R}_{\geq 0}$ in scenario $x \in X$

**Objective term:**

$$\sum_{r\in R} c_r f_r + \sum_{r\in R^{(x)}} c_r^{(x)} f_r^{(x)}$$

#### *3.6.2 Unmitigated disruption amount* (UDA)

**Description:** Measures the amount of flow that is disrupted but cannot be mitigated by the mitigation flow at the demand site. It corresponds to the bottom-line impact of the disruption.

**Required attribute:** Disruption value $D_r^{(x)} \in [0,1]$ for each nominal arc $r$ denoting which fraction of the nominal flow on this arc is disrupted in scenario $x \in X$, weight $w_d \geq 0$ for the loss at demand node $d \in D$

**Objective term:**

$$\sum_{d\in D} w_d \max\left\{ \sum_{r\in\delta^-(d)} D_r^{(x)} f_r - \sum_{r\in\delta^-(d)} f_r^{(x)}, 0 \right\}$$

**MIP formulation:** For each demand node $d \in D$ introduce a variable $u_d \in \mathbb{R}_{\geq 0}$ measuring the unmitigated flow into $d$. Add the constraints

$$u_d \geq \sum_{r\in\delta^-(d)} D_r^{(x)} f_r - \sum_{r\in\delta^-(d)} f_r^{(x)} \qquad \text{for each } d \in D.$$

The objective function is then given as

$$\sum_{d\in D} w_d u_d.$$

#### *3.6.3 Supplier activation costs* (SAC)

**Description:** When a supplier is actively used, the operational management of the relationship with the supplier requires some fixed cost. This objective function measures fixed costs which occur once a supplier is actively used.

**Required attribute:** Fixed costs $F_s \in \mathbb{R}_{\geq 0}$ for each supplier $s \in S$

**Objective term:**

$$\sum_{s\in S} F_s \cdot \mathbf{1}\Big( \sum_{r\in\delta^+(s)} f_r > 0 \Big)$$

**MIP formulation:** Introduce for each supplier $s \in S$ a binary variable $a_s \in \{0,1\}$ as an indicator for the activation of $s$. Add the constraints

$$\sum_{r\in\delta^+(s)} f_r \leq M_s a_s \qquad \text{for each } s \in S$$

where $M$ is a constant that can be chosen as in the NAS constraint (subsubsection 3.4.4). Then the objective function is given as

$$\sum_{s\in S} F_s a_s.$$

### 3.7 Further objective functions

Several other objective functions can be modeled by using MIP-constraints:

- costs for the mitigation,
- quantity discounts or in general nonlinear pricing schemes which can be modeled with binary variables that measure whether the thresholds have been met,
- non-financial objectives like environmental objectives regarding sourcing, production and transportation.

A detailed overview on this kind of extensions is given by Ackermann et al. (2020).

## 4 The Adaptive Patch Approximation Algorithm

As shown by our supply chain models, integral variables are required to model many types of constraints and objectives, for example cardinality constraints. At the same time a large number of continuous variables represent the flows in the supply chain. Hence, realistic supply chain optimization deals with mixed-integer problems that have both a significant continuous and discrete part. Computing the Pareto frontier of corresponding bicriteria problems is thus a significant challenge due to the mixed-integer structure. Available bicriteria algorithms which are focused on the exact computation of the full Pareto frontier would require a too long running time for these problems. However, due to the inherent noise in the supply chain data a good approximation of the Pareto frontier is sufficient.

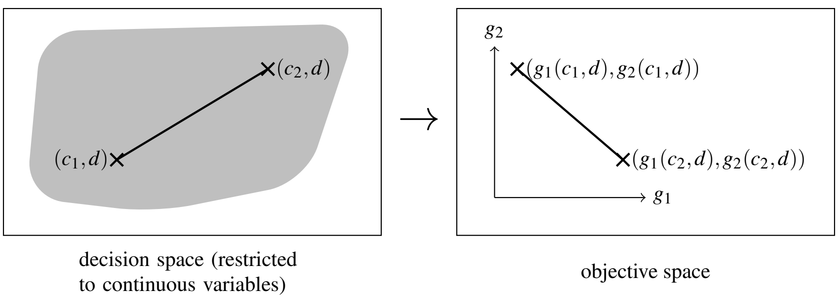
Thus, we use the novel Adaptive Patch Approximation Algorithm that is focused on the computation of increasingly better approximations to the Pareto frontier with a fast convergence. This algorithm iteratively creates patchworks that provide increasingly good approximations to the Pareto frontier. The algorithm is described in detail along with proofs on convergence guarantees in Diessel (2020a). We describe here the main ideas of the algorithm along an example and state theoretical results on the convergence.

### 4.1 The basis: Patches

Algorithms for continuous convex bicriteria problems like the Sandwiching algorithm (Rote 1992) are based on the concept of interpolation: From two solutions convex combinations can be created. Due to the convexity of the constraints all these convex combinations are also feasible. In objective space, this corresponds to a segment of feasible objective vectors.

For mixed integer problems this is not possible for every choice of two solutions since we cannot interpolate over the discrete variables. However, if we restrict us to pairs of solutions that share the same values for the discrete variables, interpolation becomes also possible for MIPs. Our algorithm is based on finding such *patches*: sets of solutions with shared values for the discrete variables. Two solutions with shared values of the discrete values already give rise to a *segment patch* by considering their convex combinations as illustrated in Figure 4. We base our algorithm on searching these segment patches since they can be found by a simple MIP.

**Figure 4** A segment patch is created by a convex combinations of two solutions $(c_1,d)$ and $(c_2,d)$ with shared values $d$ for the discrete variables. This is possible since the feasible set in decision space is convex when restricted to the continuous variables.



The advantage of using patches in comparison to using single points is that Pareto frontiers often have slopes which can be better approximated with line segments. The number of needed patches is significantly smaller than the number of points needed for a similar accuracy.

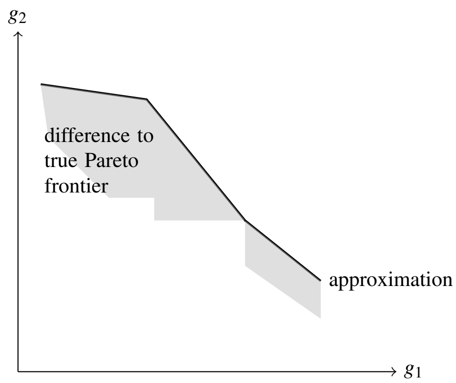
Many algorithms for bicriteria MIPs (e.g.: Mavrotas and Diakoulaki (1998)) first compute individual nondominated solutions. The discrete variable values of such a solution are then fixed and the Pareto frontier of the corresponding continuous bicriteria problem is computed. Since the choice of the original nondominated solution did not take into account the shape the Pareto frontier of the corresponding continuous problem, a large number of steps might be needed. The Adaptive Patch Approximation Algorithm instead systematically searches directly for good patches. By searching for a patch that optimally improves the quality of the current approximation it can achieve a fast convergence.

## 4.2 The algorithmic steps along an example

We describe the algorithm by illustrating its steps on an example.

*Basic setup* Assume that we are given a bicriteria MIP, represented by two linear objective functions and convex inequalities with continuous and discrete variables. The corresponding Pareto frontier is non-convex but consists of a finite number of linear segments. Our goal is to find an approximation of this Pareto frontier by combining several solutions. In this way we get an approximate Pareto frontier that has only small differences to the true Pareto frontier, as shown in Figure 5.

**Figure 5** Illustration of an approximation (black line) of the true Pareto frontier, the gray area represents the corresponding approximation error.



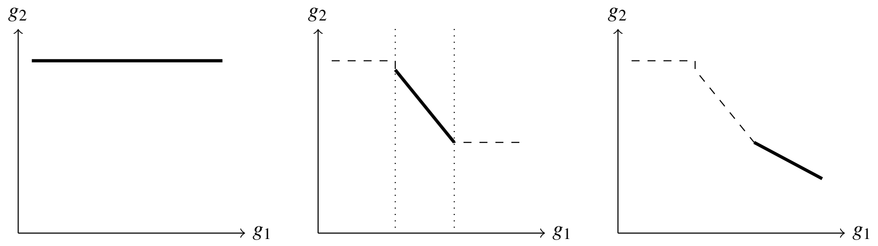

The algorithm constructs an approximation consisting of as few segments as possible while keeping the approximation error low. As a measure of the approximation error, we use the *difference volume* between the approximation and the true Pareto frontier, i.e. the the gray area in Figure 5. To ensure that we are not dependent on the ranges of the objective functions, we scale the objectives such that the Nadir point has coordinates $(1,1)$ and the ideal point coordinates $(0,0)$.

*Initialization* Our algorithm proceeds iteratively. It starts with a crude approximation which is improved in each iteration by adding an additional patch. In the beginning we compute the lexicographic optima. By taking the solution which achieves minimal value for the first objective, we obtain a valid solution for each bound on the first objective. Thus we get an approximation consisting of a single horizontal segment extending from this lexicographic optimum (Figure 6a).

*Iterative approach* We now search for a patch that optimally improves the approximation, i.e. decreases the difference volume between true Pareto frontier and approximation as much as possible. It can be shown that we can get within an constant factor of the maximal possible improvement by maximizing the integral improvement $I_A(s)$ obtained by the new segment patch $s$ and the approximation $A$, given by

$$I_A(s) := \int_{X_1(s)}^{X_2(s)} A_y(x) - s_y(x)\,dx$$

where $X_1(s)$ and $X_2(s)$ are the $x$-coordinate of the left respectively right endpoint of the segment patch $s$ and $A_y(x)$ and $s_y(x)$ denote the $y$-coordinate corresponding to the $x$-coordinate on the corresponding curve. By maximizing the improvement $I_A(s)$ we obtain a patch that can be inserted into the approximation, resulting in a better approximation as shown in Figure 6b.




(a) Initial approximation (b) After inserting the second patch (c) After inserting a third patch

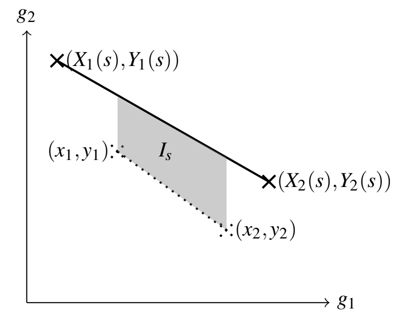
**Figure 6** The first few iterations of the algorithm.

We now continue in this way. The approximation always consists of a set of segments. To simplify the search, we compute a patch inside each region given by the segments of the approximation. The region corresponding to a segment is an interval on the axis of the first objective created by projecting the segment to this axis. Boundaries of these regions are indicated in Figure 6b as dotted vertical lines. Since the approximation is linear in each of these regions, the expression for the improvement $I_A(s)$ becomes bilinear in the coordinates of the two endpoints of the segment patch. This enables to formulate the maximization of $I_A(s)$ as a efficient optimization problem, as discussed below.

In each iteration, we thus compute in each region a segment patch maximizing the improvement $I_A(s)$. We then insert the patch which achieved the largest improvement value into the approximation. This results then in a better approximation as shown in Figure 6c. By repeating these steps yielding increasingly better approximations, the difference volume quickly converges to 0.

*Computing a segment patch* We now discuss how a segment patch can be computed. Since a segment patch is given by its two endpoints $(x_1, y_1)$ and $(x_2, y_2)$, it is sufficient to search for two solutions with shared values of the discrete variables. We maximize the integral improvement $I_s$ with respect to a fixed reference segment $s$ of the current approximation that has endpoints $(X_1(s), Y_1(s))$ and $(X_2(s), Y_2(s))$. Figure 7 provides a geometric illustration of the optimized quantity.

**Figure 7** Illustration of the integral improvement $I_s$ of a segment patch with respect to the fixed reference segment $s$.



This leads to solving a MIP of the form given in (3).

$$\max\ (x_2 - x_1) \cdot \left( Y_1(s) + (Y_2(s) - Y_1(s)) \frac{x_1/2 + x_2/2 - X_1(s)}{X_2(s) - X_1(s)} - \frac{y_1 + y_2}{2} \right) \tag{2}$$

$$\begin{aligned} \text{s.t.} \quad (c_1, d) &\in X \\ (c_2, d) &\in X \\ x_1 &\geq g_1(c_1, d) \\ x_2 &\geq g_1(c_2, d) \\ y_1 &= g_2(c_1, d) \\ y_2 &= g_2(c_2, d) \\ x_1 &\geq X_1(s) \\ x_2 &\leq X_2(s) \\ x_1 &\leq x_2 \end{aligned} \tag{3}$$

The set of feasible solutions $X \subseteq \mathbb{R}^k \times \mathbb{Z}^m$ can be represented by using the constraints and variables from the original mixed-integer problem.

Our original objective is to maximize the integral improvement $I_s$ which we formulate as follows. We denote by $\Delta x$ the $x$-length of the segment patch and by $\Delta y$ the average difference in the $y$-direction of the segment patch and the reference segment of the approximation. The integral improvement $I_s(s')$ can then be represented as

$$I_s(s') = \Delta x \cdot \Delta y$$

since the segment of the approximation is linear. These lengths $\Delta x$ and $\Delta y$ both depend linearly on the coordinates $(x_1, y_1)$ and $(x_2, y_2)$ of the endpoints of the segment patch. They can thus be represented with linear terms using the objective function values, yielding the objective term found in (2).

The bilinear term $\Delta x \cdot \Delta y$ can be replaced by maximizing the logarithm

$$\ln(I_s(s')) = \ln(\Delta x \cdot \Delta y) = \ln(\Delta x) + \ln(\Delta y)$$

instead. Since the logarithm is a monotone function this yields an equivalent problem. Because the logarithm is also concave, we can approximate it from above using linear inequalities. For this, only a relatively small number of linear inequalities is needed to get an approximation with a constant factor close to 1 which is sufficient for our purposes. This yields a linear formulation for the task of computing a segment patch, having the same structure as the original MIP only with the continuous variables and constraints doubled.

### 4.3 Convergence results

The analysis of our algorithm is based on the segment patch complexity, which characterizes the intrinsic difficulty of approximating a particular Pareto frontier.

**Definition 1 (Segment patch complexity)** The segment patch complexity $N_{\delta_{vol}}(\varepsilon)$ corresponding to a bicriteria mixed-integer problem is the smallest number of segment patches needed to create an approximation that achieves a difference volume of at most $\varepsilon$.

Since the segment patch complexity describes the minimal size any output with the desired accuracy must have, it provides a natural lower-bound for number of optimization steps that need to be done by an algorithm.

Our algorithm is able to almost match that lower bound in terms of the number of MIPs it needs to solve. In particular, it can be shown (Diessel 2020a) that the number of MIPs solved by the algorithm to reach a difference volume less than $\varepsilon$ is bounded by

$$O\left(N_{\delta_{vol}}(\varepsilon/2)\alpha\left(N_{\delta_{vol}}(\varepsilon/2)\right)\ln(1/\varepsilon)\right)$$

where $\alpha$ is the extremely slow-growing inverse Ackermann function. Thus, apart from the logarithmic factor the number of MIP solves is essentially linear in the segment patch complexity. This shows that the algorithm achieves an almost optimal convergence speed.

### 4.4 Advantages of the algorithm

In summary, the algorithm provides the following advantages that are not possible with previously available algorithms

- The algorithm has a guaranteed almost-optimal convergence rate based on the intrinsic difficulty of approximating a Pareto frontier.
- The algorithm can be stopped at any time while always providing a good guarantee on the approximation quality. It is an *anytime* algorithm.
- The representation of the approximation of the Pareto frontier by a minimal number of patches simplifies the subsequent usage of this approximation. The smaller number of parts in the Pareto frontier makes it easier to represent and understand the results. Alternative choices for the discrete variables are simpler to evaluate by the decision maker since only a small number of discrete solutions are required. Also the task of navigating on the Pareto frontier is simpler in comparison to an approximation provided by a large number of individual points.

## 5 Trade-offs in supply chain models

### 5.1 Generation of example data

For generating realistic supply chain data used as attributes in our constraints and objectives we use the open-sourced tool *scgen* (Diessel 2020b). The tool allows to randomly generate values for the various attributes present in the constraints discussed above. Dependencies between different attributes are treated automatically to ensure feasibility, e.g. capacity values are adjusted to the demand values.

For each attribute a random distribution can be chosen. The tool also allows the modeling of correlations. Data correlations are important since structural properties of real-world supply chains lead to dependencies with significant effects on the optimization. For example, a supplier might be located in a remote area. Thus, the transportation costs from this supplier to demand sites are generally larger. Such effects can be modeled by introducing a positive correlation between the transportation costs over arcs coming from the same supplier. The tool provides a simple specification of such correlations by introducing factors for each part that are then shared between different attributes. For example in the case of flow costs between suppliers and demand sites, each cost $c_r$ over an arc $r$ is randomly generated. However, the cost of the arc $r$ from supplier $s$ to demand site $d$ has multiple components:

- a component due to specific properties of the supplier $s$ (i.e. purchasing prices or location),
- a component due to properties of the demand site $d$ (for example quality requirements),
- a component that is specific to this pair of supplier and demand site, for example the geographical distance.

The tool allows to choose a distribution for each of those components. Then, values are generated for the components. The final value is then obtained by computing the product of these random values. Note that the costs over arcs which have the same originating supplier share the component value of this suppliers and are thereby correlated.

### 5.2 Experimental studies

In strategic decisions like the design of a supply chain, there are often multiple stakeholders. As the Pareto frontier is independent of subjective views about the relative importance of the conflicting objectives, it provides an objective account of the different options. This allows to separate discussions on the objective implications of different solutions from the subjective ranking of these alternatives.

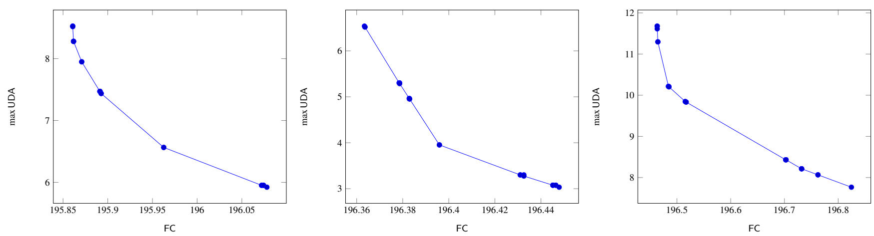
We now describe some example problems in our supply chain framework to discuss the main effects occurring in the corresponding Pareto frontiers. The configurations for the generation of our used supply chain data is available in the *scgen*-repository (Diessel 2020b) in the `bicriteria_sample_problems` folder. For creating the MIPs corresponding to the problems given in our field notation we combine the necessary inequalities and variables for the constraints and objectives occurring in the fields according to (1). We then compute the corresponding Pareto frontiers with the Adaptive Patch Approximation algorithm to a high accuracy. To illustrate the possible variations between Pareto frontiers of different instances of a problem type, we generate multiple instances from the same random distribution and display the results side-by-side.

### *5.2.1* DF, SC, NAC | RSC, RAC | FC | max UDA

This problem represents the case of trade-off between the resilience (represented by UDA: the risk after taking mitigation measures into account) and cost (represented by FC). By taking the max-operator on UDA, we consider the worst-case risk. The constraints SC and NAC limit the nominal capacity on a supplier- respectively arc-level. The mitigation flow is restricted by the constraints RSC and RAC to the remaining capacity. This problem has been considered in detail without the second objective function FC by Ackermann, Diessel, and Krumke 2020 using specialized cut-generation algorithms.

The Pareto frontiers corresponding to some instances of this problem with 10 suppliers and 10 demand nodes are shown in Figure 8. They illustrate how resilience of the supply chain can be increased (for example by diversifying the supplier base) without increasing the costs too much. A decision maker obtains the information that resilience can be significantly improved with some increase in costs. However, completely eliminating the risks is still not possible.

**Figure 8** Pareto frontier of an instance of DF, SC, NAC | RSC, RAC | FC | max UDA




(a) first instance (b) second instance (c) third instance

*Properties of convex Pareto frontiers* An effect often observed in practice is that supply chain resilience can be significantly improved with just a small increase in cost. Only a complete prevention of all risks is costly. The analysis of the convex Pareto frontier of the example instance in Figure 8 confirms this intuition. It shows that large risks (corresponding to a large value of the unmitigated disrupted amount UDA) are incurred by a solution with minimal costs. With a small increase of the cost we can however improve the resilience (i.e. decreasing the value of UDA) already by a significant amount. However improving the resilience to its optimal value is relatively costly, as this is often only possible with a very diverse supply base which requires to choose also more costly procuring options.

Note that the shape of the Pareto frontier as a convex curve with increasing slope is an immediate consequence of the fact that we do not have any integral variables in this model and all constraints are linear. Thus, also with changed instance data the Pareto frontier has similar properties. In particular due to convexity a *Pareto knee* often occurs, i.e. there is a point on the Pareto frontier with good values for both objectives where deviating from does not give a significant improvement in any objective.

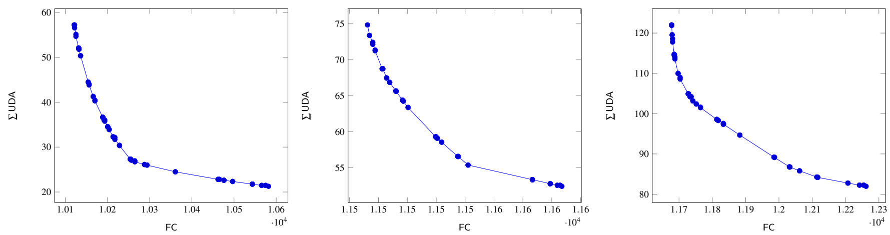
The convexity immediately implies a *diminishing returns* effect: For obtaining further improvements on an already good objective value with respect to one objective function, an increasingly larger price has to be paid for the other objective function. As this happens for both objective functions, a good trade-off is typically found with medium values on both objective function. The exact preferred situation however depends on the shape of the Pareto frontier. The knowledge on the full Pareto frontier allows to make an informed decision on this choice.

### *5.2.2* DF, SC | RSC, MAC | FC | $\sum$ UDA

In this problem type, we combine the objective $\sum$ UDA that measures the average risk due to unmitigated shortages with the total costs FC for the flow.

The constraints RSC and MAC limit the amount of mitigation. Thus, the shortage risk can only be reduced when the supply base is appropriately diversified. This limits the amount of flow that is disrupted in a scenario. The small disrupted amounts can than be recovered by the mitigation flow.

**Figure 9** Example convex Pareto frontiers for DF, SC | RSC, MAC | FC | $\sum$ UDA



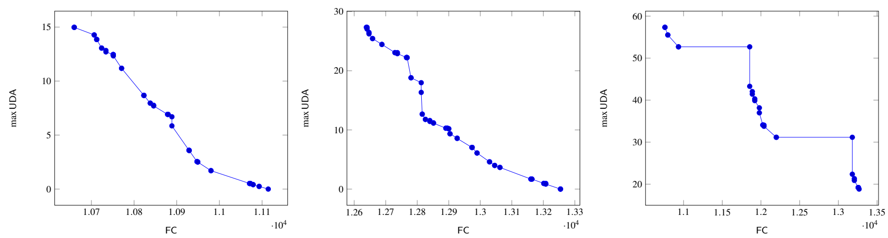

(a) first instance (b) second instance (c) third instance

The Pareto frontiers for instances of this problem type with 10 suppliers and 10 demand nodes are illustrated in Figure 9. The convex structure can be clearly seen. Although the Pareto frontier is actually piecewise linear (as each constraint is linear), it looks almost like a differentiable curve, since the number of linear segments is relatively large. Due to the stochastic type of the objective $\sum$ UDA this effect is more pronounced than in Figure 8 where an objective of robust type is used.

### *5.2.3* DF, SC, NAS | RSC, MAC | FC | max UDA

Figure 10 shows Pareto frontier of instances of DF, SC, NAS | RSC, MAC | FC | max UDA with 8 suppliers and 10 demand nodes. Since the NAS constraint introduces some binary variables into the model, we obtain a non-convex Pareto frontier. The Pareto frontier is still piecewise linear and consists of convex parts, however the Pareto frontier overall is non-convex. Thus, there is also no clear diminishing-returns effect. Instead, a reduction of the risk by the same amount can have different trade-offs depending on the location on the Pareto frontier.

**Figure 10** Example Pareto frontiers for DF, SC, NAS | RSC, MAC | FC | max UDA




(a) first instance (b) second instance (c) third instance

*Properties of non-convex Pareto frontiers* The presence of integral variables significantly changes the structure of the Pareto frontier. In particular it is not convex anymore. Due to the integrality constraints for some variables, at some positions a continuous trade-off can now longer be made. For example to reduce the risks a little bit further some additional supplier might need to be activated. But since the number of active suppliers is limited, some other supplier must be deactivated. This causes a large change on the flow which can change the risk value and the flow costs significantly. Thus, jumps occur in the Pareto frontier, where a small further decrease in risks causes the costs to go up significantly. This discrete nature complicates decision-making processes since small changes on limits for one objective function can have large consequences on the others. A fine-tuning to meet a particular target is then often not possible.

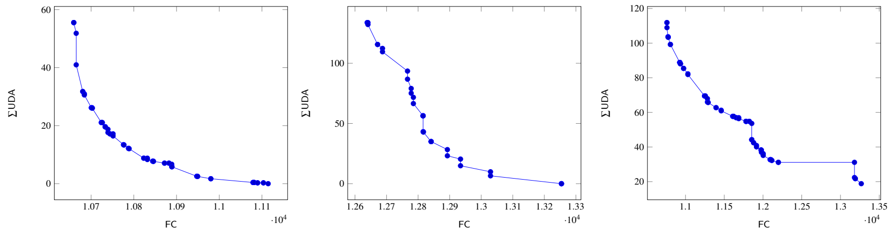
The often complex structure of the Pareto frontier highlights the need to approximate it completely in order to identify opportunities that might be overlooked by traditional management practices that focus on incremental change.

### *5.2.4* DF, SC, NAS | RSC, MAC | FC | $\sum$UDA

This model is a modified version of the previous supply chain problem. The change of the objective from max UDA to $\sum$UDA has a significant impact on the Pareto frontier. It corresponds to the change from a robust viewpoint, where the worst-case is considered, to a stochastic viewpoint where the average is considered.

In the following Figure 11, the Pareto frontiers for instances of the supply chain problem DF, SC, NAS | RSC, MAC | FC | $\sum$UDA are shown that use the exactly same data as the previous instances with max UDA-objective shown in Figure 10.

**Figure 11** Pareto frontier for instance of DF, SC, NAS | RSC, MAC | FC | $\sum$UDA




(a) first instance (b) second instance (c) third instance

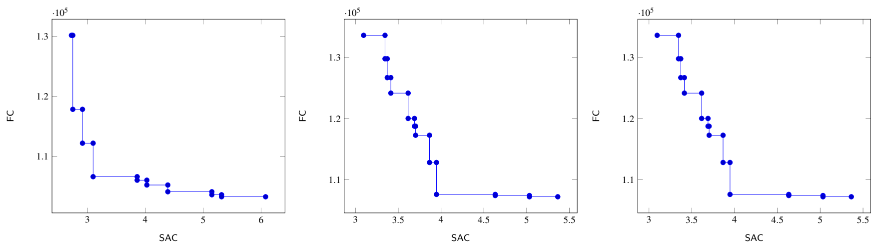
The Pareto frontier is still non-convex. However the non-convexity is much less pronounced than in the Pareto frontier of Figure 10 using the max-objective, even though the same data is used. This is caused by the change from a robust max- to a stochastic sum-objective: In the stochastic objective, several scenarios are considered simultaneously so that the rate of change of the resilience with respect to a unit of flow is smaller. With a robust objective, already a small change to some flows can have a large impact, if it affects the worst-case scenario.

### *5.2.5* DF, SC || SAC | FC

As an atypical example we consider in this supply chain problem a trade-off between two types of costs. The fixed costs for activating some suppliers, corresponding e.g. to the organizational effort to establish a new supply chain, are represented by the SAC objective. The other objective FC is the total flow cost which represents the ongoing costs for procuring materials, based on purchasing amounts. The scenarios do not play any role.

In the corresponding Pareto frontiers shown in Figure 12 we thus have a trade-off between the one-time costs given by SAC and the recurring costs given by FC. A good choice for this trade-off depends on the time span for which the supply chain will be operating as well as the balance the decision maker aims to find between short-term and long-terms costs. The information provided by the Pareto frontier enables a decision maker to choose a solution that aligns their preferences with the objectively possible options given by the model.

**Figure 12** Pareto frontiers for instances of DF, SC || SAC | FC




(a) first instance (b) second instance (c) third instance

The Pareto frontier to this problem type consists of just horizontal and vertical parts. This is due to the fact that there is a finite number of nondominated points because the SAC objective only depends on discrete variables: a supplier is either activated or not. Thus, no interpolation is possible on this SAC objective axis. Although on the other FC-axis, continuous interpolation would be possible, these solutions will not be nondominated since there is no continuous trade-off possible with the SAC objective.

Faced with a Pareto frontier of this type, a decision maker needs to choose between a discrete set of nondominated alternatives. Thus, small adjustments to a solutions are sometimes not possible without incurring big changes in a different objective or loosing Pareto-efficiency.

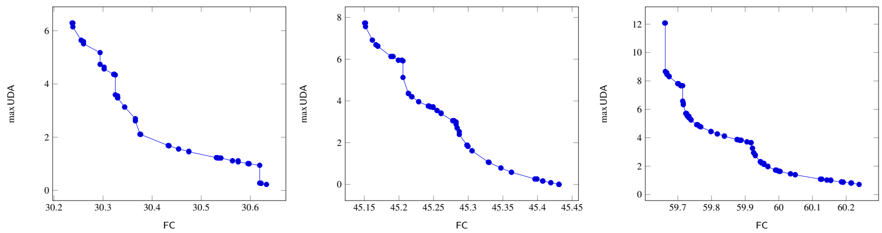
## 5.3 Running time benchmarks

To measure the efficiency of the Adaptive Patch Approximation algorithm, we compute Pareto frontiers for increasingly large supply chains. This allows us to evaluate the dependence of the running time on the size of the supply chain and the wanted accuracy. A comparison to alternative algorithms for bicriteria MIPs has shown that the running time of the Adaptive Patch Approximation is competitive on synthetic benchmarks, in particular the number of required MIP optimizations grows significantly slower than that of other algorithm with increasing problem size (Diessel 2020a).

*Computation configuration* We used the original implementation of the Adaptive Patch Approximation algorithm (Diessel 2020a) which we extended with a supply chain modeling interface. For solving the mixed-integer programs formulated by the algorithm, we use Gurobi version 8.1 (Gurobi Optimization, LLC 2018) with default settings. The benchmarks were run on a personal laptop with an Intel i7-8665U CPU with 1.9 GHz and 16 GB RAM.

*Benchmark problems* For our measurements, we use instances of the supply chain problem $\mathsf{DF}, \mathsf{SC} \mid \mathsf{RSC}, \mathsf{MAC} \mid \sum \mathsf{UDA} \mid \mathsf{FC}$. We generate these instances randomly with varying size of the supply chain according to a size parameter $n$. A supply chain of size $n$ consists of $n$ suppliers and $2n$ demand nodes. For each supply chain we create 10 disruption scenarios, where every supplier has an independent probability for failure of 20%. We generate the capacities such that a meaningful trade-off occurs in the Pareto frontier as illustrated in Figure 13.

**Figure 13** Pareto frontiers for supply chains of varying size $n$.



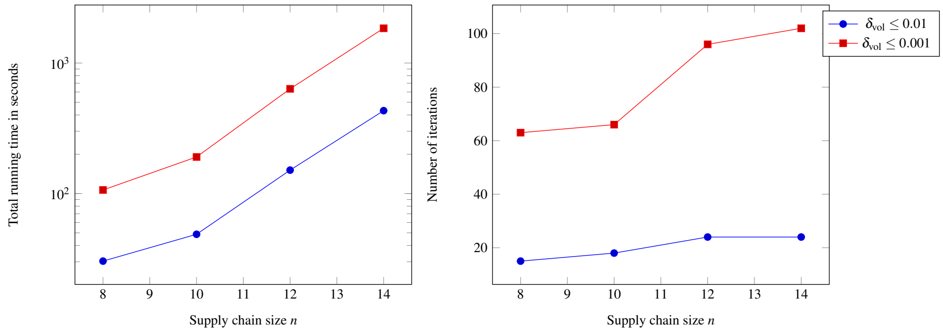

(a) $n = 8$ (b) $n = 12$ (c) $n = 16$

In Figure 13 the average running times for computing the Pareto frontier within an accuracy of a given difference volume $\delta_{\text{vol}}$ of 10 random supply chains each with varying sizes are shown. A difference volume of $\delta_{\text{vol}} \leq 0.01$ corresponds to an error of the approximation to the true Pareto frontier of about 1% of the trade-off range on average. The number of iterations required by the Adaptive Patch Approximation algorithm is small, as Figure 13 shows. In fact, the number of required iterations does not increase anymore after reaching a size of about 12. The increase of the running times with larger supply chains is mainly due to the increasing difficulty of solving the single MIPs. Note that with a supply chain size of $n = 14$ the number of continuous variables for the mitigation flows alone is

$$10|S| \cdot |D| = 10n \cdot 2n = 20n^2 = 20 \cdot 14^2 = 3920$$

which leads to a solving time of a few seconds per MIP.

The number of integral variables corresponding to the supplier activation constraint (SAC) is $n$. Since we allow the activation of up to $\frac{n}{2}$ suppliers there exist at least $2^{n-1}$ many feasible assignments. This shows that branch-and-bound methods are not feasible for such bicriteria supply chain problems since the number of feasible discrete assignments and thus number of leaf nodes exceeds $2^{13} = 8192$ with a supply chain of size $n = 14$. For each of these nodes, branch-and-bound methods would require the solution of bicriteria continuous problems, creating a computational effort infeasible for practice. Only by an adaptive choice of the best patches as done by our algorithm, the number of required optimization runs can be kept low.




(a) Running times (b) Number of iterations

## 6 Conclusion

This paper illustrates the potential of using advanced multicriteria and modeling techniques to solve a primary challenge for optimizing supply chains: identifying optimal trade-offs between conflicting goals like the increase of resilience and the reduction of costs.

We provide a general framework for modeling resilience in supply chains. By adopting scenario-based models, both robust and stochastic approaches are integrated. Together with a diverse set of possible choices for constraints and objectives, this enables a systematic treatment of different types of supply chain models. Using our methodology, corresponding formulations of bicriteria mixed-integer programs can be created.

Using a novel algorithm for bicriteria optimization of convex mixed-integer problems, the Adaptive Patch Approximation algorithm, we efficiently obtain good approximations to the Pareto frontier despite the computational hardness of the underlying problems.

The non-convex structure of the Pareto frontiers for mixed-integer problems creates special challenges for decision makers. By comparing multiple different models, we illustrate the impact of constraints and objectives on the corresponding Pareto frontier. Our general consideration of Pareto frontiers in supply chains provides practitioners with insights on the typical structure of bicriteria problems faced by them.

### 6.1 Outlook

A bicriteria approach to supply chain optimization allows us to identify opportunities for increasing the overall resilience and sustainability while still keeping other key performance indicators like costs in balance. The numerical studies presented in this paper show that bicriteria optimization can be handled efficiently by using novel algorithms even in the case of complex mixed-integer supply chain problems with large scenario sets. To apply complex models in practice, however, concrete data on the supply chain networks, possible choices, as well as predictions on future values along with appropriate estimations of the corresponding uncertainty are needed. Case studies of the Pareto frontiers associated to supply chains with real-world data would provide great opportunities to identify the underlying structure of trade-offs between different objectives. This would also provide a quantitative basis for many types of practical advice in supply chain management.

## References

[1] H Ackermann, E Diessel, and SO Krumke. "Robust Flows with Adaptive Mitigation." In: *EURO Journal on Computational Optimization* (2020). Submitted.

[2] H Ackermann et al. "Decision Support for Material Procurement." In: *Operations Research Proceedings 2019*. Ed. by J Neufeld et al. In press. Springer, 2020. DOI: 10.1007/978-3-030-48439-2.

[3] SM Ali and K Nakade. "A mathematical optimization approach to supply chain disruptions management considering disruptions to suppliers and distribution centers." In: *Operations and supply chain management* 8.2 (2015), pp. 57–66.

[4] P Belotti, B Soylu, and MM Wiecek. "Fathoming rules for biobjective mixed integer linear programs: Review and extensions." In: *Discrete Optimization* 22 (2016), pp. 341–363. DOI: 10.1016/j.disopt.2016.09.003.

[5] A Ben-Tal et al. "Adjustable robust solutions of uncertain linear programs." In: *Mathematical Programming* 99.2 (2004), pp. 351–376. DOI: 10.1007/s10107-003-0454-y.

[6] J Birge and F Louveaux. *Introduction to Stochastic Programming*. Springer, 2011.

[7] N Boland, H Charkhgard, and M Savelsbergh. "The Triangle Splitting Method for Biobjective Mixed Integer Programming." In: *Integer Programming and Combinatorial Optimization*. Ed. by J Lee and J Vygen. Vol. 8494. Lecture Notes in Computer Science. Cham: Springer, 2014, pp. 162–173. DOI: 10.1007/978-3-319-07557-0_14.

[8] V Cacchiani and C D'Ambrosio. "A branch-and-bound based heuristic algorithm for convex multi-objective MINLPs." In: *European Journal of Operational Research* 260.3 (2017), pp. 920–933. DOI: 10.1016/j.ejor.2016.10.015.

[9] T Cui, Y Ouyang, and Z-JM Shen. "Reliable Facility Location Design Under the Risk of Disruptions." In: *Operations Research* 58.4 (2010), pp. 998–1011. DOI: 10.1287/opre.1090.0801.

[10] E Diessel. "An Adaptive Patch Approximation Algorithm for Bicriteria Convex Mixed Integer problems." In: *Optimization Online* (2020). URL: http://www.optimization-online.org/DB_HTML/2020/07/7935.html.

[11] E Diessel. *scgen: A Random Supply-Chain Generator*. Version 1.0. 2020. URL: https://github.com/erikdiessel/scgen.

[12] E Diessel. "Risk Aware Flow Optimization and Application to Logistics Networks." MA thesis. Technische Universität Kaiserslautern, Department of Mathematics, 2019.

[13] M Ehrgott. *Multicriteria Optimization*. Berlin/Heidelberg: Springer-Verlag, 2005. DOI: 10.1007/3-540-27659-9.

[14] M Eskandarpour et al. "Sustainable supply chain network design: An optimization-oriented review." In: *Omega* 54 (2015), pp. 11–32. DOI: 10.1016/j.omega.2015.01.006.

[15] B Fahimnia et al. "Quantitative models for managing supply chain risks: A review." In: *European Journal of Operational Research* 247.1 (2015), pp. 1–15. DOI: 10.1016/j.ejor.2015.04.034.

[16] A Fattahi and M Turkay. "A one direction search method to find the exact nondominated frontier of biobjective mixed-binary linear programming problems." In: *European Journal of Operational Research* 266.2 (2018), pp. 415–425. DOI: 10.1016/j.ejor.2017.09.026.

[17] K Govindan, M Fattahi, and E Keyvanshokooh. "Supply chain network design under uncertainty: A comprehensive review and future research directions." In: *European Journal of Operational Research* 263.1 (2017), pp. 108–141. DOI: 10.1016/j.ejor.2017.04.009.

[18] RL Graham et al. "Optimization and Approximation in Deterministic Sequencing and Scheduling: a Survey." In: *Discrete optimization*. Ed. by PL Hammer, EL Johnson, and BH Korte. Vol. 5. Annals of Discrete Mathematics. Amsterdam, New York, and New York, 1979, pp. 287–326. DOI: 10.1016/S0167-5060(08)70356-X.

[19] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2018. URL: http://www.gurobi.com.

[20] I Heckmann, T Comes, and S Nickel. "A critical review on supply chain risk – Definition, measure and modeling." In: *Omega* 52 (2015), pp. 119–132. DOI: 10.1016/j.omega.2014.10.004.

[21] E Huang and M Goetschalckx. "Strategic robust supply chain design based on the Pareto-optimal tradeoff between efficiency and risk." In: *European Journal of Operational Research* 237.2 (2014), pp. 508–518. DOI: 10.1016/j.ejor.2014.02.038.

[22] D Ivanov et al. "Literature review on disruption recovery in the supply chain." In: *International Journal of Production Research* 55.20 (2017), pp. 6158–6174. DOI: 10.1080/00207543.2017.1330572.

[23] M Kamalahmadi and MM Parast. "A review of the literature on the principles of enterprise and supply chain resilience: Major findings and directions for future research." In: *International Journal of Production Economics* 171 (2016), pp. 116–133. DOI: 10.1016/j.ijpe.2015.10.023.

[24] IY Kim and OL de Weck. "Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation." In: *Structural and Multidisciplinary Optimization* 31.2 (2006), pp. 105–116. DOI: 10.1007/s00158-005-0557-6.

[25] W Klibi and A Martel. "Scenario-based Supply Chain Network risk modeling." In: *European Journal of Operational Research* 223.3 (2012), pp. 644–658. DOI: 10.1016/j.ejor.2012.06.027.

[26] G Mavrotas and D Diakoulaki. "A branch and bound algorithm for mixed zero-one multiple objective linear programming." In: *European Journal of Operational Research* 107.3 (1998), pp. 530–541. DOI: 10.1016/S0377-2217(97)00077-5.

[27] M Mori et al. "Risk-cost optimization for procurement planning in multi-tier supply chain by Pareto Local Search with relaxed acceptance criterion." In: *European Journal of Operational Research* 261.1 (2017), pp. 88–96. DOI: 10.1016/j.ejor.2017.01.028.

[28] T Perini et al. "A Criterion Space Method for Biobjective Mixed Integer Programming: The Boxed Line Method." In: *INFORMS Journal on Computing* (2019), pp. 16–39. DOI: 10.1287/ijoc.2019.0887.

[29] V Rajagopal, S Prasanna Venkatesan, and M Goh. "Decision-making models for supply chain risk mitigation: A review." In: *Computers & Industrial Engineering* 113 (2017), pp. 646–682. DOI: 10.1016/j.cie.2017.09.043.

[30] G Rote. "The convergence rate of the sandwich algorithm for approximating convex functions." In: *Computing* 48.3-4 (1992), pp. 337–361. DOI: 10.1007/BF02238642.

[31] T Stidsen, KA Andersen, and B Dammann. "A Branch and Bound Algorithm for a Class of Biobjective Mixed Integer Programs." In: *Management Science* 60.4 (2014), pp. 1009–1032. DOI: 10.1287/mnsc.2013.1802.

[32] CS Tang. "Perspectives in supply chain risk management." In: *International Journal of Production Economics* 103.2 (2006), pp. 451–488. DOI: 10.1016/j.ijpe.2005.12.006.

[33] B Tomlin. "On the Value of Mitigation and Contingency Strategies for Managing Supply Chain Disruption Risks." In: *Management Science* 52.5 (2006), pp. 639–657. DOI: 10.1287/mnsc.1060.0515.

[34] S Vassilvitskii and M Yannakakis. "Efficiently computing succinct trade-off curves." In: *Theoretical Computer Science* 348.2-3 (2005), pp. 334–356. DOI: 10.1016/j.tcs.2005.09.022.