CHALMERS
UNIVERSITY OF TECHNOLOGY

A criterion space decomposition approach to generalized tri-objective tactical resource allocation

Downloaded from: https://research.chalmers.se, 2024-10-12 04:30 UTC

Citation for the original published paper (version of record):
Fotedar, S., Strömberg, A., Almgren, T. et al (2023). A criterion space decomposition approach to generalized tri-objective tactical resource
allocation. Computational Management Science, 20(1).
http://dx.doi.org/10.1007/s10287-023-00442-6

N.B. When citing this work, cite the original published paper.

research.chalmers.se offers the possibility of retrieving research publications produced at Chalmers University of Technology. It covers all kind of research output: articles, dissertations, conference papers, reports etc. since 2004. research.chalmers.se is administrated and maintained by Chalmers Library

(article starts on next page)

Computational Management Science (2023) 20:17
https://doi.org/10.1007/s10287-023-00442-6

ORIGINAL PAPER

A criterion space decomposition approach to generalized tri-objective tactical resource allocation

Sunney Fotedar[1] · **Ann-Brith Strömberg**[1] · **Torgny Almgren**[2] · **Stefan Cedergren**[2]

Received: 1 March 2022 / Accepted: 31 January 2023
© The Author(s) 2023

Abstract

We present a tri-objective mixed-integer linear programming model of the tactical resource allocation problem with inventories, called the *generalized tactical resource allocation problem* (GTRAP). We propose a specialized criterion space decomposition strategy, in which the projected two-dimensional criterion space is partitioned and the corresponding sub-problems are solved in parallel by application of the *quadrant shrinking method* (QSM) (Boland in Eur J Oper Res 260(3):873–885, 2017) for identifying non-dominated points. To obtain an efficient implementation of the parallel variant of the QSM we suggest some modifications to reduce redundancies. Our approach is tailored for the GTRAP and is shown to have superior computational performance as compared to using the QSM without parallelization when applied to industrial instances.

Keywords Discrete tri-objective optimization · Resource allocation · Quadrant shrinking method · Parallel computing · Combinatorial optimization

✉ Sunney Fotedar
sunney@chalmers.se

Ann-Brith Strömberg
anstr@chalmers.se

Torgny Almgren
torgny.almgren@gknaerospace.com

Stefan Cedergren
stefan.cedergren@gknaerospace.com

[1] Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, SE-412 96 Gothenburg, Sweden

[2] GKN Aerospace Sweden AB, 461 38 Trollhättan, Sweden

Published online: 27 March 2023

1 Introduction

We present the *generalized tactical resource allocation problem* (GTRAP), which is a generalization of a previously introduced *tactical resource allocation problem* (TRAP); (Fotedar et al. 2022). The TRAP is a bi-objective *multi-item capacitated production planning problem* with a medium–to–long-term planning horizon defined for a large tier-1 aerospace engine system manufacturer. The bi-objective mixed-integer linear programming (MILP) model defined for the TRAP identifies routes that a product/part may take through the factory over a long planning horizon. There is a tendency (in our case company GKN Aerospace)[1] to utilize a few capable machines (w.r.t. tolerances, speed, and functionality) to a higher extent than other machines. This results in a resource loading imbalance in the production system which may further result in production delays. To relieve loading imbalance we consider *qualifying* alternative machines for a given type of task/job of the TRAP. Consequently, in the TRAP tasks/jobs are allocated to machines with the bi-objective goal to minimize both qualification costs and resource loading imbalance, while demand in each time period and various other side constraints are satisfied. The GTRAP, on the other hand, allows building some inventories between time periods and defines a third objective function, thus resulting in a tri-objective MILP. This work facilitates an understanding of the trade-off between the three objectives as well as the effect on the two previously defined objectives for the TRAP.

When facing real-world decision problems, decision-makers want to be mindful of the trade-off between different objectives/goals of the stakeholders, which are often in conflict. Therefore, various such problems are modeled as multi-objective optimization problems (MOOPs), which yield a set of so-called *efficient solutions* (corresponding objective vectors are called non-dominated points (NDPs)). This set contains solutions such that none of the objective functions' values can be improved (reduced in a minimization problem) without degrading (increasing for a minimization problem) one or several objective functions' value(s). A common method to solve MOOPs is the *criterion/objective space search method*.[2] Different criterion space search methods have varying effects on the computational performance depending on the problem properties and the instances. In this work, we present an approach to enhance the computational performance of the *Quadrant shrinking method* (QSM) (Boland et al. 2017) by means of parallelization.

1.1 Contribution

Our contribution is two-fold. Firstly, we modify the previously defined TRAP (Fotedar et al. 2022) to include inventories of both semi-finished and finished products/

[1] GKN Aerospace is a leading manufacturer of engine parts, e.g. fans at the front of jet engines or gas turbines, rotors, and stators for compressing the air and other turbine inlet and exit structures.

[2] For several problem classes decision space search methods have also shown positive results. However, our focus is on criterion space search methods as they exploit the power of MILP solvers (such as `Gurobi` or `CPLEX`). Hence, any future computational enhancement of the solver will be replicated for the end-user of our approach as well.

parts. We refer to this model as the GTRAP. We provide a mathematical formulation of the problem at hand. Secondly, we present a few propositions for our problem which are utilized in the development of a tailored approach to partition the criterion space. The resulting sub-problems are solved in parallel using the criterion space *Quadrant shrinking method* (QSM) by Boland et al. (2017). The choice of the QSM is due to its utilization of the so-called concept of *projection onto the two-dimensional criterion space*, which can easily be parallelized due to the properties of the GTRAP. This, along with some modifications to reduce redundancies while computing NDPs, results in an improved computational performance as compared to simply using the QSM without parallelization. Computational experiments are performed for numerical instances provided by GKN Aerospace.

1.2 Outline

In Sect. 2, we discuss literature concerning both tactical production planning models and multi-objective optimization methods. A mathematical model for the GTRAP is defined in Sect. 3. In Sect. 4, we present an approach to decompose the *projected two-dimensional criterion space* (see Def. 2) and define sub-problems solved in parallel. We also present modifications necessary for the effective implementation of the parallel variant of the QSM, the P-QSM. In Sect. 5, we test our approach on fifteen industrial instances and compare the computational performance of the QSM to that of our proposed parallel variant, the P-QSM. We perform sensitivity analyses to assess the performance of P-QSM and QSM by varying the loading threshold (an input parameter in our model). We assess the quality of resulting representations of the Pareto front using *cardinality* and *coverage gap* values.

2 Literature, scope, and the quadrant shrinking method (QSM)

We next present key aspects of tactical planning models, define the scope of the GTRAP, and provide some classical results for multi-objective (integer) linear optimization methods (criterion space search methods in particular).

2.1 Tactical planning literature and model scope

Tactical production planning determines material flows, inventory levels, and resource utilization; it is usually done 1–4 years in advance (in the Aerospace industry) and acts as an input to operational models, such as machine scheduling. There are five main characteristics of tactical planning problems (Díaz-Madroñero et al. 2014): (a) number of products/items and structure of bill–of–materials/levels (e.g. series, assembly, general, arborescence) (Pochet and Wolsey 2006, Chapter 13); (b) uncertainty in demand/processing times/costs; (c) time discretization; (d) capacity constraints; (e) types of objective functions. In any modern production system, multiple products are produced, each requiring at least one part/component. Several

articles model uncertainty using stochastic programming (Lan et al. 2011; Nourelfath 2011), less commonly using fuzzy sets (Chen and Huang 2010; Lan et al. 2011), and sometimes by robust approaches (Wei et al. 2011; Genin et al. 2008). Mainly two types of time discretizations are considered: *short time buckets*, with enough time to manufacture only one part/item; *long time buckets*, in which multiple parts or an entire product is produced. Most companies employ a so-called *rolling horizon* and prepare a production schedule only when a customer order has been created in their planning system which is generally a few hours/days/weeks (varying between companies) before the delivery time/date is fixed. Hence, simultaneous scheduling and tactical allocation are not that useful in an industrial setting. Generally, tactical planning models have *long time buckets* which vary between months and quarters of a year, depending on the lead times of products. The capacity constraints in tactical models are on machines (hours available), man-hours, fixtures, tools, and inventory levels. The most common type of objective function minimizes cost/time (processing time, set-up time, and fixture cost (Bradley and Glynn 2002; Mieghem 2003)).

The GTRAP is a multi-item, multi-level (series assembly) tactical production planning model. The series product structure is due to each product requiring a sequence of (machining) operations such as milling, turning, and grinding. The raw material undergoes a series of operations to result in a product. The capacity limitation is on the number of raw materials available, machine capacity (time limitations), availability of technical staff to verify whether an operation can be performed in a machine, and an upper limit on the total inventory. The demand is considered deterministic due to long-term contracts and the availability of an accurate forecast (over a long time bucket). We consider three objective functions: (a) minimizing the maximum resource (machine) loading above a given threshold, (b) minimizing the qualification cost required to verify that a task can be performed in a machine, and (c) minimizing the total inventory (mathematical details in Sect. 3).

2.2 Multi-objective optimization literature

The two key computational challenges in the effective implementation of a criterion space search method (for discrete MOOPs)[3] are the number/type of scalarized problems solved to identify all the NDPs, and efficiently update the search region (required to identify the region containing yet-unknown NDPs). For the latter Dächert et al. (2017) and Klamroth et al. (2015) provide efficient approaches based on updating local upper bounds, but neither of them discusses what type of scalarization to use. This work focuses on the former since for the GTRAP most of the computational effort is spent on solving the scalarized problems. Laumanns et al. (2006) provide a theoretical bound of $(|F_{\text{ndp}}| + 1)^{k-1}$ scalarized problems required to find all NDPs of a discrete MOOP, k and $|F_{\text{ndp}}|$ being the number of objectives and NDPs, respectively. Dächert and Klamroth (2014) are (to the best of our knowledge) the first to provide linear bounds $2|F_{\text{ndp}}| - 1$ on the number of scalarized problems required to identify all the NDPs of a tri-objective integer linear programming

[3] a class of MOOPs (can be mixed-integer problems) with a finite number of NDPs.

(TOILP) problem using the *split criterion space search method*. Another method specially designed for TOILP problems is the so-called *L-shaped method* (LSM); the number of scalarizations solved by LSM is, however, bounded between $2|F_{\text{ndp}}| + 1$ and $|F_{\text{ndp}}|^2 + |F_{\text{ndp}}| + 1$ (Boland et al. 2015, Theorem. 14). Theoretically, the split method of Dächert and Klamroth (2014) is superior to LSM. There is, however, a trade-off between solving fewer scalarized problems (Dächert and Klamroth 2014) with increasing difficulty (as more disjunctive constraints are added) and solving more scalarized problems of manageable sizes (Boland et al. 2015). One of the benefits of LSM is that it produces high-quality approximate efficient frontiers faster than the split method in (Dächert and Klamroth 2014). The computational benefits of LSM are demonstrated by numerical tests on numerous publicly available instances; see (Boland et al. 2015).

The main idea behind the split method by Dächert and Klamroth (2014) is the following: at an iteration $t \geq 2$, the algorithm searches for a NDP, $\mathbf{f}^t \notin \{\mathbf{f}^1, \ldots, \mathbf{f}^{t-1}\}$. This is done by adding $t - 1$ binary variables that impose $\mathcal{O}(t - 1)$ disjunctive constraints, to remove the dominated points from the search space. Consequently, with an increasing number of iterations, the scalarized problems grow larger and become much harder to solve to (near-)optimality. To avoid adding too many disjunctive constraints and binary variables, various methods suggest a decomposition of the criterion space using projections onto a two-dimensional criterion space. One such method is the *recursive method* (Özlen et al. 2013) which searches the projected two-dimensional criterion space by restricting the value of the third objective function. The projected two-dimensional criterion space can be explored using the well-known ε-constraint method. The LSM (Boland et al. 2015, Sect. 4.2) uses both rectangular as well as L-shapes in the projected two-dimensional criterion space. To reduce the number of single-objective scalarized problems required to identify all the NDPs, the authors developed the so-called *quadrant shrinking method* (QSM) (Boland et al. 2017, Algorithm 1) which requires at most $3|F_{\text{ndp}}| + 1$ single-objective ILPs to be solved as opposed to $|F_{\text{ndp}}|^2 + |F_{\text{ndp}}| + 1$ in the LSM. Boland et al. (2017) showed that the QSM has a better computational performance than both the LSM and the split method, as well as the enhanced recursive algorithm by Özlen et al. (2013) on various publicly available instances.

Parallel methods for discrete MOOPs. Each scalarization of a discrete MOOP is generally $\mathcal{NP}$-hard; hence, industry-size instances may require long computing times, which may discourage end-users from utilizing a decision support tool relying on the solution of multiple scalarized problems to near-optimality. On the other hand, the rapid growth of technology concerning processors (multi-core), networks, and architecture have led to parallelization becoming a subject of interest within the optimization community. Already, there has been an extensive parallelization of the branch–and–bound method, which is utilized in all commercial MILP solvers such as Gurobi and CPLEX; (Bixby et al. 1999). There are some frameworks presented for the parallelization of metaheuristics used to solve MOOPs (summary in (Talbi et al. 2008, Ch. 13.2)). Such general frameworks do, however, rarely exist for exact criterion space search methods, one reason being that most exact criterion space search methods are sequential and require

a solution to the previous scalarized problem to adjust the current one appropriately. A few articles parallelize parts of a given criterion space search method. For instance, Dhaenens et al. (2006) solve a bi-objective flow-shop problem by parallelizing the weighted sum method using so-called *dichotomic search*, which initiates two new parallel searches once a solution is found. Hence, several processors are idle in the first phase of the approach. Lemesre et al. (2007) present a parallel approach to solving the bi-objective flow-shop problem in which the two-phase method in (Ulungu and Teghem 1994) is parallelized. To the best of our knowledge, discrete MOOPs with more than two objectives have not been considered for parallelization in the literature.

2.3 Preliminaries, and the QSM

To enable parallel computing, we must first identify an appropriate criterion space search method. We choose the QSM (Boland et al. 2017) due to two reasons: (a) as indicated in the previous subsection it has a superior computational performance (for several benchmarking instances); (b) most importantly, it works on the projected two-dimensional criterion space, thus facilitating an efficient decomposition of the criterion space (discussed in Sect. 4). The QSM is briefly presented, starting with some important notations and definitions.

Let $k \geq 2$ (we use the notations only for comparing vectors, for scalars regular notations are used). For any two vectors $\mathbf{u}, \mathbf{w} \in \mathbb{R}^k$ it holds that

$$\mathbf{u} \leqq \mathbf{w} \quad \Longleftrightarrow \quad w_i \in [u_i, \infty) \quad \forall i \in \{1, \dots, k\}; \tag{1a}$$

$$\mathbf{u} \leq \mathbf{w} \quad \Longleftrightarrow \quad w_i \in [u_i, \infty) \quad \forall i \in \{1, \dots, k\} \text{ and } \mathbf{u} \neq \mathbf{w}; \tag{1b}$$

$$\mathbf{u} < \mathbf{w} \quad \Longleftrightarrow \quad w_i \in (u_i, \infty) \quad \forall i \in \{1, \dots, k\}. \tag{1c}$$

Definition 1 (*Efficient solutions*) *Given a TOILP problem*

$$\min_{\mathbf{x} \in X} \left\{ f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}) \right\}, \tag{2}$$

where the feasible set $X \subset \mathbb{Z}^n_+$ is defined by a set of affine constraints in the decision space, and the objective functions $f_1, f_2, f_3 : \mathbb{Z}^n_+ \mapsto \mathbb{R}_+$ are linear and non-negative on the feasible set. The image F of X, defined as $F := \mathbf{f}(X) := \{ \mathbf{f} \in \mathbb{R}^3_+ \mid \mathbf{f} = \mathbf{f}(\mathbf{x}) \text{ for some } \mathbf{x} \in X \}$, represents the feasible set in the criterion space.

A solution $\mathbf{x}^* \in X$ is called *weakly efficient* in (2) if there exists no feasible solution $\mathbf{x} \in X$ such that $\mathbf{f}(\mathbf{x}) < \mathbf{f}(\mathbf{x}^*)$.

A solution $\mathbf{x}^* \in X$ to (2) is called *efficient* if there exists no feasible solution $\mathbf{x} \in X$ such that $\mathbf{f}(\mathbf{x}) \leq \mathbf{f}(\mathbf{x}^*)$. The corresponding point in the criterion space $\mathbf{f}(\mathbf{x}^*)$ is called a *non-dominated point* (NDP). A set of efficient solutions is denoted as X_{eff}.

Any two points $\mathbf{u}^1, \mathbf{u}^2 \in \mathbb{R}^2_+$ such that $\mathbf{u}^1 \leq \mathbf{u}^2$ define a rectangle (or a line segment if either $u^1_1 = u^2_1$ or $u^1_2 = u^2_2$) $R(\mathbf{u}^1, \mathbf{u}^2) := \{\mathbf{u} \in \mathbb{R}^2_+ \mid u^1_1 \leq u_1 \leq u^2_1, u^1_2 \leq u_2 \leq u^2_2\}$. The next definition is used in the QSM and several other methods employing the projection of NDPs onto lower-dimensional subspaces.

Definition 2 (*Projection onto two-dimensional criterion space*) We denote the projection of a point $\mathbf{f} := (f_1, f_2, f_3)^\top \in \mathbb{R}^3_+$ onto the two-dimensional criterion space of the two first dimensions as $\hat{\mathbf{f}} := (f_1, f_2)^\top \in \mathbb{R}^2_+$. Subsequently, a *projected two-dimensional criterion space* is defined as the space that contains the points obtained by projecting each objective point corresponding to a feasible solution onto the two-dimensional criterion space.

The QSM works in a projected two-dimensional criterion space (w.l.o.g. we consider projection onto the first two objective functions). Given an upper bound (on the first two objectives) $\mathbf{u} \in \mathbb{R}^2_+$, the main operation is to find yet-unknown NDPs in $R(\mathbf{0}, \mathbf{u})$ (also referred to as a *quadrant with upper bound* $\mathbf{u}$). Consider a NDP $\mathbf{f}^* = \mathbf{f}(\mathbf{x}^*)$ possessing the property that its projection $\hat{\mathbf{f}}^* \in \mathbb{R}^2_+$ satisfies $\hat{\mathbf{f}}^* \leqq \mathbf{u}$ for a given $\mathbf{u}$ in the two-dimensional projected criterion space and that can be found by solving two integer programming problems (see Def. 3). The results are formally described in a general setting as follows.

Definition 3 (*Two-stage scalarization*; (*Kirlik and Sayın* 2014)) For $\mathbf{u} \in \mathbb{R}^2_+$, $\mathbf{f} : \mathbb{Z}^n_+ \to \mathbb{R}^3_+$ (linear and non-negative functions), $X \subset \mathbb{Z}^n_+$ and $k \in \{1, 2, 3\}$ the set $X^k(\mathbf{u})$ of optimal solutions to the two-stage scalarization of (2) is defined as

$$X^k(\mathbf{u}) := \quad \arg\min_{\mathbf{x} \in X} \sum_{j=1}^{3} f_j(\mathbf{x}), \tag{3a}$$

$$\text{s.t.} \qquad f_j(\mathbf{x}) \leq u_j, \quad j \in \{1, 2, 3\} \setminus \{k\}, \tag{3b}$$

$$f_k(\mathbf{x}) = \bar{f}_k(\mathbf{u}), \tag{3c}$$

where

$$\bar{f}_k(\mathbf{u}) := \quad \min_{\mathbf{x} \in X} \; f_k(\mathbf{x}), \tag{4a}$$

$$\text{s.t.} \quad f_j(\mathbf{x}) \leq u_j, \quad j \in \{1, 2, 3\} \setminus \{k\}. \tag{4b}$$

Theorem 4 (*Kirlik and Sayın* 2014, *Theorem* 1) *Let* $k \in \{1, 2, 3\}$. *For any* $\mathbf{u} \in \mathbb{R}^2_+$ *any optimal solution* $\mathbf{x}^k(\mathbf{u}) \in X^k(\mathbf{u})$ *to the two-stage scalarization Eqs.* (3)–(4) *is efficient for the TOILP problem* (2).

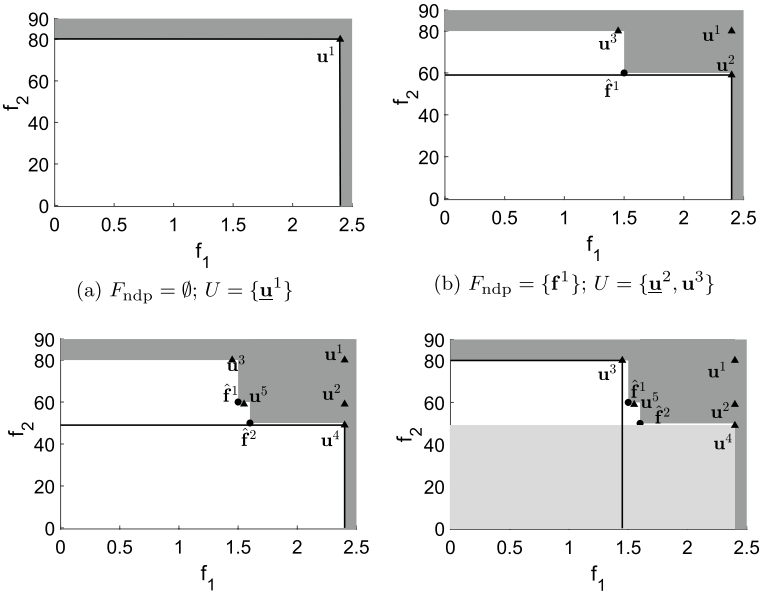


Fig. 1 In each sub-figure the set F_{ndp} (circles) contains projections of some of the NDPs onto the two-dimensional criterion space, while the set U (triangles) contains upper bounds on the quadrants used to search for NDPs (the upper bound of the explored quadrant is underlined). Shaded areas indicate regions without any projection of NDPs (dark) and with no feasible projection (light)

Theorem 5 (*Kirlik and Sayın* 2014, *Theorem* 2) *For any efficient solution* $\bar{\mathbf{x}}$ *of a TOILP* (2), *there exists an* $\mathbf{u} \in \mathbb{R}^2_+$ *such that for some* $k \in \{1, 2, 3\}$, $\bar{\mathbf{x}} \in X^k(\mathbf{u})$ *holds, i.e.* $\bar{\mathbf{x}}$ *is optimal in the two-stage scalarization* (3)–(4).

To give a brief overview of the QSM we provide an example (for more details, see (Boland et al. 2017, Algorithm 1, pp. 875–877)). Let F_{ndp} be the set of NDPs identified and U be the set of upper bounds of quadrants that may contain projections onto the (g_1, g_2)-criterion space of unidentified NDPs. The set U is initialized with the pre-defined upper bounds $\mathbf{u}^1 \in \mathbb{R}^2_+$ on the first two objectives, as depicted in Fig. 1a. We apply the two-stage scalarization (see Def. 3) for the vector $\mathbf{u}^1$. The resulting solution $\mathbf{x}^1 \in X^3(\mathbf{u}^1)$ corresponds to the NDP $\mathbf{f}^1 \in \mathbb{R}^3_+$, which is such it has the least value for the third objective. Consequently, the set $F_{\text{ndp}} = \{\mathbf{f}^1\}$. The projection $\hat{\mathbf{f}}^1 = (f^1_1, f^1_2)^\top$ is illustrated in Fig. 1b. From (Boland et al. 2017, Prop. 6) follows that the search region $\{\, \hat{\mathbf{f}} \in \mathbb{R}^2 \mid \hat{\mathbf{f}} \geq \hat{\mathbf{f}}^1 \,\}$ contains projections of points that are dominated by the point $\mathbf{f}^1 \in \mathbb{R}^3_+$. Hence, the dark grey-shaded region does not contain projections of NDPs. Subsequently, the set U of upper bounds is updated with the inclusion of $\mathbf{u}^2$ and $\mathbf{u}^3$, and the removal of $\mathbf{u}^1$. To avoid recomputing $\mathbf{f}^1$, the upper bounds are defined as $\mathbf{u}^2 = (u^1_1, \hat{f}^1_2 - \delta_2)^\top$, where $\delta_2 > 0$ is an appropriate discretization of f_2 (δ_2 can equal 1 if f_2 is integer-valued). Analogously, the usage of a discretization parameter $\delta_1 > 0$ results

in $\mathbf{u}^3 = (\hat{f}_1^1 - \delta_1, u_2^1)^\top$. In the next iteration, the quadrant $R(\mathbf{0}, \mathbf{u}^2)$ is searched for NDPs; the NDP $\mathbf{f}^2$ is found (its projection $\hat{\mathbf{f}}^2$ is illustrated in Fig. 1c). The set F_{ndp} is updated by the inclusion of $\mathbf{f}^2$, and U by the removal of $\mathbf{u}^2$ and the inclusion of $\mathbf{u}^4$ and $\mathbf{u}^5$. In the next iteration, $R(\mathbf{0}, \mathbf{u}^4)$ is searched for NDPs, resulting in an infeasible model (4); hence, no objective vector is projected onto this region (light grey-shaded in Fig. 1d). A reasoning about the order of removal and inclusion of elements from/into the set U is given in (Boland et al. 2017, Lemma 11).

3 Description of the GTRAP

Definition 6 (*Generalized Tactical Resource Allocation Problem* (*GTRAP*)) Given a set $\mathcal{L}$ of part/product types, for each part/product $\ell \in \mathcal{L}$ an ordered sequence $\mathcal{J}_\ell := \{1, \dots, J_\ell\}$ of manufacturing operations are performed, starting from raw material and ending with a finished part, when all operations in $\mathcal{J}_\ell$ are performed. In each time period $t \in \mathcal{T}$, orders of *task* $(j, \ell) \in \mathcal{M}$, must be assigned to one or several machines from the set $\mathcal{K}$. Let p_{jk}^ℓ be the processing time (including set-up time) of task (j, ℓ) when performed in a compatible machine $k \in \mathcal{K}_j^\ell \subseteq \mathcal{K}$. Each machine $k \in \mathcal{K}$ has the capacity C_{kt} (time units) in time period $t \in \mathcal{T}$ and a (relative) loading threshold $\zeta \in [0, 1]$. The demand a_t^ℓ of part $\ell \in \mathcal{L}$ in time period $t \in \mathcal{T}$ must be met. The number of machines allocated to the same task in each time period may not exceed the value of the parameter $\tau \in \mathbb{Z}_+$. Any task–to–machine assignment (j, ℓ, k) such that $k \in \mathcal{N}_j^\ell \subseteq \mathcal{K}_j^\ell$ requires a so-called *qualification* (or verification), which generates the additional one-time cost β_{jk}^ℓ, $j \in \mathcal{J}_\ell$, $\ell \in \mathcal{L}$. For (pre-qualified) machines $k \in \mathcal{K}_j^\ell \setminus \mathcal{N}_j^\ell$ the assignment (j, ℓ, k) does not require a qualification. Since inventories of semi-finished or finished parts[4] can be stored between time periods, the production in each time period may exceed or fall short of the demand. The total number of qualifications performed per time period t may not exceed the value of the parameter $\gamma \in \mathbb{Z}_+$. Upper limits $\bar{r}_{jt}^\ell$, $j \in \widetilde{\mathcal{J}}_\ell := \mathcal{J}_\ell \cup \{0\}$, $t \in \mathcal{T}$, $\ell \in \mathcal{L}$ (0 referring to the raw material) are set on the inventories. The total amount of raw material for $\ell \in \mathcal{L}$ ordered in each time period t may not exceed $d\, a_t^\ell$, where $d > 0$ is a user-defined limiting factor and a_t^ℓ is the demand of part type ℓ in time period t. □

The three objectives considered are the following: (i) minimize the sum (over time periods) of the maximum excess resource loading above the given threshold ζ; (ii) minimize the total qualification cost; (iii) minimize the total inventory. Since the objectives are measured in different units they cannot be combined in a single objective function, neither they obey any clear priority order but are considered equally important. The notations are summarized in Table 1.

[4] A part ℓ is finished when its raw material has gone through all operations in $\mathcal{J}_\ell$; it is semi-finished when its raw material has gone through at least one but not all operations in $\mathcal{J}_\ell$.

Table 1 Notations for the GTRAP

Sets	Description
$\mathcal{L} = \{1, \dots, L\}$	Set of part/product types
$\mathcal{J}_\ell = \{1, \dots, J_\ell\}$	Ordered set of indices of operations to produce part $\ell \in \mathcal{L}$; J_ℓ is the index of the final operation
$\mathcal{M}$	$:=\{(j, \ell) : j \in \mathcal{J}_\ell,\ \ell \in \mathcal{L}\}$ set of pairs (j, ℓ) of task indices
$\widetilde{\mathcal{J}}_\ell = \{0, \dots, J_\ell\}$	Extended ordered set of operations to produce a part type $\ell \in \mathcal{L}$ (the fictitious operation 0 represents raw material)
$\mathcal{K}$	Set of machines
$\mathcal{K}_j^\ell \subseteq \mathcal{K}$	Set of machines feasible for task $(j, \ell) \in \mathcal{M}$
$\mathcal{N}_j^\ell \subseteq \mathcal{K}_j^\ell$	Set of machines feasible but not qualified for task $(j, \ell) \in \mathcal{M}$
$\mathcal{T} = \{1, \dots, T\}$	Set of time periods (index 0 represents the beginning of time period 1)
Variables	**Description**
$x_{jkt}^\ell \in \mathbb{Z}_+$	Number of orders of task $(j, \ell) \in \mathcal{M}$ performed in machine $k \in \mathcal{K}_j^\ell$ in time period $t \in \mathcal{T}$
$s_{jkt}^\ell \in \{0, 1\}$	Equals 1 if an order of task (j, ℓ) is allocated to machine $k \in \mathcal{K}_j^\ell$ in time period $t \in \mathcal{T}$; equals 0 otherwise
$z_{jkt}^\ell \in \{0, 1\}$	Equals 1 if machine $k \in \mathcal{N}_j^\ell$ is qualified for task $(j, \ell) \in \mathcal{M}$, in time period $t \in \mathcal{T}$; equals 0 otherwise
$n_t \in \mathbb{R}_+$	Maximum resource loading above thresholds ζ, in time period $t \in \mathcal{T}$
$r_{jt}^\ell \in \mathbb{Z}_+$	Stock/inventory of raw material or (semi-)finished parts resulting from task $(j, \ell) \in \mathcal{M}$, at the end of time period $t \in \mathcal{T} \cup \{0\}$
$m_t^\ell \in \mathbb{Z}_+$	Total amount of raw materials ordered for part ℓ in time period t
Parameters	**Description**
$a_t^\ell \in \mathbb{Z}_+$	Demand/orders of part $\ell \in \mathcal{L}$ in time period $t \in \mathcal{T}$
$p_{jk}^\ell \in \mathbb{Q}_+$	Average processing time in machine $k \in \mathcal{K}_j^\ell$ for completing task $(j, \ell) \in \mathcal{M}$
$C_{kt} \in \mathbb{Z}_+$	Capacity (hours) available in machine $k \in \mathcal{K}$ in each time period $t \in \mathcal{T}$
$\beta_{jk}^\ell \in \mathbb{Z}_+$	Cost associated with qualifying machine $k \in \mathcal{N}_j^\ell$ for task $(j, \ell) \in \mathcal{M}$
$\gamma \in \mathbb{Z}_+$	Upper limit for the number of qualifications in any given time period
$\tau \in \mathbb{Z}_+$	Upper limit for the number of machines to which a given task can be assigned in each time period
$\zeta \in [0, 1]$	Loading threshold for machines
M_{jkt}^ℓ	$:= \min\left\{\sum_{q\in\{t,\dots,T\}} a_q^\ell; \left\lfloor C_{kt}/p_{jk}^\ell \right\rfloor\right\}$ upper limit for machine $k \in \mathcal{K}_j^\ell$ on the production of task $(j, \ell) \in \mathcal{M}$, in time period $t \in \mathcal{T}$
$\bar{r}_{jt}^\ell \in \mathbb{Z}_+$	Upper limit for the inventory of raw material/semi-finished/finished part resulting from performing task $(j, \ell) \in \mathcal{M}$ in time period $t \in \mathcal{T}$
$\bar{r}_{j0}^\ell \in \mathbb{Z}_+$	Initial inventory ($t = 0$) of raw material/semi-finished/finished part for $j \in \widetilde{\mathcal{J}}_\ell, \ell \in \mathcal{L}$
$d \in \mathbb{Z}_+$	Limiting factor for the order of the raw material in any time period

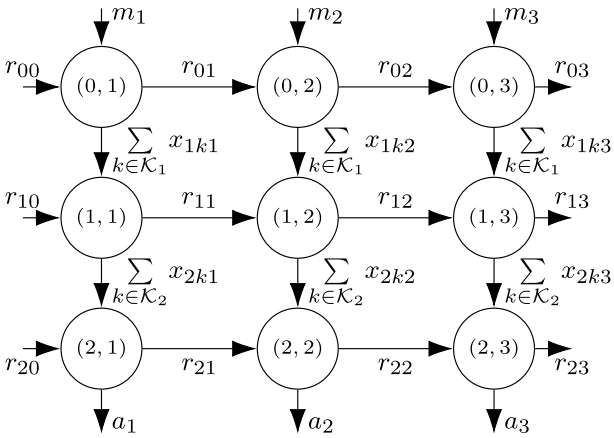


Fig. 2 Network illustration for a given $\ell \in \mathcal{L}$, where $J_\ell = 2$ (the upper index ℓ being omitted in the figure notations). The node labels denote for $j \in \widetilde{\mathcal{J}}_\ell$ and $t \in \mathcal{T}$

3.1 Network formulation

Fig. 2 illustrates the underlying network structure of the GTRAP for a single part $\ell \in \mathcal{L}$. The three horizontal levels (from top to bottom) correspond to $\widetilde{\mathcal{J}}_\ell = \{0, 1, 2\}$, while the three vertical levels (from left to right) represent $\mathcal{T} = \{1, 2, 3\}$. Each node corresponds to a 2-tuple (j, t), where $j \in \widetilde{\mathcal{J}}_\ell$ and $t \in \mathcal{T}$. The network should fulfill a balanced flow according to the following. Each node labelled (j, t) receives a flow of semi-finished ($0 < j < J_\ell$) or finished ($j = J_\ell$) parts, or raw materials ($j = 0$) either from the inventory (the variable $r^\ell_{j,t-1}$; horizontal incoming arcs) or after operation j is performed in a compatible machine in the set $\mathcal{K}^\ell_j$ in time period t (vertical incoming arcs) or raw materials (the variable m^ℓ_t). The flow out of node (j, t) either (if $j < J_\ell$) undergoes operation $j + 1$ in one of the compatible machines in $\mathcal{K}^\ell_{j+1}$ (vertical outgoing arcs), or (if $j = J_\ell$) it is used to meet the demand a^ℓ_t or results in the inventory r^ℓ_{jt} (horizontal outgoing arcs). The amount of raw material ordered in each time period is represented by m^ℓ_t.

We next present a mathematical model defining the set of feasible solutions to the GTRAP. The constraints (5a) model the flow balance of the raw materials for each part. The constraints (5b) model the flow balance for all semi-finished parts (when the raw material has gone through operations $j \in \mathcal{J}_\ell \setminus \{J_\ell\}$), for which there is no external demand. The constraints (5c) model the flow balance for (J_ℓ, ℓ, t), i.e. for the final operation to obtain a part $\ell \in \mathcal{L}$. The constraints (5d) set the initial inventory of semi-finished/finished parts to their respective input values. The constraints (5e) set the variables s^ℓ_{jkt} to one if $x^\ell_{jkt} > 0$; see Table 1 for the coefficients M^ℓ_{jkt}. The constraints (5f) restrict the number of machines for a given type of task in each time period to a user-defined value τ. The constraints (5g) set the upper limit on the total number of hours available in a given machine in each time period, and establish the

variables n_t, $t \in \mathcal{T}$, used in the first objective function. The constraints (5h) set the value of the variable z^{ℓ}_{jkt} to 1 if a task (j, ℓ) is to be qualified for a machine $k \in \mathcal{N}^{\ell}_j$ in time period t. These constraints imply that if $s^{\ell}_{jkq} = 1$, where $k \in \mathcal{N}^{\ell}_j$, $q \in \mathcal{T}$, then a qualification of machine k for task (j, ℓ) must be done once during the time periods $\{1, \dots, q\}$. The constraints (5i) define an upper limit on the total number of qualifications performed during each time period, due to a limited availability of technical staff. The constraints (5j) set upper limits for the inventories. Similarly, (5k) sets an upper limit on the maximum amount of raw materials ordered in each time period, which is due to the reluctance of the manufacturer to increase the tied-up working capital. The constraints (5l)–(5q) state that the variables **s** and **z** are binary, **r**, **x**, and **m** are non-negative integers, and $\mathbf{n} \geq \mathbf{0}$ are continuous.

$$m^{\ell}_t + r^{\ell}_{0,t-1} = r^{\ell}_{0t} + \sum_{k \in \mathcal{K}^{\ell}_1} x^{\ell}_{1kt}, \quad \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5a}$$

$$\sum_{k \in \mathcal{K}^{\ell}_j} x^{\ell}_{jkt} + r^{\ell}_{j,t-1} = r^{\ell}_{jt} + \sum_{k \in \mathcal{K}^{\ell}_{j+1}} x^{\ell}_{j+1,k,t}, \quad j \in \mathcal{J}_{\ell} \setminus J_{\ell}, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5b}$$

$$\sum_{k \in \mathcal{K}^{\ell}_{J_{\ell}}} x^{\ell}_{J_{\ell}kt} + r^{\ell}_{J_{\ell},t-1} = r^{\ell}_{J_{\ell}t} + a^{\ell}_t, \quad \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5c}$$

$$r^{\ell}_{j0} = \bar{r}^{\ell}_{j0}, \quad j \in \widetilde{\mathcal{J}}_{\ell}, \ell \in \mathcal{L}, \tag{5d}$$

$$x^{\ell}_{jkt} \leq M^{\ell}_{jkt} s^{\ell}_{jkt}, \quad k \in \mathcal{K}^{\ell}_j, j \in \mathcal{J}_{\ell}, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5e}$$

$$\sum_{k \in \mathcal{K}^{\ell}_j} s^{\ell}_{jkt} \leq \tau, \quad j \in \mathcal{J}_{\ell}, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5f}$$

$$\frac{1}{C_{kt}} \sum_{\ell \in \mathcal{L}} \sum_{j \in \mathcal{J}_{\ell}} p^{\ell}_{jk} x^{\ell}_{jkt} \leq n_t + \zeta \leq 1, \quad k \in \mathcal{K}, t \in \mathcal{T}, \tag{5g}$$

$$\sum_{t \in \mathcal{T}: t \leq q} z^{\ell}_{jkt} \geq s^{\ell}_{jkq}, \quad k \in \mathcal{N}^{\ell}_j, j \in \mathcal{J}_{\ell}, \ell \in \mathcal{L}, q \in \mathcal{T}, \tag{5h}$$

$$\sum_{\ell \in \mathcal{L}} \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{N}^{\ell}_j} z^{\ell}_{jkt} \leq \gamma, \quad t \in \mathcal{T}, \tag{5i}$$

$$r^{\ell}_{jt} \leq \bar{r}^{\ell}_{jt}, \quad j \in \widetilde{\mathcal{J}}_{\ell}, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5j}$$

$$m_t^\ell \leq d \cdot a_t^\ell, \quad \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5k}$$

$$x_{jkt}^\ell \in \mathbb{Z}_+, \quad k \in \mathcal{K}_j^\ell, j \in \mathcal{J}_\ell, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5l}$$

$$s_{jkt}^\ell \in \mathbb{B}, \quad k \in \mathcal{K}_j^\ell, j \in \mathcal{J}_\ell, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5m}$$

$$z_{jkt}^\ell \in \mathbb{B}, \quad k \in \mathcal{N}_j^\ell, j \in \mathcal{J}_\ell, \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5n}$$

$$m_t^\ell \in \mathbb{Z}_+, \quad \ell \in \mathcal{L}, t \in \mathcal{T}, \tag{5o}$$

$$n_t \geq 0, \quad t \in \mathcal{T}, \tag{5p}$$

$$r_{jt}^\ell \in \mathbb{Z}_+, \quad j \in \widetilde{\mathcal{J}}_\ell, \ell \in \mathcal{L}, t \in \mathcal{T} \cup \{0\}. \tag{5q}$$

We consider three objective functions possessing equal priority, namely the excess resource loading, defined as $g_1 := \sum_{t \in \mathcal{T}} n_t$, the sum of all qualification costs $g_2 := \sum_{t \in \mathcal{T}} \sum_{\ell \in \mathcal{L}} \sum_{j \in \mathcal{J}_\ell} \sum_{k \in \mathcal{N}_j^\ell} \beta_{jk}^\ell z_{jkt}^\ell$, and the total inventory,[5] expressed as $g_3 := \sum_{t \in \mathcal{T}} \sum_{\ell \in \mathcal{L}} \sum_{j \in \widetilde{\mathcal{J}}_\ell} r_{jt}^\ell$. The feasible set for the GTRAP is $B := \{(\mathbf{y}, \mathbf{r}) \mid (\mathbf{y}, \mathbf{r}) \text{ satisfy } (5)\}$, where $\mathbf{y} := (\mathbf{x}, \mathbf{s}, \mathbf{z}, \mathbf{n}, \mathbf{m})$ and $\mathbf{r}$ are vectors of variables. The objective functions are defined as $g_1 : \text{proj}_y(B) \mapsto \mathbb{R}_+$, $g_2 : \text{proj}_y(B) \mapsto \mathbb{Z}_+$, and $g_3 : \text{proj}_r(B) \mapsto \mathbb{Z}_+$. All the variables except $\mathbf{n}$ take integer/binary values; hence, $\mathbf{n}$ will take only discrete values. Consequently, the GTRAP is a (discrete) tri-objective optimization problem such that it has a finite number of NDPs. The projection of each NDP onto the (g_1, g_2)-criterion space must be in $R(\mathbf{0}, \hat{\mathbf{g}}^{\text{ub}})$, where $\hat{\mathbf{g}}^{\text{ub}} \in \mathbb{R}_+^2$ is a vector of upper bounds on g_1 and g_2 over the feasible set B. The set of efficient solutions to the GTRAP is denoted as B_{eff} and the corresponding NDPs are denoted as $\mathbf{g}(B_{\text{eff}})$.

4 Decomposing the projected two-dimensional criterion space

The QSM as well as several other discrete tri-objective criterion space search methods include solving a sequence of single-objective (i.e. scalarized) optimization problems. One way to reduce the solution time is to solve these single-objective subproblems simultaneously on separate computers in a cluster. However, all criterion space search methods (including the QSM) generate NDPs sequentially and they are non-trivial to parallelize. We exploit the structure of the GTRAP to enable a parallel computing approach. The following definitions and propositions are used when presenting our proposed approach.

[5] Since the initial inventory is assumed to be available at no additional cost at the start of the planning period, the summations are made over $\mathcal{T}$, thus excluding time step 0.

Definition 7 (*Orthogonal projection*) The orthogonal projection of a set $S \subset \mathbb{R}^{n+m}$ onto the linear space $\mathbb{R}^n \times \{0\}^m$ is $\text{proj}_y(S) := \{\mathbf{y} \in \mathbb{R}^n : \exists\, \mathbf{z} \in \mathbb{R}^m \text{ s.t. } (\mathbf{y}, \mathbf{z}) \in S\}$.

Definition 8 (ε-constrained subset of S w.r.t. $\mathbf{g}$ and $\boldsymbol{\varepsilon}$) Let $k > 0$, $S \subset \mathbb{R}^{n+m}$, the function $\mathbf{g} : S \mapsto \mathbb{R}^k_+$, and $\boldsymbol{\varepsilon} \in \mathbb{R}^k_+$. An ε-constrained subset of S w.r.t. $\mathbf{g}$ and $\boldsymbol{\varepsilon}$ is defined as $S^{\mathbf{g}}(\boldsymbol{\varepsilon}) := \{(\mathbf{y}, \mathbf{z}) \in S \mid \mathbf{g}(\mathbf{y},\mathbf{z}) \leq \boldsymbol{\varepsilon}\} \subseteq S$, where $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{z} \in \mathbb{R}^m$.

We observe that if $B^{g_3}_{\text{eff}}(0) = \{\, (\mathbf{y},\mathbf{r}) \in B_{\text{eff}} \mid g_3(\mathbf{r}) \leq 0 \,\} \neq \emptyset$ holds (cf. Def. 8 with $S := B_{\text{eff}}$, $k := 1$, $\mathbf{g} := g_3$, and $\varepsilon := 0$). Then, since g_3 equals the sum of inventories, the constraints $\mathbf{r} = \mathbf{0}$ (no inventory) must hold. Consequently, it results in a bi-objective optimization model (with the objectives g_1 and g_2). A point $(g_1^*, g_2^*) \in \mathbb{R}^2_+$ is a NDP in the resulting bi-objective optimization problem exactly when $(g_1^*, g_2^*, 0)$ is a NDP corresponding to the tri-objective GTRAP. Next, we formally prove this in a more general setting.

Proposition 9 *Consider a discrete tri-objective optimization problem with a feasible set* $S \subset \mathbb{R}^{n_1}_+ \times \mathbb{Z}^{n_2}_+ \times \mathbb{Z}^m_+$, *being a mixed-integer polyhedral set of the variables* $\mathbf{y} \in \mathbb{R}^{n_1}_+ \times \mathbb{Z}^{n_2}_+$ *and* $\mathbf{z} \in \mathbb{Z}^m_+$, *where* $n_1 + n_2 = n \geq 2$ *and* $m \geq 2$. *The objective functions are* $g_1 : \text{proj}_y(S) \mapsto \mathbb{R}_+$, $g_2 : \text{proj}_y(S) \mapsto \mathbb{R}_+$, *and* $g_3 : \text{proj}_z(S) \mapsto \mathbb{R}_+$, *where* $g_i(\mathbf{y}) = \mathbf{c}_i^\top \mathbf{y}$, $i = 1, 2$, $g_3(\mathbf{z}) = \mathbf{c}_3^\top \mathbf{z}$, *and* $\mathbf{c}_i \geq \mathbf{0}$, $i = 1, 2, 3$. *Let* $\mathcal{D}$ *and* $\widehat{\mathcal{D}}$ *denote the non-empty and finite sets of NDPs of the MOOPs* $\min_{\mathbf{y},\mathbf{z}}\{(g_1(\mathbf{y}), g_2(\mathbf{y}), g_3(\mathbf{z})) : (\mathbf{y}, \mathbf{z}) \in S^{g_3}(0)\}$ *and* $\min_{\mathbf{y}}\{(g_1(\mathbf{y}), g_2(\mathbf{y})) : \mathbf{y} \in \text{proj}_y(S^{g_3}(0))\}$, *respectively. Then, it holds that* $\mathcal{D} = \{(\mathbf{u}, 0) \mid \mathbf{u} \in \widehat{\mathcal{D}}\,\}$.

Proof Since $\mathcal{D} \neq \emptyset$, it holds that $S^{g_3}(0) \neq \emptyset$. Let $(\mathbf{y}, \mathbf{z}) \in S^{g_3}(0)$; then $g_3(\mathbf{z}) \leq 0$. Further, since $\mathbf{c}_3 \geq \mathbf{0}^m$ and $\mathbf{z} \in \mathbb{Z}^m_+$, it holds that $g_3(\mathbf{z}) \geq 0$. It follows that $g_3(\mathbf{z}) = 0$ and $\mathbf{z} = \mathbf{0}^m$.

We first show that the inclusion $\{(\mathbf{u}, 0) \mid \mathbf{u} \in \widehat{\mathcal{D}}\,\} \subseteq \mathcal{D}$ holds; then that the inclusion $\mathcal{D} \subseteq \{(\mathbf{u}, 0) \mid \mathbf{u} \in \widehat{\mathcal{D}}\,\}$ holds.

Consider a NDP $\mathbf{u}^{\text{b}} := (g_1(\mathbf{y}^{\text{b}}), g_2(\mathbf{y}^{\text{b}}))^\top \in \widehat{\mathcal{D}}$ (i.e. for the bi-objective MOOP) and assume the contradiction, that $(\mathbf{u}^{\text{b}}, 0) \notin \mathcal{D}$. This implies there exists a NDP $(\mathbf{u}^*, 0) \in \mathcal{D}$ (with corresponding solution $(\mathbf{y}^*, \mathbf{0}) \in S^{g_3}(0)$) such that $(\mathbf{u}^*, 0) \leq (\mathbf{u}^{\text{b}}, 0)$ (see (1) for vector comparison) which implies that $u_1^* \leq u_1^{\text{b}}$, $u_2^* \leq u_2^{\text{b}}$, and $\mathbf{u}^* \neq \mathbf{u}^{\text{b}}$. Hence, it holds that $g_i(\mathbf{y}^*) \leq g_i(\mathbf{y}^{\text{b}})$, for $i = 1, 2$, and $(g_1(\mathbf{y}^*), g_2(\mathbf{y}^*)) \neq (g_1(\mathbf{y}^{\text{b}}), g_2(\mathbf{y}^{\text{b}}))$. We know that if $(\mathbf{y}^*, \mathbf{0}) \in S^{g_3}(0)$, by Def. 7

the inclusion $\mathbf{y}^* \in \text{proj}_y(S^{g_3}(0))$ holds. Hence, $\mathbf{y}^{\text{b}}$ is not an efficient solution in the bi-objective MOOP, which is a contradiction. The inclusion follows.

Now, consider $(\mathbf{u}^{\text{t}}, 0) \in \mathcal{D}$, where $u_i^{\text{t}} = g_i(\mathbf{y}^t), i = 1, 2$, and $g_3(\mathbf{0}^m) = 0$. Assume that $\mathbf{u}^{\text{t}} \notin \widehat{\mathcal{D}}$; then $\exists$ a NDP $\mathbf{u}^* \in \widehat{\mathcal{D}}$ such that $u_1^* \leq u_1^{\text{t}}$, $u_2^* \leq u_2^{\text{t}}$, and $\mathbf{u}^* \neq \mathbf{u}^{\text{t}}$. The corresponding solution fulfills $\mathbf{y}^* \in \text{proj}_y(S^{g_3}(0))$ (Def. 7); hence, $(\mathbf{y}^*, \mathbf{0}) \in S^{g_3}(0)$ holds, which implies $(\mathbf{u}^{\text{t}}, 0) \notin \mathcal{D}$, i.e. a contradiction. The proposition follows. □

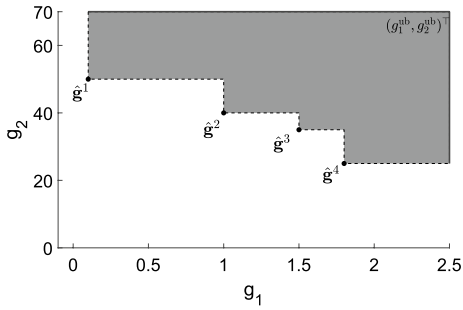


Fig. 3 The region of the projected two-dimensional criterion space that does not contain projections of NDPs is shaded in dark grey

Note that if $\mathbf{g}^* \in \mathbb{R}^3_+$ is a NDP of the GTRAP, the projection of $\mathbf{g}^*$ onto the (g_1, g_2)-criterion space is not dominated by the projection of any NDP satisfying $g_3 = 0$. Fig. 3 illustrates this for the upper bounds $g_1^{\text{ub}} = 2.5$ and $g_2^{\text{ub}} = 70$, along with the projections of the four elements in the set $\mathbf{g}(B^{g_3}_{\text{eff}}(0)) := \{\mathbf{g}(\mathbf{y}, \mathbf{r}) \mid (\mathbf{y}, \mathbf{r}) \in B^{g_3}_{\text{eff}}(0)\} = \{\mathbf{g}^1, \mathbf{g}^2, \mathbf{g}^3, \mathbf{g}^4\}$. The dark grey-shaded region contains no projections of NDPs onto the (g_1, g_2)-criterion space.

We next show this formally.

Proposition 10 *Consider a discrete tri-objective optimization problem with objective functions* $g_1 : \text{proj}_y(S) \mapsto \mathbb{R}_+$, $g_2 : \text{proj}_y(S) \mapsto \mathbb{R}_+$, *and* $g_3 : \text{proj}_z(S) \mapsto \mathbb{R}_+$, *where* $S \subset \mathbb{R}^{n_1}_+ \times \mathbb{Z}^{n_2}_+ \times \mathbb{Z}^m_+$ *is a mixed-integer polyhedral set of the variable vectors* $\mathbf{y} \in \mathbb{R}^{n_1}_+ \times \mathbb{Z}^{n_2}_+$ *and* $\mathbf{z} \in \mathbb{Z}^m_+$, *i.e.* $(\mathbf{y}, \mathbf{z}) \in S$. *Further*, $g_i(\mathbf{y}) = \mathbf{c}_i^\top \mathbf{y}$, $i = 1, 2$, $g_3(\mathbf{z}) = \mathbf{c}_3^\top \mathbf{z}$, $\mathbf{c}_i \geq \mathbf{0}$, $i = 1, 2, 3$ ($n_1 + n_2 \geq 2$, $m \geq 2$). *We denote the set of efficient solutions by* S_{eff}. *Then, the projections of the (finite number of) NDPs onto the two-dimensional* (g_1, g_2)*-criterion space are in the set (see the white region in Fig. 3):*

$$N(S^{g_3}_{\text{eff}}(0)) := \left\{ \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} \in \mathbb{R}^2_+ \,\middle|\, g_i \leq g_i^{\text{ub}}, i = 1, 2; \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} \not\geq \begin{pmatrix} g'_1 \\ g'_2 \end{pmatrix}, \forall \mathbf{g}' \in \mathbf{g}(S^{g_3}_{\text{eff}}(0)) \right\},$$

where g_1^{ub} *and* g_2^{ub} *denote the upper bounds on the values of the objective functions* g_1 *and* g_2, *respectively*, $S^{g_3}_{\text{eff}}(0) := \{(\mathbf{y}; \mathbf{z}) \in S_{\text{eff}} \mid g_3(\mathbf{z}) \leq 0\})$ *(see Def. 8), and the corresponding NDPs are denoted as* $\mathbf{g}(S^{g_3}_{\text{eff}}(0)) \subset \mathbb{R}^3_+$.

Proof Let $\mathbf{g}^* \in \mathbf{g}(S_{\text{eff}})$ be a NDP of the given tri-objective problem. We will show that $(g_1^*, g_2^*) \in N(S^{g_3}_{\text{eff}}(0))$ by considering the two cases $S^{g_3}_{\text{eff}}(0) = \emptyset$ and $S^{g_3}_{\text{eff}}(0) \neq \emptyset$.

Suppose $S^{g_3}_{\text{eff}}(0) = \emptyset$. Then, $\mathbf{g}(S^{g_3}_{\text{eff}}(0)) = \emptyset$ and $N(S^{g_3}_{\text{eff}}(0)) = \{(g_1, g_2) \in \mathbb{R}^2_+ | g_i \leq g_i^{\text{ub}}, i = 1, 2\}$. It is obvious that if $\mathbf{g}^*$ is a NDP (i.e. it has a corresponding feasible solution) then $(g_1^*, g_2^*) \leqq (g_1^{\text{ub}}, g_2^{\text{ub}})$; hence, $(g_1^*, g_2^*) \in N(S^{g_3}_{\text{eff}}(0))$.

Suppose the opposite, i.e. $S^{g_3}_{\text{eff}}(0) \neq \emptyset$. We assume contradictory that $\mathbf{g}^* \in \mathbf{g}(S_{\text{eff}})$ is such that $(g_1^*, g_2^*) \notin N(S^{g_3}_{\text{eff}}(0))$. Since $\mathbf{g}^* \in \mathbb{R}^3_+$ is a feasible objective vector, the inequalities $g_i^* \leq g_i^{\text{ub}}$, $i = 1, 2$, hold. Consequently, there exists a point $\mathbf{g}' \in \mathbf{g}(S^{g_3}_{\text{eff}}(0))$ fulfilling the inequality $(g_1^*, g_2^*) \geq (g'_1, g'_2)$. Since the least value of the third function is 0 and $g'_3 = 0$, we have that $(g_1^*, g_2^*, g_3^*) \geq (g'_1, g'_2, 0)$. Since $\mathbf{g}^*$ is dominated (at least

Table 2 Summary of set notations

Set	Description	
B	$:= \{(\mathbf{y}, \mathbf{r}) \mid (\mathbf{y}, \mathbf{r})$ satisfies (5)$\}$	
B_{eff}	$:= \{(\mathbf{y}^*, \mathbf{r}^*) \in B \mid \nexists\, \mathbf{g}(\mathbf{y}, \mathbf{r}) \leq \mathbf{g}(\mathbf{y}^*, \mathbf{r}^*),\ (\mathbf{y}, \mathbf{r}) \in B\}$	
$\mathbf{g}(B_{\text{eff}})$	$:= \{\mathbf{g}(\mathbf{y}, \mathbf{r}) \in \mathbb{R}^3_+ \mid (\mathbf{y}, \mathbf{r}) \in B_{\text{eff}}\}$	
$\widehat{\mathbf{g}}(B_{\text{eff}})$	$:= \{(g_1(\mathbf{y}), g_2(\mathbf{y})) \in \mathbb{R}^2_+ \mid \mathbf{g}(\mathbf{y}, \mathbf{r}) \in \mathbf{g}(B_{\text{eff}})\}$	(Note: $\widehat{\mathbf{g}} = (g_1, g_2)$)
$B^{g_3}(0)$	$:= \{(\mathbf{y}, \mathbf{r}) \in B \mid g_3(\mathbf{r}) \leq 0\}$	(Note: $g_3(\mathbf{r}) = 0 \iff \mathbf{r} = \mathbf{0}$)
$B^{g_3}_{\text{eff}}(0)$	$:= \{(\mathbf{y}, \mathbf{r}) \in B_{\text{eff}} \mid g_3(\mathbf{r}) \leq 0\}$	
$B^{\widehat{\mathbf{g}}}(\mathbf{u})$	$:= \{(\mathbf{y}, \mathbf{r}) \in B \mid g_i(\mathbf{y}) \leq u_i,\ i = 1, 2\}$	(Note: $\mathbf{u} \in \mathbb{R}^2_+$, $\widehat{\mathbf{g}} = (g_1, g_2)$)
$B^{\widehat{\mathbf{g}}}_{\text{eff}}(\mathbf{u})$	$:= \{(\mathbf{y}, \mathbf{r}) \in B_{\text{eff}} \mid g_i(\mathbf{y}) \leq u_i,\ i = 1, 2\}$	(Note: $\mathbf{u} \in \mathbb{R}^2_+$, $\widehat{\mathbf{g}} = (g_1, g_2)$)
$\mathbf{g}(B^{g_3}_{\text{eff}}(0))$	$:= \{\mathbf{g}(\mathbf{y}, \mathbf{r}) \in \mathbb{R}^3_+ \mid (\mathbf{y}, \mathbf{r}) \in B^{g_3}_{\text{eff}}(0)\}$	
$R(0, \mathbf{u}^k)$	$:= \{\mathbf{u} \in \mathbb{R}^2_+ \mid 0 \leq u_1 \leq u_1^k,\ 0 \leq u_2 \leq u_2^k\}$	
$N(B^{g_3}_{\text{eff}}(0))$	$:= \{(g_1, g_2) \in \mathbb{R}^2_+ \mid g_i \leq g_i^{\text{ub}}, i = 1, 2; (g_1, g_2) \not\geq (g_1', g_2'), \forall \mathbf{g} \in \mathbf{g}(B^{g_3}_{\text{eff}}(0))\}$, where g_1^{ub} (g_2^{ub}) is an upper bound on the first (second) objective	

weakly) by $\mathbf{g}'$, it is not a NDP. Hence, the assumption that $\mathbf{g}^* \in \mathbf{g}(S_{\text{eff}})$ cannot hold. The proposition follows. □

We summarize some of the set notations presented hitherto in Table 2.

4.1 Outline of the Algorithm

Consider $B^{g_3}_{\text{eff}}(0) \neq \emptyset$. We need to compute the NDPs of the tri-objective problem $\min\{(g_1(\mathbf{y}), g_2(\mathbf{y}), g_3(\mathbf{r})) : (\mathbf{y}, \mathbf{r}) \in B^{g_3}(0)\}$. These NDPs can be obtained by solving a corresponding discrete bi-objective optimization problem (cf. Prop. 9):

$$\min_{\mathbf{y} \in \text{proj}_y(B^{g_3}(0))} \{g_1(\mathbf{y}), g_2(\mathbf{y})\}.$$

We use the solution approach proposed by Fotedar et al. (2022) to solve the above bi-objective optimization problem (also known as the TRAP).

4.2 Defining the sub-problems

Once we identify the set $B^{g_3}_{\text{eff}}(0) \neq \emptyset$, the search region $N(B^{g_3}_{\text{eff}}(0)) \subset \mathbb{R}^2$ (as defined in Prop. 10, replacing S by B)[6] can be split into smaller regions such that their union covers the total search region. Let us consider $U(B^{g_3}_{\text{eff}}(0)) \subset \mathbb{R}^2_+$ as the set of upper

[6] Note that GTRAP is a discrete MOOP, as TRAP is proved to be a discrete bi-objective optimization problem in (Fotedar et al. 2022, Prop. 3).

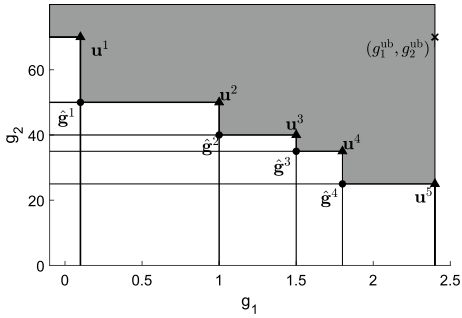


Fig. 4 Decomposing the projected (g_1, g_2)-criterion space. Circles represent projections of NDPs corresponding to $B^{g_3}_{\text{eff}}(0)$ onto this space; triangles represent upper bounds for the five sub-problems

bounds corresponding to the NDPs $\mathbf{g}(B^{g_3}_{\text{eff}}(0)) \subset \mathbb{R}^3_+$. The following two conditions must hold for a set of upper bounds;[7]

1. The inclusion $N(B^{g_3}_{\text{eff}}(0)) \subseteq \bigcup_{\mathbf{u} \in U(B^{g_3}_{\text{eff}}(0))} R(\mathbf{0}, \mathbf{u})$ holds;
2. For any two $\mathbf{u}^1, \mathbf{u}^2 \in U(B^{g_3}_{\text{eff}}(0))$ it holds that $R(\mathbf{0}, \mathbf{u}^1) \not\subset R(\mathbf{0}, \mathbf{u}^2)$.

The first condition ensures that the region $N(B^{g_3}_{\text{eff}}(0))$ is covered by the rectangles while the second ensures that none of the rectangles strictly contains other rectangles as that would lead to redundancies. Consequently, a sub-problem is defined for each quadrant $R(\mathbf{0}, \mathbf{u})$, with upper bounds $\mathbf{u} \in U(B^{g_3}_{\text{eff}}(0))$. Now, consider $K = |B^{g_3}_{\text{eff}}(0)|$ and define $K + 1$ sub-problems. The set of objective vectors $H^k \subset \mathbb{R}^3_+$ are obtained after solving the kth sub-problem using the QSM method on the set $B^{\hat{\mathbf{g}}}(\mathbf{u}^k) := \{(\mathbf{y}, \mathbf{r}) \in B \mid g_i(\mathbf{y}) \leq u^k_i, i = 1, 2\}$. Then, the equality $\bigcup_{k=1}^{K+1} H^k = \mathbf{g}(B_{\text{eff}})$ will be satisfied, as stated in Prop. 11, below.

We create an ordered list of the NDPs corresponding to $\mathbf{g}(B^{g_3}_{\text{eff}}(0))$ such that they are sorted in an increasing order of the first objective function's value.[8] We define a local upper bound $\mathbf{u}^k$ as

$$\mathbf{u}^k := \begin{cases} (g^1_1, g^{\text{ub}}_2), & \text{if } k = 1, \\ (g^k_1, g^{k-1}_2), & \text{if } 2 \leq k \leq K, \\ \left(g^{\text{ub}}_1, g^K_2\right), & \text{if } k = K + 1, \end{cases} \tag{6}$$

where $(g^{\text{ub}}_1, g^{\text{ub}}_2)$ are upper bounds on the first two objective functions. Fig. 4 illustrates an example with $|B^{g_3}_{\text{eff}}(0)| = 4$ satisfying the aforementioned conditions. The following proposition ensures that all the NDPs of the GTRAP are identified (see Sect. 1 (appendix) for a proof of the following proposition).

[7] Cf. the conditions for a set of upper bounds corresponding to a stable set in (Klamroth et al. 2015, Def. 2.3).

[8] The ordered set $\{\mathbf{g}^1, \ldots, \mathbf{g}^K\}$ of NDPs fulfills $g^1_1 < \ldots < g^K_1$.

Proposition 11 *Let* $\mathbf{u}^k$ *be defined by* (6) *and let* H^k *be the set of objective vectors obtained by applying the QSM to the GTRAP* (5) *with the additional constraints* $g_1(\mathbf{y}) \leq u_1^k$ *and* $g_2(\mathbf{y}) \leq u_2^k$. *If* $B_{\text{eff}}^{g_3}(0) \neq \emptyset$, *then the equality* $\bigcup_{k=1}^{K+1} H^k = \mathbf{g}(B_{\text{eff}})$ *holds, where* $K = |B_{\text{eff}}^{g_3}(0)|$.

4.3 Parallel QSM

We next present a brief overview of our approach, referred to as the Parallel QSM (P-QSM) and outlined in Algorithm 1. The first step initializes an empty set of efficient solutions to the GTRAP, i.e. B_{eff}. The discretization parameters δ_1 and δ_2 are used.[9] Next, the set $B_{\text{eff}}^{g_3}(0)$ is computed by first solving a bi-objective optimization model called the TRAP (without the variables $\mathbf{r}$) and applying Prop. 9. The algorithm includes three modifications to avoid several redundancies.

Algorithm 1 Parallel QSM (P-QSM)

$B_{\text{eff}} := \emptyset$; $\delta_1 \in \{0.05, 0.1\}$ and $\delta_2 := 1$
$B_{\text{eff}}^{g_3}(0) :=$ {subset of the efficient frontier of the TRAP}
$K := |B_{\text{eff}}^{g_3}(0)|$; $B_{\text{eff}} := B_{\text{eff}} \bigcup B_{\text{eff}}^{g_3}(0)$. Define $\mathbf{g}^i$, $i = 1, \ldots, K$, using $\mathbf{g}(B_{\text{eff}}^{g_3}(0))$ as in Sec. 4.2
if $g_1^K = g_1^{\text{ub}}$ **then** ▷ Third modification
  $E := \{1, \ldots, K\}$
else
  $E := \{1, \ldots, K+1\}$
end if
if $g_1^1 = 0$ **then**
  $E := E \setminus \{1\}$
end if
Identify values of the upper bounds $\{\tilde{\mathbf{u}}^i\}_{i \in E}$ using (6) and $\delta_2 = 1$ ▷ First modification
for $i \in E$ **do**
  if $i \neq K + 1$ **then**
    $\mathbf{u}^{\text{f}} := (\tilde{u}_1^i, g_2^i - 1)$; $\mathbf{u}^{\text{b}} := (g_1^i - \delta_1, \tilde{u}_2^i)$; $U := \{\mathbf{u}^f, \mathbf{u}^b\}$ ▷ Second modification
  else
    $U := \{\tilde{\mathbf{u}}^i\}$
  end if
  # Send the sub-problem for $R(\mathbf{0}, \tilde{\mathbf{u}}^i)$ to i^{th} computer
  $A^i :=$ {solutions from the application of QSM to an ordered set U}
end for
return $B_{\text{eff}} := B_{\text{eff}} \bigcup \{\bigcup_{i \in E} A^i\}$

4.3.1 Modifications

First modification. As the sub-problems are solved simultaneously, it is possible that some of the NDPs corresponding to $B_{\text{eff}}^{g_3}(0)$ are recomputed. For instance, in Fig. 4 while solving the third sub-problem we apply the QSM on $B^{\hat{\mathbf{g}}}(\mathbf{u}^3)$. It is obvious that we will recompute both $\mathbf{g}^2$ and $\mathbf{g}^3$. Note that both of these NDPs are also recomputed

[9] The second objective, g_2 is integer-valued; typically, $\delta_2 = 1$ and $\delta_1 \in \{0.05, 0.1\}$ are used.

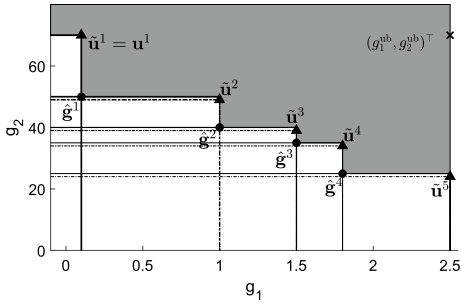


Fig. 5 First modification. Circles represent projections of NDPs corresponding to $B^{g_3}_{\text{eff}}(0)$ onto the (g_1, g_2)-criterion space; triangles represent modified upper bounds for each sub-problem

while solving the second, and the fourth sub-problems, respectively. To avoid this redundancy we simply modify the values of the upper bounds for each sub-problem.

Since the second objective function g_2 is integer-valued we define a new (adjusted) upper bound[10] $\tilde{u}^k_1 := u^k_1$, $\tilde{u}^k_2 := u^k_2 - 1$, for $k = 2, \ldots, |B^{g_3}_{\text{eff}}(0)| + 1$ and $\tilde{\mathbf{u}}^1 = \mathbf{u}^1$. Consequently, each quadrant $R(\mathbf{0}, \tilde{\mathbf{u}}^k)$, for $k = 1, \ldots, |B^{g_3}_{\text{eff}}(0)| + 1$ contains the projection of at most one NDP from the set $\mathbf{g}(B^{g_3}_{\text{eff}}(0))$. In the example in Fig. 5 each adjusted quadrant $R(\mathbf{0}, \tilde{\mathbf{u}}^k)$ is shown and the positions of the upper bounds are adjusted. We need to show that $\bigcup_{k=1}^{K+1} R(\mathbf{0}, \tilde{\mathbf{u}}^k)$, where $K = |B^{g_3}_{\text{eff}}(0)|$, contains projections of all the NDPs of the GTRAP onto the two-dimensional (g_1, g_2)-criterion space. It is formally stated as follows (see 1 (appendix) for the proof):

Proposition 12 *If $B^{g_3}_{\text{eff}}(0) \neq \emptyset$ then the projection of a NDP onto the (g_1, g_2)-criterion space of the GTRAP is contained in the set $\bigcup_{k=1}^{K+1} R(\mathbf{0}, \tilde{\mathbf{u}}^k)$, where $K = |B^{g_3}_{\text{eff}}(0)|$.*

Second modification. A NDP $\mathbf{g}^k \in \mathbf{g}(B^{g_3}_{\text{eff}}(0))$ is pre-computed for each sub-problem $k = 1, \ldots, |B^{g_3}_{\text{eff}}(0)|$, and by construction $g^k_3 = 0$ holds. We will again generate $\mathbf{g}^k$ first while applying the QSM for $B^{\hat{\mathbf{g}}}(\mathbf{u}^k)$ (see Table 2), i.e. the kth sub-problem. Consequently, each sub-problem $k = 1, \ldots, |B^{g_3}_{\text{eff}}(0)|$ is initialized with the two upper bounds $(\tilde{u}^k_1, g^k_2 - 1)$ and $(g^k_1 - \delta_1, \tilde{u}^k_2)$ instead of the pre-defined upper bound $\tilde{\mathbf{u}}^k$ used in the QSM. This avoids recomputing $\mathbf{g}^k, k = 1, \ldots, |B^{g_3}_{\text{eff}}(0)|$. For instance, in Fig. 5 for the second sub-problem, instead of exploring the quadrant $R(\mathbf{0}, \tilde{\mathbf{u}}^2)$, we search $R(\mathbf{0}, (\tilde{u}^2_1, g^2_2 - 1))$ and $R(\mathbf{0}, (g^2_1 - \delta_1, \tilde{u}^2_2))$ (see line 15 in Algorithm 1).

Third modification. Let $K := |B^{g_3}_{\text{eff}}(0)|$. If $g^K_1 = g^{\text{ub}}_1$, then $\mathbf{u}^K := (g^{\text{ub}}_1, g^{K-1}_2)$ follows from (6), and the modified upper bound is $\tilde{\mathbf{u}}^K = (g^{\text{ub}}_1, g^{K-1}_2 - 1)$. By construction, $u^K_2 = g^{K-1}_2 > g^K_2 = u^{K+1}_2$ (see (6)), and in this case $u^K_1 = g^{\text{ub}}_1 = u^{K+1}_1$. Consequently, $R(\mathbf{0}, \tilde{\mathbf{u}}^{K+1}) \subset R(\mathbf{0}, \tilde{\mathbf{u}}^K)$ holds. Hence, the $(K+1)$'th sub-problem needs not be solved (lines 4–5 in Algorithm 1). Similarly, if $g^1_1 = 0$ (which is common in our instances) then $\mathbf{u}^1 = (g^1_1, g^{\text{ub}}_2)$ (see (6)), which implies $\mathbf{u}^1 = \tilde{\mathbf{u}}^1 = (g^1_1, g^{\text{ub}}_2)$, and $g^1_1 = 0$, which implies $\tilde{\mathbf{u}}^1 = (0, g^{\text{ub}}_2)$. Now $R(\mathbf{0}, \tilde{\mathbf{u}}^1)$ is actually a line segment between $\tilde{\mathbf{u}}^1$ and the origin. Hence, the only region that may contain projections of NDPs not identified by

[10] When $u^k_2 = 0$, the kth sub-problem need not be solved.

solving the second sub-problem is $\{\hat{\mathbf{g}} \in \mathbb{R}^2_+ \mid g^1_2 \leq \hat{g}_2 \leq g^{\text{ub}}_2, \hat{g}_1 = 0\}$. Any point with projection on this line segment is, however, at least weakly dominated by $\mathbf{g}^1$. Consequently, there is no need to solve the first sub-problem (lines 9–11 in Algorithm 1).

5 Computational experiments, and conclusion

This section provides an overview of the input data, the computational set-up, and the experiment design. We present the different performance indicators used to benchmark P-QSM with QSM. Further, we investigate the sensitivity of the performance of algorithms by varying the discretization parameter δ_1 and the threshold value $\zeta \in \{0.70, 0.75, 0.8\}$.

All algorithms are written using Spyder IDE for Windows system(s) with a 1.70 GHz processor, 16 GB RAM, and 4 cores. We use `Python 3.7` as the interpreter and `Gurobi 9` as the solver. The sub-problems are simulated and solved on parallel identical computers in a cluster. Each computer solves a sub-problem and its full processor capacity and all CPU cores are available to `Gurobi` when solving a sub-problem. For each scalarized problem we use two stopping criteria: (a) MIP duality gap of 10^{-4}; (b) a time limit of 1500 seconds (for sensitivity analyses it is increased to 2500 seconds); hence, the solutions found are approximately efficient. Further, the discretization parameter δ_1 is selected as either 0.1 or 0.05, while $\delta_2 = 1$, yielding two different representations of the Pareto front for each algorithm.

5.1 Industrial instances

We use data provided by GKN Aerospace for most of the parameters and sets indicated in Table 1. However, the processing times p^{ℓ}_{jk} and qualification cost parameters β^{ℓ}_{jk} for product type $\ell \in \mathcal{L}$, operation $j \in \mathcal{J}_\ell$ in machine $k \in \mathcal{N}^{\ell}_j$ (qualification required) were not available. We have uploaded fifteen public instances along with details about the distributions used to generate each instance at https://bit.ly/3B4uLes. The demand is from quarterly forecasts provided by GKN Aerospace in January 2015 for the period 2016–2017. There are $L = 85$ part types, $|\mathcal{M}| = 517$ tasks, and $|\mathcal{K}| = 125$ machines. The minimum, maximum, and median values of the demand are 1, 149, and 12, respectively (we consider only non-zero values for the calculation of minimum, maximum, and median). For the processing times p^{ℓ}_{jk}, $k \in \mathcal{K}^{\ell}_j$, $j \in \mathcal{J}_\ell$, $\ell \in \mathcal{L}$ the minimum, maximum, and median are 0.1, 80, and 5.0 hours, respectively. Each machine has a yearly capacity of 5000 hours which can be equally divided among four quarters in a year. The planning period of two years with quarterly time buckets yields $T = 8$ time periods. The parameter values $\tau = 3$, $\gamma = 4$, and $\zeta = 0.70$ (although for sensitivity analyses we also investigate for $\zeta \in \{0.75, 0.80\}$) apply to all fifteen instances. The limiting factor for ordering raw materials is $d = 1.5$. The initial values and upper limits on inventories are $\bar{r}^{\ell}_{j0} = 0$ and $\bar{r}^{\ell}_{jt} = \lceil 0.5\, a^t_\ell \rceil$, $t \in \mathcal{T}$, respectively, $j \in \mathcal{J}_\ell$, $\ell \in \mathcal{L}$.

5.2 Design of experiments

To assess the performance of the two algorithms QSM and P-QSM we use solution times[11] and two different *representation measures*. Due to the discretization of g_1 and the limited solution time set for the scalarized problems, only a subset of the set of (approximately) efficient solutions is known, and the exact Pareto front is not available. Therefore, we need to investigate the quality of representation of the Pareto front obtained by each of the two methods. We denote by B^i_{eff} the set of efficient solutions to the ith instance, while the *representative set of approximately efficient solutions* to the ith instance obtained by an algorithm $j \in \mathcal{A} := \{\text{QSM}, \text{P-QSM}\}$ is denoted as $B^i_{\widetilde{\text{eff}}}(j)$. We then denote by $B^i_{\widetilde{\text{eff}}}(\mathcal{A})$ the set of solutions that are *efficient*[12] with respect to the set $\bigcup_{j\in\mathcal{A}} B^i_{\widetilde{\text{eff}}}(j)$. Then, for the ith instance and algorithm $j \in \mathcal{A}$, we define the cardinality (of NDPs) ratio measure[13] $\mu^i(j) := \left|B^i_{\widetilde{\text{eff}}}(\mathcal{A}) \bigcap B^i_{\widetilde{\text{eff}}}(j)\right| / \left|B^i_{\widetilde{\text{eff}}}(\mathcal{A})\right|$.

5.3 Results

In Table 3 the columns #NDPs refer to $\left|B^i_{\widetilde{\text{eff}}}(\mathcal{A}) \bigcap B^i_{\widetilde{\text{eff}}}(j)\right|$ for the respective instance i and algorithm j. With the discretization $\delta_1 = 0.1$, P-QSM found all the NDPs in $B^i_{\widetilde{\text{eff}}}(\mathcal{A})$ for nine out of the fifteen instances, i.e. $\mu^i(\text{P-QSM}) = 1$; for the other six instances, it found more (or equal) NDPs than QSM, i.e. $\mu^i(\text{P-QSM}) \geq \mu^i(\text{QSM})$. For thirteen out of the fifteen instances P-QSM had a significantly shorter computing time than QSM; only for instances 7 and 12 the solution time of P-QSM was longer than that of QSM, but then P-QSM identified two and seven additional NDPs, respectively. With $\delta_1 = 0.05$ we obtained, as expected, more NDPs. Fourteen out of the fifteen instances (i.e. all for except instance 10) satisfy $\mu^i(\text{P-QSM}) \geq \mu^i(\text{QSM})$. For twelve out of the fifteen instances, the solution time of P-QSM was shorter than that of QSM.

Visualization. Pareto fronts of TOILP problems are much more complicated to visualize than for BOILP problems. A scatter plot matrix is a simple technique adept at representing a pair-wise relationship of objectives. The drawback is, however, that one has to study three separate plots. Heatmap plots (as shown in Fig. 6) represent objective values by different shades of grey. In a heatmap each column corresponds to a NDP (there are 19 NDPs in Fig. 6), and each row corresponds to an objective function to which an intensity of greyness is assigned that reflects its value. The lighter the grey-shade, the higher the value. To define the color gradient we

[11] The solution time of P-QSM is measured as the sum of the time used to obtain $B^{g_3}_{\text{eff}}(0)$ (solve TRAP) and the maximum of the solution times for the sub-problems, which are (simulated to be) solved in parallel.

[12] $B^i_{\widetilde{\text{eff}}}(\mathcal{A})$ is the best known representation of the set B^i_{eff}, which is approximated.

[13] Here, the numerator equals the number of approximately efficient solutions obtained by algorithm $j \in \mathcal{A}$, while the denominator equals the number of (non-dominated) approximately efficient solutions identified by either of the algorithms in $\mathcal{A}$.

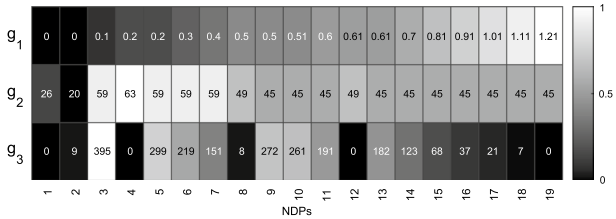


Fig. 6 Heatmap for the instance $i = 14$, algorithm j = P-QSM, $\delta_1 = 0.1$, $\delta_2 = 1$, and #NDPs = 19

normalized each objective function's values. From the heatmap the decision makers can identify some interesting NDPs, for instance, the NDPs 4 and 5 have (almost) the same values for the first (second) objective function, but for the third objective function the former is significantly better, by 299 units.

5.4 Sensitivity analysis

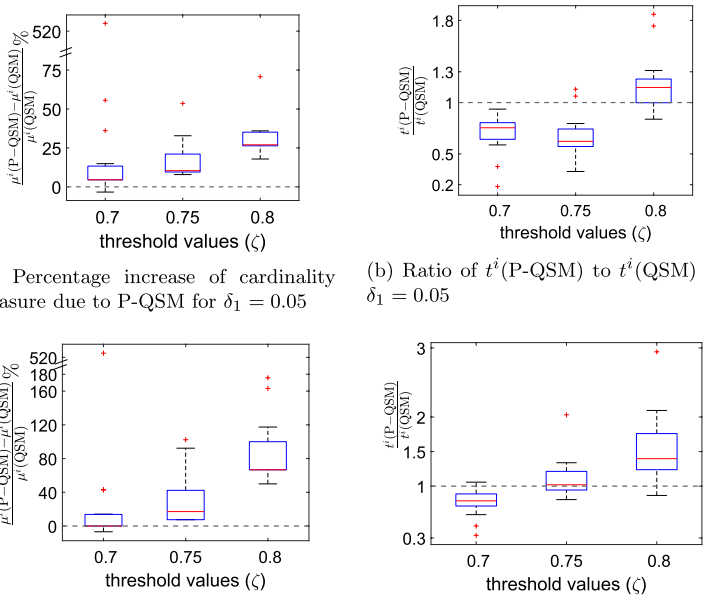
We perform sensitivity analyses to investigate the effect of varying the threshold (ζ) and discretization (δ_1) parameters on the solution times and representation measures, such as cardinality and *coverage gap* (Ceyhan et al. 2019, Def. 6) for P-QSM and QSM. In these experiments, we increase the time limit to 2500 seconds per scalarized problem.

Table 3 Comparison of QSM and P-QSM for $\zeta = 0.70$ and two discretizations. The time $t^i(j)$ is in seconds

	Discretization: $\delta_1 = 0.1$; $\delta_2 = 1$						Discretization: $\delta_1 = 0.05$; $\delta_2 = 1$					
	j = QSM			j = P-QSM			j = QSM			j = P-QSM		
i	#NDPs	$\mu^i(j)$	$t^i(j)$	#NDPs	$\mu^i(j)$	$t^i(j)$	#NDPs	$\mu^i(j)$	$t^i(j)$	#NDPs	$\mu^i(j)$	$t^i(j)$
1	15	0.88	4185	15	0.88	2587	25	0.89	6833	26	0.93	4682
2	13	0.76	4737	17	1.00	3159	25	0.89	6804	26	0.93	5003
3	15	0.88	4007	17	1.00	3089	25	0.89	7257	26	0.93	6395
4	25	0.51	10744	28	0.55	8885	44	0.47	27763	50	0.54	16566
5	15	0.88	5430	16	0.94	3827	23	0.82	11856	25	0.89	3778
6	6	1.00	7610	6	1.00	1856	6	1.00	5100	6	1.00	2539
7	15	0.88	2373	17	1.00	5562	25	0.89	7482	26	0.93	4772
8	15	0.80	6330	17	1.00	4684	25	0.89	11547	26	0.93	9039
9	15	0.88	3139	17	1.00	2456	25	0.89	3428	26	0.93	4804
10	14	0.87	7418	15	0.93	5849	26	0.90	12930	25	0.86	8599
11	15	0.88	3448	17	1.00	3112	25	0.89	6531	26	0.93	6660
12	20	0.72	4640	27	0.96	5673	34	0.72	7748	46	0.98	6916
13	18	0.72	6650	20	0.80	2861	24	0.54	24296	37	0.84	5551
14	15	0.79	3881	19	1.00	2615	26	0.86	4907	27	0.90	2858
15	14	0.87	4123	16	1.00	4001	26	0.90	6919	27	0.93	6944

In Fig. 7a and b we employ the discretization $\delta_1 = 0.05$. Figure 7a presents the distribution (over all fifteen instances i) of percentage increase in the cardinality ratio measure using P-QSM as opposed to QSM, i.e. $\bar{\mu}^i(\text{P-QSM}) := \frac{\mu^i(\text{P-QSM})-\mu^i(\text{QSM})}{\mu^i(\text{QSM})} \cdot 100\%$; for the three values of the loading threshold for machines (ζ), there is an increased cardinality ratio (values above 0%) using P-QSM. For $\zeta = 0.70$ ($\zeta = 0.75, \zeta = 0.8$) the median is 4.49% (10.38%, 27%). Figure 7b illustrates the effect on solution time of using P-QSM. For the threshold, $\zeta = 0.70$ all the instances require less time when P-QSM is applied (values < 1). For $\zeta = 0.8$, P-QSM requires more computational time than QSM for several instances. It should be noted that–-as indicated in Fig. 7a—P-QSM identifies more NDPs than QSM for $\zeta = 0.8$. For $\zeta = 0.75$ the ratio of the solution time is < 1 for most of the instances. For $\zeta = 0.70$ ($\zeta = 0.75, \zeta = 0.8$) the median values of the ratio of solution time of P-QSM to QSM is 0.75 (0.625, 1.15). Figure 8a illustrates the difference of solution times for QSM and P-QSM; the median value of the difference is 1410 (1351, −207) seconds for the threshold 0.70 (0.75, 0.8).

Figures 7c and 7d employ the discretization $\delta_1 = 0.1$, $\delta_2 = 1$. For $\zeta = 0.70$ ($\zeta = 0.75, \zeta = 0.80$) the median value of the ratio of solution time of P-QSM to QSM is 0.787 (1.02, 1.408). However, as indicated in Fig. 7c, with an increasing threshold, the number of NDPs found by P-QSM is significantly larger than by the QSM. For $\zeta = 0.70$ ($\zeta = 0.75, \zeta = 0.80$) the median value of the cardinality ratio is



(a) Percentage increase of cardinality measure due to P-QSM for $\delta_1 = 0.05$

(b) Ratio of t^i(P-QSM) to t^i(QSM) for $\delta_1 = 0.05$

(c) Percentage increase of cardinality measure due to P-QSM for $\delta_1 = 0.1$

(d) Ratio of t^i(P-QSM) to t^i(QSM) for $\delta_1 = 0.1$

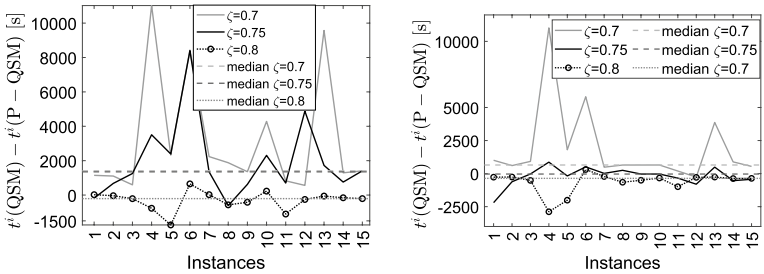
Fig. 7 Sensitivity plots: distributions of two performance indicators over the fifteen instances, separated for two levels of the discretization parameter δ_1, and grouped by three values of the threshold ζ. A red plus sign denotes an outlier outside 1.5 times the inter-quartile range

Table 4 Summary of sensitivity analyses

Cardinality ratio	Coverage gap ratio			
	$\delta_1 = 0.1$		$\delta_1 = 0.05$	
	$\bar{\alpha}^i$(P-QSM) > 0	$\bar{\alpha}^i$(P-QSM) ≤ 0	$\bar{\alpha}^i$(P-QSM) > 0	$\bar{\alpha}^i$(P-QSM) ≤ 0
$\bar{\mu}^i$(P-QSM) > 0	30	6	35	9
$\bar{\mu}^i$(P-QSM) ≤ 0	8	1	1	0

0% (17%, 66.66%). The median values for the difference (in seconds) of computing times of QSM and P-QSM (see Fig. 8b) are 650, −33, and −360 seconds for thresholds 0.70, 0.75, and 0.8, respectively.

It is known (Ceyhan et al. 2019) that finding more NDPs does not always result in the corresponding representation becoming significantly better. Hence, *coverage gap* values can be used. We adjust (Ceyhan et al. 2019, Def. 6) for minimization problems and normalize the three objectives, as they have different dimensions, to calculate the coverage gap values. The coverage gap, $\alpha^i(j)$, of algorithm j for instance i, measures the distance to the worst represented point in the set $B^i_{\widetilde{\text{eff}}}(\mathcal{A})$ (i.e. the best known representation of the efficient set B^i_{eff}) from the obtained representative set of approximately efficient solutions, i.e. $B^i_{\widetilde{\text{eff}}}(j)$. Figures 9a–9b present on the right axis the ratio of the difference of the coverage gap values of QSM and P-QSM to the coverage gap value of QSM (i.e. $\bar{\alpha}^i(\text{P-QSM}) := \frac{\alpha^i(\text{QSM}) - \alpha^i(\text{P-QSM})}{\alpha^i(\text{QSM})}$), referred to as *coverage gap ratio*. As it is desired to have a lower coverage gap value; for P-QSM to outperform QSM, the ratio needs to be greater than zero. Table 4 summarizes the results from the fifteen instances and over all three threshold values. Hence, there are 45 values reported for both $\delta_1 = 0.05$ and $\delta_1 = 0.1$. For $\delta_1 = 0.1$, there are 30 occurrences when both the cardinality ratio and coverage gap ratio are positive. In only one occurrence both of them are less than or equal to zero. The results for $\delta_1 = 0.05$ are analogously presented. There is no occurrence where both coverage and cardinality ratios are less than or equal to zero.



(a) $\delta_1 = 0.05$ (b) $\delta_1 = 0.1$

Fig. 8 Difference of solution time for QSM and P-QSM [s] for threshold values $\zeta \in \{0.70, 0.75, 0.80\}$

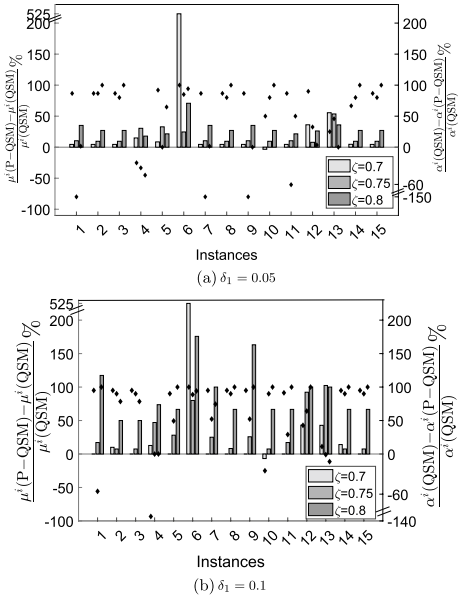


Fig. 9 Sensitivity plots. Left axis: grey-bars with corresponding cardinality ratio. Right axis: black-rhombuses representing coverage gap ratios. Each group of three bars corresponds to the threshold values 0.7, 0.75, and 0.8

Discussion: The reduction in solution time can be attributed to the parallelization used in P-QSM, although there is a significant improvement in the coverage gap that is not due to the parallelization. In most of our computations, the time limit of 2500 seconds per scalarized problem is not reached, but still, the approximation of the Pareto front provided by QSM is much inferior to that of P-QSM in almost all of the instances (over variation of threshold and discretization parameters). As an explanation, consider the following example, which is illustrated in Fig. 10. Assume there are four NDPs and three of them, $\mathbf{g}^1, \mathbf{g}^2$, and $\mathbf{g}^3$, correspond to the efficient solutions in $B^{g_3}_{\text{eff}}(0)$, i.e. $g^1_3 = g^2_3 = g^3_3 = 0$. In P-QSM, four sub-problems are initialized corresponding to the upper bounds $\tilde{\mathbf{u}}^1, \tilde{\mathbf{u}}^2, \tilde{\mathbf{u}}^3$, and $\tilde{\mathbf{u}}^4$. QSM, however, starts with the upper bound $\tilde{\mathbf{l}}^1$. The first application of the two-stage scalarizations will result in one of the NDPs corresponding to $B^{g_3}_{\text{eff}}(0)$, w.l.o.g. assume the NDP $\mathbf{g}^3$ is identified.[14] For $\delta_1 = 0.5$, the subsequent upper bounds will be $\tilde{\mathbf{l}}^2$ and $\tilde{\mathbf{l}}^3$. Due to the discretization, the NDPs $\mathbf{g}^2$ and $\mathbf{g}^4$ will be overlooked by the QSM, but not by the P-QSM as both points are identified while solving the second sub-problem, i.e. for $R(\mathbf{0}, \tilde{\mathbf{u}}^2)$. Hence, both the parallelization and the solving of the bi-objective TRAP first give P-QSM a computational advantage over QSM.

[14] Either of $\mathbf{g}^1$, $\mathbf{g}^2$, and $\mathbf{g}^3$ can be identified, as all of them are in $B^{g_3}_{\text{eff}}(0)$.

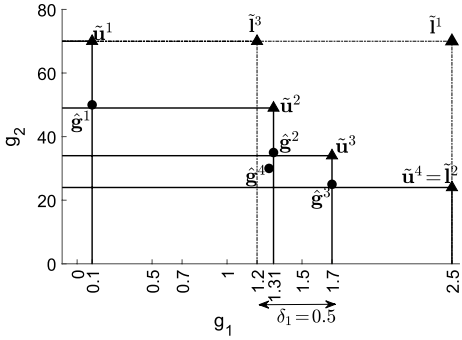


Fig. 10 Illustration of the reason for P-QSM yielding a better representation. For QSM and P-QSM the upper bounds used are $\{\tilde{\mathbf{l}}^1, \tilde{\mathbf{l}}^2, \tilde{\mathbf{l}}^3\}$ and $\{\tilde{\mathbf{u}}^1, \tilde{\mathbf{u}}^2, \tilde{\mathbf{u}}^3, \tilde{\mathbf{u}}^4\}$, respectively

5.5 Conclusion

We present a modified version, P-QSM, of the *quadrant shrinking method* (QSM) for identifying a *representative set of approximately efficient solutions* to a generalized tri-objective tactical resource allocation problem. The computational benefits of the P-QSM are due to solving sub-problems in parallel/simultaneously on different computers and to initiating the algorithm with a subset of the NDPs. For most of the instances, the P-QSM identifies more NDPs than the QSM; it also results in a smaller coverage gap. Future work may consider providing tighter formulations of the scalarization of the sub-problems in P-QSM.

Appendix

Proof of Proposition 11 We will first show that $\mathbf{g}(B_{\text{eff}}) \subseteq \bigcup_{k=1}^{K+1} H^k$. Consider an efficient solution $(\mathbf{y}^*, \mathbf{r}^*) \in B_{\text{eff}}$ and the corresponding objective vector $\mathbf{g}(\mathbf{y}^*, \mathbf{r}^*) \in \mathbf{g}(B_{\text{eff}})$. Assume the contradiction, that $\mathbf{g}(\mathbf{y}^*, \mathbf{r}^*) \notin \bigcup_{k=1}^{|B_{\text{eff}}^{g_3}(0)|+1} H^k$. We know from Prop. 10 that the projection of any NDP onto the (g_1, g_2)-criterion space lies in $N(B_{\text{eff}}^{g_3}(0))$, and by the construction of upper bounds, it holds that $N(B_{\text{eff}}^{g_3}(0)) \subseteq \bigcup_{k=1}^{|B_{\text{eff}}^{g_3}(0)|+1} R(\mathbf{0}, \mathbf{u}^k)$. This implies $\mathbf{g}(\mathbf{y}^*, \mathbf{r}^*)$ has a projection $\hat{\mathbf{g}}(\mathbf{y}^*, \mathbf{r}^*) = (g_1(\mathbf{y}^*), g_2(\mathbf{y}^*))^\top \in R(\mathbf{0}, \mathbf{u}^k)$ for at least one sub-problem k. Note that $\mathbf{g}(\mathbf{y}^*, \mathbf{r}^*)$ is a NDP corresponding to a feasible set B, which implies that it must be a NDP for any (feasible) subset $B^{\hat{\mathbf{g}}}(\mathbf{u}^k) := \{\, (\mathbf{y}, \mathbf{r}) \in B \mid g_1(\mathbf{y}) \le u_1^k,\ g_2(\mathbf{y}) \le u_2^k \,\} \subseteq B$. Since applying the QSM algorithm to the feasible set $B^{\hat{\mathbf{g}}}(\mathbf{u}^k)$ results in all the NDPs (denoted as H^k), it must identify $\mathbf{g}(\mathbf{y}^*, \mathbf{r}^*)$ as well. Hence, we have a contradiction, i.e. $\mathbf{g}(\mathbf{y}^*, \mathbf{r}^*) \in \bigcup_{k=1}^{|B_{\text{eff}}^{g_3}(0)|+1} H^k$.

Next, we show that $\bigcup_{k=1}^{K+1} H^k \subseteq \mathbf{g}(B_{\text{eff}})$. We assume the converse, i.e. $\mathbf{g}^* \notin \mathbf{g}(B_{\text{eff}})$ and $\mathbf{g}^* \in H^k$ (for one of the sub-problems k). Consequently, $\exists\, \bar{\mathbf{g}} \in \mathbf{g}(B_{\text{eff}})$ such that $\bar{\mathbf{g}} \le \mathbf{g}^*$. However, by the property of the QSM we know $\mathbf{g}^*$ is a NDP of the GTRAP with additional constraints $g_i(\mathbf{y}) \le u_i^k$, $i = 1, 2$. Hence, at least one of the constraints

$\bar{g}_1 > u_1^k$ and $\bar{g}_2 > u_2^k$ must hold. Consequently, at least one of the constraints $g_1^* \leq u_1^k < \bar{g}_1$ and $g_2^* \leq u_2^k < \bar{g}_2$ holds, which implies that $\bar{\mathbf{g}}$ does not dominate $\mathbf{g}^*$. The proposition follows by contradiction.

Proof of Proposition 12 Let $K := |B_{\text{eff}}^{g_3}(0)|$. The projections onto the two-dimensional criterion space of all the NDPs are in the set $\bigcup_{k=1}^{K+1} R(\mathbf{0}, \mathbf{u}^k)$ (as stated in Prop. 11). If we prove $A := \left\{ \bigcup_{k=1}^{K+1} R(\mathbf{0}, \mathbf{u}^k) \setminus \bigcup_{k=1}^{K+1} R(\mathbf{0}, \tilde{\mathbf{u}}^k) \right\} \bigcap \hat{\mathbf{g}}(B_{\text{eff}}) = \emptyset$ (see Table 2 for notations) then the proposition follows. Let us assume a feasible solution $(\bar{\mathbf{y}}, \bar{\mathbf{r}}) \in B_{\text{eff}}$ and $(g_1(\bar{\mathbf{y}}), g_2(\bar{\mathbf{y}})) \in A$. This implies the inequalities $\tilde{u}_2^k < g_2(\bar{\mathbf{y}}) \leq u_2^k$ for at least one $k \in \{2, \dots, K+1\}$ ($k = 1$ is not included since $\mathbf{u}^1 = \tilde{\mathbf{u}}^1$). Since g_2 is integer-valued function and $\tilde{u}_2^k = u_2^k - 1$, it follows that $g_2(\bar{\mathbf{y}}) = u_2^k$. Moreover, the inequalities $0 \leq g_1(\bar{\mathbf{y}}) \leq u_1^k$ hold. Let us first consider the case when $0 \leq g_1(\bar{\mathbf{y}}) < u_1^{k-1}$. This implies that $g_2(\bar{\mathbf{y}}) = u_2^k < u_2^{k-1}$ (the latter strict inequality follows from (6)). This can be equivalently stated as $g_2(\bar{\mathbf{y}}) < \tilde{u}_2^{k-1} + 1 \iff g_2(\bar{\mathbf{y}}) \leq \tilde{u}_2^{k-1}$. Hence, it holds that $(g_1(\bar{\mathbf{y}}), g_2(\bar{\mathbf{y}})) \in R(\mathbf{0}, \tilde{\mathbf{u}}^{k-1})$ which, from the assumption that $(g_1(\bar{\mathbf{y}}), g_2(\bar{\mathbf{y}})) \in A$, is obviously a contradiction.

Hence, we consider the case when $g_1(\bar{\mathbf{y}}) \in [u_1^{k-1}, u_1^k]$. From (6) we know that $u_2^k = g_2^{k-1}$, $k = 2, \dots, K+1$, such that $g_2(\bar{\mathbf{y}}) = u_2^k = g_2^{k-1}$ holds. Subsequently, from (6) we know that $u_1^k = g_1^k$, $k = 2, \dots, K$, such that $u_1^{k-1} = g_1^{k-1}$ holds. Thus, we have that the inequalities $g_1^{k-1} \leq g_1(\bar{\mathbf{y}}) \leq u_1^k$ hold. Therefore, it holds that $g_1^{k-1} \leq g_1(\bar{\mathbf{y}})$, $g_2^{k-1} = g_2(\bar{\mathbf{y}})$, and by definition $g_3^{k-1} = 0$. Moreover, since $(g_1^{k-1}, g_2^{k-1}) \in R(\mathbf{0}, \tilde{\mathbf{u}}^{k-1})$ we have that $\mathbf{g}^{k-1} \neq \mathbf{g}(\bar{\mathbf{y}})$, because $\mathbf{g}(\bar{\mathbf{y}}) \in A$. Consequently, $\mathbf{g}^{k-1}$ must dominate $\mathbf{g}(\bar{\mathbf{y}})$, which is a contradiction as the latter is assumed to be a NDP. □

Funding Open access funding provided by Chalmers University of Technology.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Bixby R, Cook W, Cox A, Lee E (1999) Computational experience with parallel mixed integer programming in a distributed environment. Annal Oper Res 90:19–43. https://doi.org/cg7gh9

Boland N, Charkhgard H, Savelsbergh M (2015) The L-shape search method for triobjective integer programming. Math Prog Comput 8(2):217–251. https://doi.org/hbtm

Boland N, Charkhgard H, Savelsbergh M (2017) The quadrant shrinking method: a simple and efficient algorithm for solving tri-objective integer programs. Eur J Oper Res 260(3):873–885. https://doi.org/ftpq

Bradley JR, Glynn PW (2002) Managing capacity and inventory jointly in manufacturing systems. Manag Sci 48(2):273–288. https://doi.org/bm3pr5

Ceyhan G, Köksalan M and Lokman B (2019) Finding a representative nondominated set for multi-objective mixed integer programs. Eur J Oper Res 272(1): 61–77. https://doi.org/jccf

Chen SP and Huang WL (2010) A membership function approach for aggregate production planning problems in fuzzy environments. Int J Prod Res 48: 7003–7023. https://doi.org/chbpvb

Dhaenens C, Lemesre J, Melab N, Mezmaz MS and Talbi EG (2006) Parallel exact methods for multiobjective combinatorial optimization. Wiley Series on Parallel and Distributed Computing, 187–210. Wiley https://doi.org/dw97rm

Díaz-Madroñero M, Mula J and Peidro D (2014) A review of discrete-time optimization models for tactical production planning. Int J Prod Res 52(17): 5171–5205. https://doi.org/ftpt

Dächert K and Klamroth K (2014) A linear bound on the number of scalarizations needed to solve discrete tricriteria optimization problems. J Glob Optimiz 61(4): 643–676. https://doi.org/f659q7

Dächert K, Klamroth K, Lacour R and Vanderpooten D (2017) Efficient computation of the search region in multi-objective optimization. Eur J Oper Res 260(3): 841–855. https://doi.org/gbg9

Fotedar S, Strömberg A-B, Almgren T (2022) Bi-objective optimization of the tactical allocation of job types to machines: mathematical modeling, theoretical analysis, and numerical tests. Int Trans Oper Res. https://doi.org/10.1111/itor.13180

Genin P, Lamouri S and Thomas A (2008) Multi-facilities tactical planning robustness with experimental design. Prod Plan Control 19(2): 171–182. https://doi.org/dgfdxd

Kirlik G and Sayın S (2014) A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems. Eur J Oper Res 232(3): 479–488. https://doi.org/ftpr

Klamroth K, Lacour R and Vanderpooten D (2015) On the representation of the search region in multi-objective optimization. Eur J Oper Res 245(3): 767–778. https://doi.org/h2w6

Lan Y, Zhao R and Tang W (2011) Minimum risk criterion for uncertain production planning problems. Comput Ind Eng 61(3): 591–599. https://doi.org/b5ffz9

Laumanns M, Thiele L and Zitzler E (2006) An efficient, adaptive parameter variation scheme for metaheuristics based on the epsilon-constraint method. Eur J Oper Res 169(3): 932–942. https://doi.org/dn45j3

Lemesre J, Dhaenens C and Talbi E (2007) An exact parallel method for a bi-objective permutation flowshop problem. Eur J Oper Res 177(3): 1641–1655. https://doi.org/d7xhgx

Mieghem JAV (2003) Capacity management, investment, and hedging: Review and recent developments. Manuf Serv Oper Manag 5(4): 269–302. https://doi.org/dhx6pz

Nourelfath M (2011) Service level robustness in stochastic production planning under random machine breakdowns. Eur J Oper Res 212(1): 81–88. https://doi.org/chbswp

Özlen M,Burton BA and MacRae CAG (2013) Multi-objective integer programming: an improved recursive algorithm. J Optimiz Theory Appl 160(2): 470–482. https://doi.org/f5vxmg

Pochet Y, Wolsey LA (2006) Production Planning by Mixed Integer Programming. Springer, New York

Talbi EG, Mostaghim S, Okabe T, Ishibuchi H, Rudolph G and Coello Coello CA 2008. Parallel Approaches for Multiobjective Optimization, pp. 349–372. Berlin, Heidelberg: Springer Berlin Heidelberg

Ulungu EL and Teghem J (1994) Multi-objective combinatorial optimization problems: a survey. J Multi-Criteria Decis Anal 3(2): 83–104. https://doi.org/b2g9vz

Wei C, Li Y and Cai X (2011) Robust optimal policies of production and inventory with uncertain returns and demand. Int J Prod Econ 134(2): 357–367. https://doi.org/bv67qg

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.