Expert Systems with Applications 42 (2015) 3852–3867

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Composite goal methods for transportation network optimization

Marco Veluscek [a,*], Tatiana Kalganova [a], Peter Broomhead [a], Anthony Grichnik [b]

[a] Electronic and Computer Engineering, School of Engineering and Design, Brunel University, Kingston Lane, Uxbridge UB8 3PH, United Kingdom
[b] Innovation Leader – Intelligent Systems, Logistics Research & Innovation, Caterpillar Logistics Inc., 100 NE Adams St, Peoria, IL, United States

A R T I C L E I N F O

Article history:
Available online 19 December 2014

Keywords:
Transportation network optimization
Ant colony optimization
Multi-goal methods
Logistics optimization

A B S T R A C T

Lately the topic of multi-objective transportation network optimization has received increased attention in the research literature. The use of multi-objective transportation network optimization has led to a more accurate and realistic solution in comparison to scenarios where only a single objective is considered. The aim of this work is to identify the most promising multi-objective optimization technique for use in solving real-world transportation network optimization problems. We start by reviewing the state of the art in multi-objective optimization and identify four generic strategies, which are referred to as *goal synthesis, superposition, incremental solving* and *exploration*. We then implement and test seven instances of these four strategies. From the literature, the preferred approach lies in the combination of goals into a single optimization model (a.k.a. goal synthesis). Despite its popularity as a multi-objective optimization method and in the context of our problem domain, the experimental results achieved by this method resulted in poor quality solutions when compared to the other strategies. This was particularly noticeable in the case of the superposition method which significantly outperformed goal synthesis.

1. Introduction

When optimizing transportation networks, several criteria can be used as the optimization goal, criteria such as the shortest distance traveled minimum inventory, minimum transportation cost and highest network resilience. In the case of industry based applications, it is often advantageous to simultaneously consider several of these goals with a view to developing a model that more accurately represents the operation of the actual business. Defining a mathematical model that incorporates the perspective of more than one criterion in itself is not a simple task and often involves the definition of complex non-linear models. Moreover, the goals of such criteria may well be mutually exclusive and result in the definition of a multi-goal model that is not or not always achievable in practice.

A simple way to handle the multi-objective optimization problem is to construct a composite objective function that is the weighted sum of the conflicting objectives (Aslam & Ng, 2010). In the literature this technique is also referred to as the *preference-based strategy* and is the approach most often adopted by academic studies. The preference-based strategy is a trade-off that reduces a multi-goal approach to a single-goal optimization problem. However, in reality as a solution this trade-off has proved to be very sensitive to the relative preferences assigned to the goals (Aslam & Ng, 2010) and in practice it is difficult for practitioners, even those familiar with the problem domain to precisely and accurately select such weightings (Konak, Coit, & Smith, 2006).

As part of this work, we identify the principal alternative methods for use in multi-objective optimization when applied to the solution of real-world transportation network optimization problems. The work reported here is an extension to previously work Veluscek et al. (2014). The problem models and the data sets have been defined in collaboration with a world leading manufacturer of construction and mining equipment and represent a snapshot of the day-to-day complexities and operational challenges faced by our industrial partners business. The aim of this work is to identify and test those multi-objective optimization techniques that better address the complexities of such operating environments.

In the following sections, we identify four generic strategies used to optimize multi-goal problem scenarios and formally present seven implementations of these strategies. The methods have been designed and implemented with a view to solving the transportation network optimization problem reported in Veluscek et al. (2014).

In Sections 2 and 3 we present the background to this work and introduce previously work on multi-goal optimization. In Section 4 we formally describe the methods used to combine single-goal

* Corresponding author.
E-mail addresses: marco.veluscek@brunel.ac.uk (M. Veluscek), tatiana.kalganova@brunel.ac.uk (T. Kalganova), peter.broomhead@brunel.ac.uk (P. Broomhead), grichnik_anthony_j@cat.com (A. Grichnik).

http://dx.doi.org/10.1016/j.eswa.2014.12.017
0957-4174/

optimization problems. In Section 5 we present the outcome of the numerical experiments undertaken to verify and test the effectiveness of the proposed methods.

2. Context and motivations

A robust solution to the multi-goal optimization problem is of particular interest to real-world applications where several optimization objectives are commonly involved. Multi-goal problems usually do not have a single 'best' solution, but are characterized by a set of solutions that are superior to others when considering all objectives (Alaya, Solnon, & Ghedira, 2007). This set is referred to as the Pareto set or as the non-dominated solution (Alaya et al., 2007). This multiplicity of solutions can be explained by the fact that individual objectives are often in conflict (Alaya et al., 2007). For example, Altiparmak, Gen, Lin, and Paksoy (2006) defined three objectives for the transportation network optimization problem: the total cost, the total satisfied customer demand and the equity of the capacity utilization ratio for each production source. The authors then implement a genetic algorithm to find the set of Pareto-optimal solutions. A similar example is presented in Yagmahan and Yenisey (2008) for the flow shop scheduling problem. The multi-objective function in this instance consists of minimizing the distance between the values of all the single-objective functions.

In our experience, most of the solutions proposed for multi-objective optimization problems are either specific to the kind of problem or to the kind of technique used to determine the optimal solution. We have identified four generic solution strategies that in general are used to solve multi-objective optimization problems.

The first strategy is called **Goal Synthesis** and requires the definition of a mathematical model which includes all the single-goal problems. This category is also referred to as the *preference-based strategy* (Aslam & Ng, 2010). The model defines one search space which is a sub-space of the intersection of the single-goal problem search spaces. The best composite solution is then sought on this space along one path. The solution found is feasible for each single-goal problem separately, but it is not necessarily the optimal one. Applying this strategy is no different from solving any other optimization problem: firstly a mathematical model is defined and then an optimal solution is sought using an appropriate optimization algorithm. However, there is no guarantee that the intersection of the single-goal problems exists or that the definition of such a multi-goal model is even possible.

The second strategy is called **Superposition** and in contrast to the previous method does not require the definition of a multi-objective problem model. Firstly, a solution is computed for each of the single-goal problems and then a combination of them are taken as the multi-goal solution. The applicability of this strategy relies on the definition of a combination operator. Again it is possible that the combination of the single-goal solutions is empty and a feasible solution does not exist. Das and Dennis (1998) designed a method based on this strategy to solve generic non-linear multi-objective optimization problems.

The third strategy is called **Incremental Solving**. Here each single-goal problem is solved sequentially in accordance with a predefined order, and the starting exploration point of the ith problem is the solution or stopping point of the $(i-1)$th problem. The solution for the multi-goal problem depends on the order used to solve the single-goal problems. Boudahri, Sari, Maliki, and Bennekrouf (2011) adopted this strategy to optimize an agricultural products supply chain.

The final strategy is called **Exploration** and is based on a 'brute force' approach. Firstly, a large number of feasible solutions are generated for each single-goal problem and then the multi-goal solution is taken as the solution that represents the 'best' compromise for the set of single-goal problems. Applying this strategy should always lead to a solution, provided a feasible solution exists for at least one of the single-goal problems. In common with many brute force approaches the cost of producing a quality solution is computational expensive. Bevilacqua et al. (2012) adopted this strategy to solve a generic distribution network and employed a genetic algorithm to improve the generation of solutions.

Aslam and Ng (2010) and Ogunbanwo et al. (2014) provide extensive reviews of the work undertaken for the problem of transportation network optimization. We have analyzed the works presented in such reviews and have categorized the reported methods with respect to those developed to solve multi-objective optimization problems. Table 1 and Fig. 1 show the results of that analysis. We can clearly see that in recent years the Goal Synthesis strategy is the dominant method used. Nevertheless, despite its popularity we will show that it may not necessarily be the best choice when solving real-world transportation network optimization problems.

As will be discussed in the following sections, the method used in this work to solve our specific real-world optimization problem is the Ant Colony System algorithm (Dorigo & Gambardella, 1997). García-Martínez, Cordón, and Herrera (2007) analyzed several ant colony optimization variants for multi-goal optimization and presented a taxonomy for them. The authors also performed an empirical analysis for the travel salesman problem and compared their results with two other well-known multi-objective genetic algorithms. It is worth noting that a prerequisite of such analysis is to define a multi-goal model to generate the Pareto optimal frontier. Once again, the authors proposed a model that simultaneously considers all optimization goals (i.e. goal synthesis). This indicates a preference for the goal synthesis strategy over the use of alternatives.

3. Transportation network optimization

A transportation network optimization problem may be express in terms of a minimization objective function, a set of variables and a set of constraints over these variables, regardless of the goal type (functions having to be maximized may be multiplied by -1). Given a vector of variables $x \in \mathbb{R}^n$ and a vector of cost coefficients $c \in \mathbb{R}^n$, a transportation network optimization problem may be defined as:

$$v = \min\left\{c^T x | Ax = b \wedge \ x \geqslant o\right\} \tag{1}$$

where $A \in \mathbb{R}^{m \times x}$ is a matrix of coefficients, $b \in \mathbb{R}^m$ is a vector of coefficients and $v \in \mathbb{R}^n$ is a vector of assignments for the variables x such that the value of the objective function $c^T x$ is minimum. The matrix A and the vector b define the constraints over the decision variables x and define the problem search space. Therefore, a transportation network optimization problem is defined by the tuple $lp := (x, c, A, b, v)$. A multi-goal optimization problem is a set of tuples representing single-goal optimization problems:

$$LP(x, A, b) = \left\{(x, c, A, b, v) | \exists c \in \mathbb{R}^{|x|} \wedge \exists v \in \mathbb{R}^{|x|}\right\}, \tag{2}$$

where the vector of variables $x \in \mathbb{R}^n$ and the set of coefficients A and b are the same for all the single-goal problems.

In a transportation network optimization problem, the variables x define the number of products to send on a given network route. The coefficients c usually depend on the goal and are typically information associated with a given route on the network (e.g. having to optimize for minimum transportation cost, $c_i \in c$ is the cost to send products via route i). Typically the constraints defined by A and b are the constraints placed on production capacity and customer demand. The solution v is a distribution plan for the network.

Table 1
Objectives investigated and strategies used in existing approaches for solving several multi-objective optimization problems in the area of operations research.

Author (year)	Multi-objective method	Description
Altiparmak et al. (2006)	GS	TNO for minimum transit time and minimum transportation costs
Bevilacqua et al. (2012)	EX	TNO for minimum transit time and minimum transportation costs
Boudahri et al. (2011)	IS	TNO for minimum traveled distance and minimum transportation costs
Cardona-Valdés, Álvarez, and Ozdemir (2011)	GS	TNO for minimum transit time and minimum transportation costs
Che and Chiang (2010)	GS	TNO for minimum transit time, minimum transportation costs, and maximum product quality
Che (2012)	GS	TNO for minimum transit time and minimum transportation costs
Chen, Yuan, and Lee (2007)	GS	TNO for minimum transit time and minimum transportation costs
Cintron, Ravindran, and Ventura (2010)	GS	TNO for minimum traveled distance, minimum transportation costs, maximum service level, and maximum product quality
Ding, Benyoucef, and Xie (2004)	GS	TNO for minimum transit time and minimum transportation costs
Ding, Benyoucef, and Xie (2009)	GS	TNO for maximum service level and minimum transportation costs
Ghoseiri and Nadjari (2010)	GS	TNO for minimum traveled distance and minimum transportation costs
Huang, Li, and Wang (2011)	GS	TNO for minimum transportation costs and maximum network resilience
Kamali, Fatemi Ghomi, and Jolai (2011)	GS	TNO for minimum transportation costs and maximum service level
Liang (2008)	GS	TNO for minimum traveled distance, minimum transit time, and minimum transportation costs
Lin and Wang (2008)	GS	TNO for minimum transit time, minimum transportation costs, and maximum service level
Sadjady and Davoudpour (2012)	GS	TNO for minimum traveled distance, minimum transportation costs, and minimum transit time
Utama, Djatna, Hambali, Marimin, and D. (2011)	GS	TNO for minimum traveled distance, minimum transportation costs, maximum service level, maximum product quality, and minimum environmental impact
Wang (2009)	GS	TNO for minimum transportation costs and maximum network resilience
Yeh and Chuang (2011)	GS	TNO for minimum transportation costs, minimum transit time, minimum environmental impact, and maximum product quality
Chen and Lee (2004)	GS	TNO for minimum transportation costs, maximum service level, and maximum network resilience
Sabri and Beamon (2000)	IS	TNO for minimum traveled distance, minimum transportation costs, maximum network resilience, maximum service level and maximum product quality
Joines, King, Kay, and Gupta (2002)	GS	TNO for minimum transportation costs and maximum service level
Wang, Lai, and Shi (2011)	GS	TNO for minimum transportation costs and minimum environmental impact
Torabi and Hassini (2008)	GS	TNO for minimum transportation costs and maximum product quality
Amid, Ghodsypour, and O'Brien (2011)	GS	Multi-goal supplier selection optimization
Wang, Huang, and Dismukes (2004)	GS + SP	Multi-goal supplier selection optimization. GS is used to have a MO model and SP to determine the weights
Weber and Current (1993)	GS	Multi-goal supplier selection optimization
Liu, Ding, and Lall (2000)	GS	Multi-goal supplier selection optimization with goal synthesis for combination of 23 goals/factors
Kumar, Vrat, and Shankar (2004)	GS	Multi-goal supplier selection optimization
Leung, Tsang, Ng, and Wu (2007)	GS	Trade-off between robustness and effectiveness of solution for multi-site production planning optimization problem
Yıldız (2009)	EX	Hybrid hill climbing optimization for manufacturing optimization with goals of minimizing the mass of the brake and minimizing the stopping time
Chaharsooghi and Meimand Kermani (2008)	GS + EX	Efficient multi-goal ACO for multi-objective resource allocation problem. GS is used to have a MO model and EX to efficiently explore the Pareto optimal frontier
McMullen and Tarasewich (2006)	GS + EX	Assembly line balancing optimization for the goals of crew size optimization, system utilization, jobs scheduling, and system design costs. GS is used to have a MO model and EX to efficiently explore the Pareto optimal frontier
Das and Dennis (1998)	SP	New multi-purpose method for generating the Pareto optimal points

TNO is *transportation network optimization*, MO is *multi-objective*, GS is *goal synthesis*, EX is *exploration*, IS is *incremental solving*, SP is *superposition*, and ACO is *ant colony optimization*.

4. Composite goal methods

We now present and describe seven different $LP(x, A, b)$ means to solve the multi-objective optimization problem. These methods are a formalization of the four generic strategies described above in Section 2.

Given a vector of variables $x \in \mathbb{R}^n$, a vector of coefficients $b \in \mathbb{R}^m$ and a coefficient matrix $A \in \mathbb{R}^{m \times n}$, let $S := (x, a, b,)$ be the tuple defining the problem search space. Recall from Eq. (2) that $LP(x, A, b)$ is the set of single-goal problems or the multi-goal optimization problem to be solved. The set LP is defined in Section 3. For simplicity, whenever there is no ambiguity, we take LP to be a synonym for $LP(x, A, b)$.

Let us define the projection operators on the search space S and on a given optimization problem $lp \in LP$ as $\pi_x(S) = x$, $\pi_A(S) = A$, $\pi_b(S) = b$, $\pi_c(lp) = c$ and $\pi_v(lp) = v$.

In order to improve readability, whenever there is no ambiguity, we write x, A and b instead of $\pi_x(S), \pi_A(S)$ and $\pi_b(S)$ respectively. Similarly, we write c^j and v^j instead of $\pi_c(lp_j)$ and $\pi_v(lp_j)$.

The proposed methods require a function to solve the optimization problem. Here we use the ant colony optimization algorithm described in Dorigo and Gambardella (1997). The methods defined below in Sections 4.1–4.6 are completely independent of this choice. The method defined in Section 4.7 is a specialization of the Ant Colony Optimization algorithm for solving multi-goal problems and it is used solely for the purpose of comparison.

Let ACS be the function representing the ant colony solver

$$\begin{aligned} ACS : LP &\to \mathbb{R}^n, \\ lp &\mapsto v, \end{aligned} \tag{3}$$

where $lp \in LP$ is the optimization problem to be solved and $v \in \mathbb{R}^n$ is a feasible solution to the problem lp. The specific details of how to find this solution are not relevant to the purpose of this work.

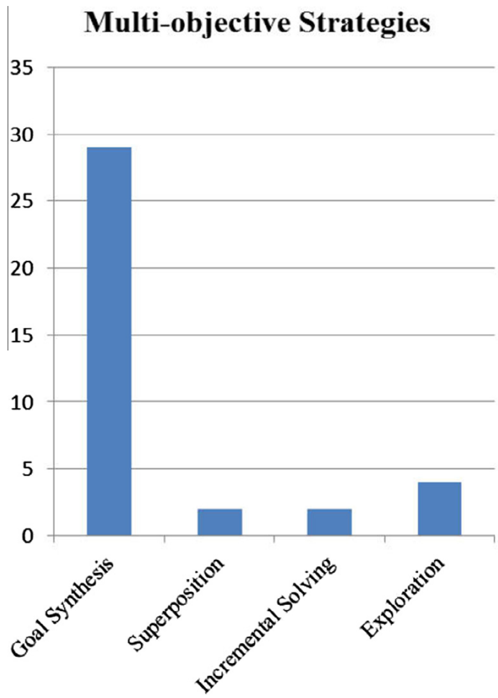

Fig. 1. Summary of the multi-objective strategies presented in Table 1 for solving multi-objective transportation network optimization problems.

In several of the methods described below, a reduction function is employed to narrow the problem search space, given a partial solution. Given an optimization problem $lp_j \in LP$, the reduction function is defined as:

$$\begin{aligned} red &: \mathbb{R}^n \rightarrow \mathbb{R}^m, \\ & v^j \mapsto b - A \cdot v^j. \end{aligned} \tag{4}$$

Application of the reduction function has the effect of reducing the production capacity and the customer demand by the amount of product already sent through the network.

The performance of a solution is the value of the objective function and is defined as the sum of the values of the vector v weighted by the cost coefficients c. Given an optimization problem $lp_j \in LP$, let p be the function that measures the performance of a solution:

$$\begin{aligned} p &: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}, \\ & v^j, c^j \mapsto \sum_{i=1}^{n} v_i^j \cdot c_i^j. \end{aligned} \tag{5}$$

4.1. Δ-Unification (DU)

The first method is based on the *goal synthesis* strategy. The first step consists of finding a solution for each single-goal problem. This provides an estimation of the optimal solution for each of the single-goal problem. We then define a new optimization problem, whose goal is to minimize the difference between the current solution and the worst performing single-goal solution. Let of_{diff} be the objective function of such a problem:

$$of_{diff}(x) = \min_{lp_j \in LP}(|p(v^j, c^j) - p(x, c^j)|/p(v^j, c^j)), \tag{6}$$

where x are the variables of the optimization problem.

The new optimization problem is defined as:

$$lp_{diff} := \min\{of_{diff}(x) | Ax = b \wedge x \geqslant 0\}. \tag{7}$$

Solving the optimization problem lp_{diff} causes the solver to walk through the solution space along the intersection of the solution surfaces. Algorithm 1 shows the pseudo code for the above procedure.

Algorithm 1: Pseudo code for the procedure Δ-*Unification*

Algorithm "Δ-Unification"
Require: Set of optimization problems LP, Problem space S
1. **for all** $lp_j \in LP$ **do** ▹ Solve each single-goal problem
2. $v^j \leftarrow ACS(lp_j)$
3. **end for**
4. ▹ An estimation of the optimal solution for each single-goal problem is established
5. $v^{\Delta} \leftarrow ACS(lp_{diff})$
6. **return** v^{Δ}

4.2. Weighted Frontier Walk (WFW)

The second method is also based on the *goal synthesis* strategy and involves the definition of a multi-goal problem whose objective function consists of a weighted combination of the single-goal problems. Let $w \in \mathbb{R}^{|LP|}$ be a vector of weights, where each weight is associated with a single-goal problem of the set LP. Let of_{wfw} be the objective function of such a problem:

$$of_{wfw}(x) = \min_{lp_j \in LP}(w_j \cdot |p(v^j, c^j) - p(x, c^j)|/p(v^j, c^j)). \tag{8}$$

The objective function of_{wfw} is similar to the objective function defined for the Δ-Unification method in Section 4.1. In scenarios where the weights w are all equal, then this method is equivalent to the Δ-Unification method described in the previous section (see Section 4.1).

The new multi-goal optimization problem is defined as:

$$lp_{wfw} := \min\{of_{wfw}(x) | Ax = b \wedge x \geq 0\}. \tag{9}$$

Algorithm 2 shows the pseudo code for the above procedure.

Algorithm 2: Pseudo code for the procedure *Weighted Frontier Walk*

Algorithm "Weighted Frontier Walk"
Require: Set of optimization problems LP, Problem space S, Vector of weights w
1. **for all** $lp_j \in LP$ **do** ▹ Solve each single-goal problem
2. $v^j \leftarrow ACS(lp_j)$
3. **end for**
4. ▹ An estimation of the optimal solution for each single-goal problem is established
5. $v^{wfw} \leftarrow ACS(lp_{wfw})$
6. **return** v^{wfw}

4.3. Iterative Superposition (IS)

The third method is based on the idea of *superposition*. A complete solution is first required for each of the single-goal problems and then the solution to the multi-goal problem is taken as combination of them. The combination is computed as the minimum intersection of the distribution plans. Let v^{int} be the result of the minimum intersection of the distribution plans. Each element $v_i^{int} \in v^{int}$ is defined as:

$$v_i^{int} = \min_{lp_j \in LP}(v_i^j). \tag{10}$$

The vector result of the minimum intersection is then used to reduce the problem space of each single-goal problem, by applying the reduction function defined above in Eq. (4).

It is unlikely that the first solution will satisfy all the required demands. As such the solution for the multi-goal problem is initialized as a vector of zeros of dimension $|x| = n$, $v^{mg} = \mathbf{0}_n$, and the intersection is added at each step $v^{mg} = v^{mg} + v^{int}$.

The procedure is repeated until such time as the demands are satisfied; a solution in the reduced space is computed for each single-goal problem, and the solution to the multi-goal problem is once again the minimum intersection. The pseudo code for this procedure is shown in Algorithm 3.

During the first step of the procedure, a complete solution is found for each single-goal problem. The solution to the multi-goal problem is then generated from the individual single-goal problem solutions. Uniformly from each single-goal solution v^j, we iteratively take the best elements of v^j and add them to the multi-goal solution v^{mg}, until such time as all demands are satisfied. It should be noted that the possibility exists such that the intersection of the solutions is empty i.e. $v^{int} = \mathbf{0}$; in such instances the reduction function will not modify the search space and the procedure itself may not converge.

Algorithm 3: Pseudo code for the procedure *Iterative Superposition*

Algorithm "Iterative Superposition"
Require: Set of optimization problems LP, Problem space S
$\triangleright$ Initialize multi-goal solution to zero
$v^{mg} \leftarrow \mathbf{0}_n$
repeat
  for all $lp_j \in LP$ **do** $\triangleright$ Solve each single-goal problem
    $v^j \leftarrow ACS(lp_j)$
  end for
  $\triangleright$ Compute solutions intersection
  for all $v_i^j \in v^j$ **do**
    $v_i^{int} \leftarrow \min_{lp_j \in LP}(v_i^j)$
  end for
  $\triangleright$ Reduce the problem space
  $b \leftarrow red(v^{int})$
  $\triangleright$ Add the intersection to the multi-goal solution
  $v^{mg} \leftarrow v^{mg} + v^{int}$
  if $v^{int} = \mathbf{0}$ **then** $\triangleright$ The intersection is null
    Complete v^{mg} with the best elements from $v^j \forall lp_j \in LP$
  end if
until all demands are satisfied
return v^{mg}

4.4. Incremental Solving via Tuning (IT)

The fourth method is based on the *incremental solving* strategy. The procedure starts by solving one of the single-goal problems and then iteratively adjusts the solution to increase its performance according to the remaining single-goal problems. The solution is adjusted by eliminating elements, the X elements that have the greatest negative impact on the current problem solution are eliminated, where $X \in \mathbb{R}$ and $X \leqslant |x|$. Once every single-goal problem has been considered, the problem space is reduced and the process is repeated until such time as all demand is satisfied.

Let ni be the function used to find an element in a given vector $v \in \mathbb{R}^n$ that has the greatest negative impact on the performance of a given optimization problem $lp \in LP$.

$$\begin{aligned} &ni : \mathbb{R}^n \times LP \to \mathbb{N}^+, \\ &v, lp \mapsto \sum \operatorname{argmax}_{i \in [0,|v|]} p\left(v - \left(I_{|v|} \cdot v_i\right)_{[0,|v|],i}, \pi_c(lp)\right). \end{aligned} \tag{11}$$

The pseudo code for this procedure is shown in Algorithm 4.

The method requires that a single-goal optimization problem is set as the starting point and the results are dependent on the order in which single-goal problems are solved. It follows that the procedure should be run on all possible single-goal problem orderings as part of a complete analysis.

Algorithm 4: Pseudo code for the procedure *Incremental Solving via Tuning*

Algorithm "Incremental Solving via Tuning"
Require: Set of optimization problems LP, Problem space S, Number of elements to neglect X, Starting optimization problem $lp_j \in LP$
$\triangleright$ Initialize multi-goal solution to zero
$v^{mg} \leftarrow \mathbf{0}_n$
repeat
  $v^j \leftarrow ACS\left(lp_j\right)$
  $\triangleright$ Adjust the solution by removing the X elements with greatest negative impact on the remaining single-goal problems
  for $i \leftarrow 0$ **to** $X/(|LP| - 1)$ **do**
    for all $lp_k \in LP \setminus lp_j$ **do**
      $v^j_{ni(v^j, lp_k)} \leftarrow 0$
    end for
  end for
  $b \leftarrow red(v^j)$
  $v^{mg} \leftarrow v^{mg} + v^j$
until all demands are satisfied
return v^{mg}

4.5. Incremental Solving via Retention (IR)

The fifth method is based on the *incremental solving* strategy and in reality is a variation on the Incremental Solving via Tuning method described in the Section 4.4. Again the procedure consists of solving each single-goal problem in sequence, but on this occasion rather than eliminating the elements with the greatest negative impact on the performance of the remaining problems, on this occasion the Y elements that have contribute the most to the performance of the current problem are retained, where $Y \in \mathbb{R}$ and $Y \leqslant |x|$.

Let hi be the function used to find the element of a given vector $v \in \mathbb{R}^n$ that has the greatest positive impact on the performance of the given optimization problem $lp \in LP$.

$$\begin{aligned} &hi : \mathbb{R}^n \times LP \to \mathbb{N}^+, \\ &\qquad v, lp \mapsto \sum \operatorname{argmax}_{i \in [0,|v|]} \left(v_i \cdot (\pi_c(lp))_i\right). \end{aligned} \tag{12}$$

The pseudo code for this procedure is shown in Algorithm 5.

Algorithm 5: Pseudo code for the procedure *Incremental Solving via Retention*

Algorithm "Incremental Solving via Retention"
Require: Set of optimization problems *LP*, Problem space *S*, Number of elements to retain *Y*
▹ Initialize multi-goal solution to zero
$v^{mg} \leftarrow \mathbf{0}_n$
▹ Initialize counter of remaining elements that may be retained
$r \leftarrow n$
repeat
  for all $lp_j \in LP$ **do** ▹ Sequentially solve for each single-goal problem
    $v^j \leftarrow ACS\left(lp_j\right)$
    if $r \leq Y$ **then** ▹ All elements have to be retained
      return $v^{mg} + v^j$
    end if
    ▹ Adjust the solution by retaining the Y elements with greatest positive impact on the current single-goal problem
    $v^k \leftarrow \mathbf{0}_n$
    for $i \leftarrow 0$ **to** Y **do**
      $v^k_{hi(v^j - v^k, lp_j)} \leftarrow v^j_{hi(v^j - v^k, lp_j)}$
    end for
    $r \leftarrow r - Y$
    $b \leftarrow red(v^k)$
    $v^{mg} \leftarrow v^{mg} + v^k$
  end for
until all demands are satisfied
return v^{mg}

4.6. Taguchi QLF-based Approach

The idea behind the sixth method originates from the theory of *Robust Engineering* and *Taguchi's Quality Loss Function* (*QLF*) (Taguchi, Elsayed, & Hsiang, 1988) and can be classified as a *goal synthesis* based strategy. Taguchi's quality loss function encodes a penalty term for deviations from a particular target. Here we define a loss function L for each single-goal problem $lp \in LP$, in the form of:

$$L_{lp} : \mathbb{R} \rightarrow \mathbb{R}, \quad t' \mapsto k(t' - t)^2, \tag{13}$$

where $t \in \mathbb{R}$ is the value of the target solution for the problem lp, $t' \in \mathbb{R}$ is the evaluation for another proposed design and $k \in R$ is the loss coefficient in terms of deviation from the target metric. The function computes the penalty, the loss for deviating from the target. Given an optimization problem $lp \in LP$, the value of t is an estimation of the optimum solution to the problem, which may be computed by applying the Ant Colony Solver *ACS*:

$$t = p(ACS(lp), \pi_c(lp)). \tag{14}$$

We may now define a new multi-goal optimization problem based on the loss function L. Let of_{qlf} be the objective function of such a problem:

$$of_{qlf}(x) = \sum_{lp_j \in LP} L_{lp_j}(p(x, \pi_c(lp_j))). \tag{15}$$

The new multi-goal problem consists of minimizing the total deviation loss from the best known solutions of the single-goal problems:

$$lp_{qlf} := \min\{of_{qlf}(x) | Ax = b \wedge x \geqslant 0\}. \tag{16}$$

The pseudo code for this procedure is shown in Algorithm 6.

Algorithm 6: Pseudo code for the procedure *Taguchi QLF-based Approach*

Algorithm "Taguchi QLF-based Approach"
Require: Set of optimization problems *LP*, Problem space *S*
for all $lp_j \in LP$ **do** ▹ Solve each single-goal problem
  $v^j \leftarrow ACS(lp_j)$
end for
▹ An estimation of the optimal solution for each single-goal problem is established
$v^{qlf} \leftarrow ACS(lp_{qlf})$
return v^{qlf}

4.7. ACO-specific Multi-goal Method

The last and final method differs from those previously described in that it is specific to the Ant Colony Optimization algorithm; the main idea here is to improve the global pheromone updating strategy in order to simultaneously consider more than one goal. Given a solution generated by the solver, the level of pheromone deposited is increased in accordance with the performance improvement in each single-goal problem. A generated solution receives a full pheromone update if and only if it is an improved solution for each of the single-goal problems. The method may be classified as belonging to the *goal synthesis* category, despite the fact that it does not formally involve the definition of a multi-goal problem. This classification as *goal synthesis* is justified on the basis that the method employs exploration of the search space by taking into account more than one goal at a time as it walks along the intersection of the single-goal problem spaces.

As stated in Dorigo and Gambardella (1997), the original global pheromone rule is defined as follow:

$$\tau(r,s) = (1 - \rho) \cdot \tau(r,s) + \rho \cdot \Delta\tau(r,s), \tag{17}$$

where (r,s) is an edge of the ant tour or a route in the network, $\tau(r,s)$ is the pheromone value deposited on the edge, ρ is the decay parameter $\rho \in [0,1]$, and $\Delta\tau(r,s)$ is a measure of the improvement in the solution.

Given a solution $v \in R^n$ for the problem $lp \in LP$, let δ be the function to measure the increase applied to the pheromone level:

$$\delta : \mathbb{R}^n \rightarrow \mathbb{R}, \ v \mapsto \sum_{lp \in LP} \frac{1}{|LP|} \cdot \left| \frac{p(v, \pi_c(lp))}{p(v^*, \pi_c(lp))} \right|, \tag{18}$$

where $v^* \in \mathbb{R}^n$ is the best known solution for the problem lp. The improved pheromone update strategy maybe stated as:

$$\tau(r,s) = ((1 - \rho) \cdot \tau(r,s) + \rho \cdot \Delta\tau(r,s)) \cdot \delta(v). \tag{19}$$

Let ACS_{mo} be the variant of the Ant Colony Optimization algorithm based on such a global pheromone update strategy.

The pseudo code for this procedure is shown in Algorithm 7.

The procedure requires selecting a single-goal problem $lp_k \in LP$ to be used by the solver as the main problem to solve. The advantage of using a single-goal problem instead of defining a multi-goal one is that the procedure should find a feasible solution for lp_k even if the intersection of the single-goal problem search spaces is empty. Although it is not unreasonable to expect that the solution will be strongly influenced by the goal of the problem lp_k.

Algorithm 7: Pseudo code for the procedure *ACO-specific Multi-goal Method*

Algorithm "ACO-specific Multi-goal Method"
Require: Set of optimization problems LP, Problem space S, Single-goal problem used by the ACS_{mo} variant $lp_k \in LP$
1. **for all** $lp_j \in LP$ **do** ▹ Solve each single-goal problem
2. $\quad v^j \leftarrow ACS(lp_j)$
3. **end for**
4. ▹ An estimation of the optimal solution for each single-goal problem is established
5. $v^{mo} \leftarrow ACS_{mo}(lp_k)$
6. **return** v^{mo}

Table 2
Ant Colony System set of parameters for all tested problem instances. These parameters are from the original definition of the Ant Colony System in Dorigo, Maniezzo, and Colorni (1996).

Parameter	Value
Number of ants	20
Maximum N° of iterations	1000
Pheromone evaporation rate (ρ)	0.1
Weight on pheromone information (α)	1
Weight on heuristic information (β)	20
Exploitation to exploration ratio (Q0)	0.9

5. Experiments

Each of the proposed methods has been tested on a set of 4 single-goal optimization problems: for maximum profit, for minimum transportation cost, for minimum transit and inventory time, and for maximum network resilience. The mathematical definition of the problems are taken from Veluscek et al. (2014). The profit maximization problem has been extended to consider inventory policy and stochastic variability in transportation costs (see Bravo and Vidal (2013) for examples of models that consider inventory policy and stochastic variability). As in Veluscek et al. (2014), the data sets were provided by a real-world manufacturing company with a worldwide dealership network and an active interest in logistics optimization. The company provided the transportation network map, demand data for 432 dealers in the period from January 2010 to December 2011, and data relating to the manufacturing costs, production capacities and regional sale prices.

The problem complexity is quite significant due to the fact that the underlying transportation network is made up of 8 production facilities, 432 dealer locations and 48 shipping ports. The network representation is a four layer graph where:

1. The production facilities are connected both to the outbound shipping ports and the dealer locations;
2. At the outbound shipping ports it is possible to send product to the set of inbound shipping ports;
3. And the inbound shipping ports are connected to the dealer locations.

This network design resulted in almost 8 million potential routes between production facilities and dealer locations.

The single-goal problems have first been solved to define a baseline against which the performance of the proposed composite goal methods can be compared. The solutions produced by the methods have been evaluated according to the single-goal objectives. Table 3 shows the percent difference between the performance of the single-goal problems and the performance of the combination methods. The incremental methods (Sections 4.4 and 4.5) have been run on all the possible orders of the single-goal problems and the method Weighted Frontier Walk (Section 4.2) has been run on a set of 16 weight combinations.

As discussed the method used to solve the optimization problems is the Ant Colony System algorithm from Dorigo and Gambardella (1997), with Vogel's Approximation Method of Allocation as described in Samuel and Venkatachalapathy (2011) being used to establish the starting solution. The parameters used for the test cases are reported in Table 2. Table 3 shows the runtime results for the experiments. This particular implementation of the Ant Colony System has been successfully deployed on the production line of Caterpillar Inc. for 4 out 16 planned products.

When comparing optimization methods for multi-goal problems, it is usually difficult to rank one approach over another in absolute terms. Ideally, we want a method that produces a solution with same performance as those produced when optimizing for each single goal, but in practice this is difficult to achieve. The results not only depend on the definition of the multi-goal method, but also on the properties of the single-goal problems. For example, problems might conflict or be mutually exclusive.

In the case of transportation network optimization, one common denominator could be profit: most of the metrics such as transportation time and network resilience can be monetized. However, in real business environments profit alone may not always be the dominant factor, distribution plans that yield lower profit, but offer greater value with respect to other metrics may be preferred. For instance, resilience implies risk, some companies are more averse to risk taking than the impact on profit alone would imply. Total inventory carrying costs equate to cash flow and/or funds tied up in the business that cannot otherwise be invested elsewhere; when trading volumes are low cash flow may become more important than pure profit.

However, one possible evaluation scenario would be to calculate the relative performance of methods by ranking each method by goal and then combine the ranked position of a method on each goal by summing its position on the different goals. Table 4 shows the result of the ranking procedure.

From Table 4, we can observed that methods based on the incremental solving strategy (i.e. Incremental Solving by Tuning 4.4 and Incremental Solving by Retention 4.5) are positioned in the top ranking. While a ranking approach is appropriate in creating a discrete ordering, it does not necessarily convey information about the relative relationships between the experiments/goals.

Fig. 2 shows an alternative visual representation of the data from Table 3. From Fig. 2, we can see that there is no consistent difference between the methods that belong to the same generic strategy (as defined in Section 2). For instance, methods based on goal synthesis, such as *Weighted Frontier Walk*, *Delta-Unification*, *Taguchi QLF-based* and *ACO-specific* all produce distribution plans with similar performances on the single-goal problems (i.e. very low difference for profit, slightly higher for resilience and higher still for time). Similarly for methods based on incremental solving strategy, such as *Incremental Solving by Tuning* and *Incremental Solving by Retention* have produce distribution plans whose performance is dependent on the order in which the single-goal problems have been solved. In that the first single-goal problem to be solved has greatest influence on the overall solution. For example, by taking the *Incremental Solving by Tuning* method and running it in the order Resilience-Time-Profit produces a distribution plan that is not dissimilar to the performance produced for the order Resilience-Profit-Time, but is completely different from that produced for the order Time-Resilience-Profit.

Arguably the *Iterative Superposition* method is the best one. The distribution plan found performs well for each single-goal problem

Table 3
Percentage difference between single-goal problems and composite goal methods performances.

	Profit (%)	Time (%)	Cost (%)	Resilience (%)	Running time (s)
WFW 52-87-62-35	1.56	156.62	1.57	4.89	1008
WFW 1-14-36-84	1.62	173.94	1.64	5.05	1022
WFW 74-92-43-81	2.08	191.25	2.17	5.05	1022
WFW 94-97-33-25	1.65	159.74	1.68	5.22	1008
WFW 0-92-11-50	1.49	165.82	1.49	5.24	1022
WFW 63-28-60-2	1.45	166.37	1.45	5.37	1022
WFW 2-28-32-7	1.81	180.15	1.86	4.42	1022
WFW 54-88-21-57	1.59	159.68	1.60	5.06	1008
WFW 59-54-86-77	2.08	149.90	2.16	4.09	1022
WFW 92-5-11-64	1.48	164.61	1.48	5.20	1022
WFW 69-85-60-0	1.45	156.10	1.44	5.20	1022
WFW 22-94-90-56	1.90	175.05	1.96	4.70	1022
WFW 29-50-31-2	1.52	175.51	1.53	5.11	1022
WFW 57-40-75-28	1.48	177.30	1.48	4.83	1008
WFW 1-31-7-75	1.89	164.00	1.95	4.56	1022
WFW 100-100-100-100	1.46	160.06	1.46	5.22	1022
Taguchi QLF	5.21	193.63	100.98	4.24	3234
IS	37.34	1.92	42.28	1.86	616
IT T-P-R-C	36.99	1.56	41.88	1.67	924
IT C-R-P-T	1.01	128.43	1.05	5.83	910
IT C-R-T-P	1.01	128.43	1.05	5.83	910
IT C-P-R-T	1.01	128.22	1.05	5.83	910
IT C-P-T-R	1.01	128.22	1.05	5.83	924
IT C-T-R-P	1.01	128.43	1.05	5.83	910
IT C-T-P-R	1.01	128.22	1.05	5.83	308
IT R-C-P-T	49.16	353.93	55.72	2.36	308
IT R-C-T-P	49.00	363.48	55.53	2.51	322
IT R-P-C-T	51.56	389.74	58.45	3.25	308
IT R-P-T-C	49.38	359.98	55.97	2.90	308
IT R-T-C-P	50.33	383.10	57.06	2.74	308
IT R-T-P-C	49.12	364.35	55.67	2.65	910
IT P-C-R-T	1.04	117.83	1.11	5.89	924
IT P-C-T-R	1.04	117.83	1.11	5.89	910
IT P-R-C-T	1.04	117.83	1.11	5.89	910
IT P-R-T-C	1.03	118.73	1.10	5.39	910
IT P-T-C-R	1.03	118.73	1.10	5.39	924
IT P-T-R-C	1.03	118.73	1.10	5.39	616
IT T-C-R-P	36.70	1.04	41.55	1.24	630
IT T-C-P-R	36.70	1.04	41.55	1.24	616
IT T-R-C-P	36.70	1.04	41.55	1.24	630
IT T-R-P-C	36.99	1.56	41.88	1.67	616
IT T-P-C-R	36.99	1.56	41.88	1.67	5684
IR T-P-R-C	24.86	88.69	28.08	1.71	6356
IR C-R-P-T	9.85	154.35	11.00	3.48	6076
IR C-R-T-P	5.13	161.40	5.63	2.53	6398
IR C-P-R-T	7.44	158.35	8.26	2.68	6468
IR C-P-T-R	6.81	146.43	7.54	2.89	6300
IR C-T-R-P	10.01	158.84	11.19	1.51	6412
IR C-T-P-R	6.64	146.32	7.36	2.80	5236
IR R-C-P-T	14.73	151.08	16.55	3.79	5138
IR R-C-T-P	16.43	150.41	18.49	3.04	5208
IR R-P-C-T	14.79	145.78	16.63	3.16	5152
IR R-P-T-C	14.20	135.92	15.95	2.76	5096
IR R-T-C-P	15.26	145.97	17.16	3.27	5068
IR R-T-P-C	13.94	157.32	15.65	2.87	5544
IR P-C-R-T	2.05	120.04	2.13	4.83	5782
IR P-C-T-R	2.00	119.60	2.08	4.75	5418
IR P-R-C-T	2.36	122.07	2.48	4.92	5362
IR P-R-T-C	2.38	119.70	2.51	4.73	5544
IR P-T-C-R	1.89	118.72	1.94	4.79	5446
IR P-T-R-C	2.31	118.91	2.42	4.52	5754
IR T-C-R-P	26.60	78.76	30.06	1.20	5838
IR T-C-P-R	24.78	65.05	27.99	1.15	5558
IR T-R-C-P	26.24	85.80	29.64	1.47	5474
IR T-R-P-C	24.64	88.11	27.83	1.48	5810
IR T-P-C-R	24.47	89.50	27.63	1.14	6358
ACO-Specific	3.01	183.73	3.22	5.70	3234
DU	1.30	167.10	1.28	5.78	840

WFW a–b–c–d stands for Weighted Frontier Walk and a, b, c, and d are the percentage weights assigned to the single goals. IT g1–g2–g3–g4 and IR g1–g2–g3–g4 stand for Incremental Solving via Tuning and Incremental Solving via Retention respectively and g1–g2–g3–g4 defines the order used to solve the single goal problems. P stands for maximum profit, T stands for minimum transit time, R stands for highest resilience, and C stands for minimum transportation cost.

Table 4
Composite goal methods ranking.

	Profit	Time	Cost	Resilience	Sum	Rank of sum
IS	65	8	64	16	153	51
IT T-P-R-C	61	5	60	12	138	31
IT C-R-P-T	2	31	5	64	102	8
IT C-R-T-P	2	31	5	64	102	8
IT C-P-R-T	5	28	2	64	99	5
IT C-P-T-R	5	28	2	64	99	5
IT C-T-R-P	2	31	5	64	102	8
IT C-T-P-R	5	28	2	64	99	5
IT R-C-P-T	68	66	67	17	218	68
IT R-C-T-P	66	68	65	18	217	67
IT R-P-C-T	72	72	71	30	245	72
IT R-P-T-C	69	67	68	27	231	70
IT R-T-C-P	71	71	70	22	234	71
IT R-T-P-C	67	69	66	20	222	69
IT P-C-R-T	11	15	12	70	108	11
IT P-C-T-R	11	15	12	70	108	11
IT P-R-C-T	11	15	12	70	108	11
IT P-R-T-C	8	19	9	57	93	1
IT P-T-C-R	8	19	9	57	93	1
IT P-T-R-C	8	19	9	57	93	1
IT T-C-R-P	58	2	57	6	123	18
IT T-C-P-R	58	2	57	6	123	18
IT T-R-C-P	58	2	57	6	123	18
IT T-R-P-C	61	5	60	12	138	31
IT T-P-C-R	61	5	60	12	138	31
IR T-P-R-C	55	13	54	15	137	30
IR C-R-P-T	44	43	43	32	162	57
IR C-R-T-P	39	52	39	19	149	45
IR C-P-R-T	43	47	42	21	153	51
IR C-P-T-R	42	39	41	26	148	43
IR C-T-R-P	45	48	44	11	148	43
IR C-T-P-R	41	38	40	24	143	37
IR R-C-P-T	48	42	47	33	170	61
IR R-C-T-P	51	41	50	28	170	61
IR R-P-C-T	49	36	48	29	162	57
IR R-P-T-C	47	34	46	23	150	47
IR R-T-C-P	50	37	49	31	167	60
IR R-T-P-C	46	46	45	25	162	57
IR P-C-R-T	32	25	32	43	132	28
IR P-C-T-R	31	23	31	41	126	21
IR P-R-C-T	36	26	36	46	144	39
IR P-R-T-C	37	24	37	40	138	31
IR P-T-C-R	28	18	28	42	116	15
IR P-T-R-C	35	22	35	37	129	25
IR T-C-R-P	57	10	56	5	128	24
IR T-C-P-R	54	9	53	4	120	16
IR T-R-C-P	56	11	55	9	131	27
IR T-R-P-C	53	12	52	10	127	22
IR T-P-C-R	52	14	51	3	120	16
ACO-Specific	38	63	38	62	201	64
DU	15	57	15	63	150	47
WFW 52-87-62-35	23	45	23	45	136	29
WFW 1-14-36-84	25	58	25	47	155	54
WFW 74-92-43-81	34	64	34	48	180	63
WFW 94-97-33-25	26	50	26	54	156	55
WFW 0-92-11-50	21	55	21	55	152	49
WFW 63-28-60-2	17	56	17	56	146	40
WFW 2-28-32-7	27	62	27	36	152	49
WFW 54-88-21-57	24	49	24	49	146	40
WFW 59-54-86-77	33	40	33	34	140	35
WFW 92-5-11-64	20	54	20	52	146	40
WFW 69-85-60-0	16	44	16	51	127	22
WFW 22-94-90-56	30	59	30	39	158	56
WFW 29-50-31-2	22	60	22	50	154	53
WFW 57-40-75-28	19	61	19	44	143	37
WFW 1-31-7-75	29	53	29	38	149	45
WFW 100-100-100-100	18	51	18	53	140	35
Taguchi QLF	40	65	72	35	212	66

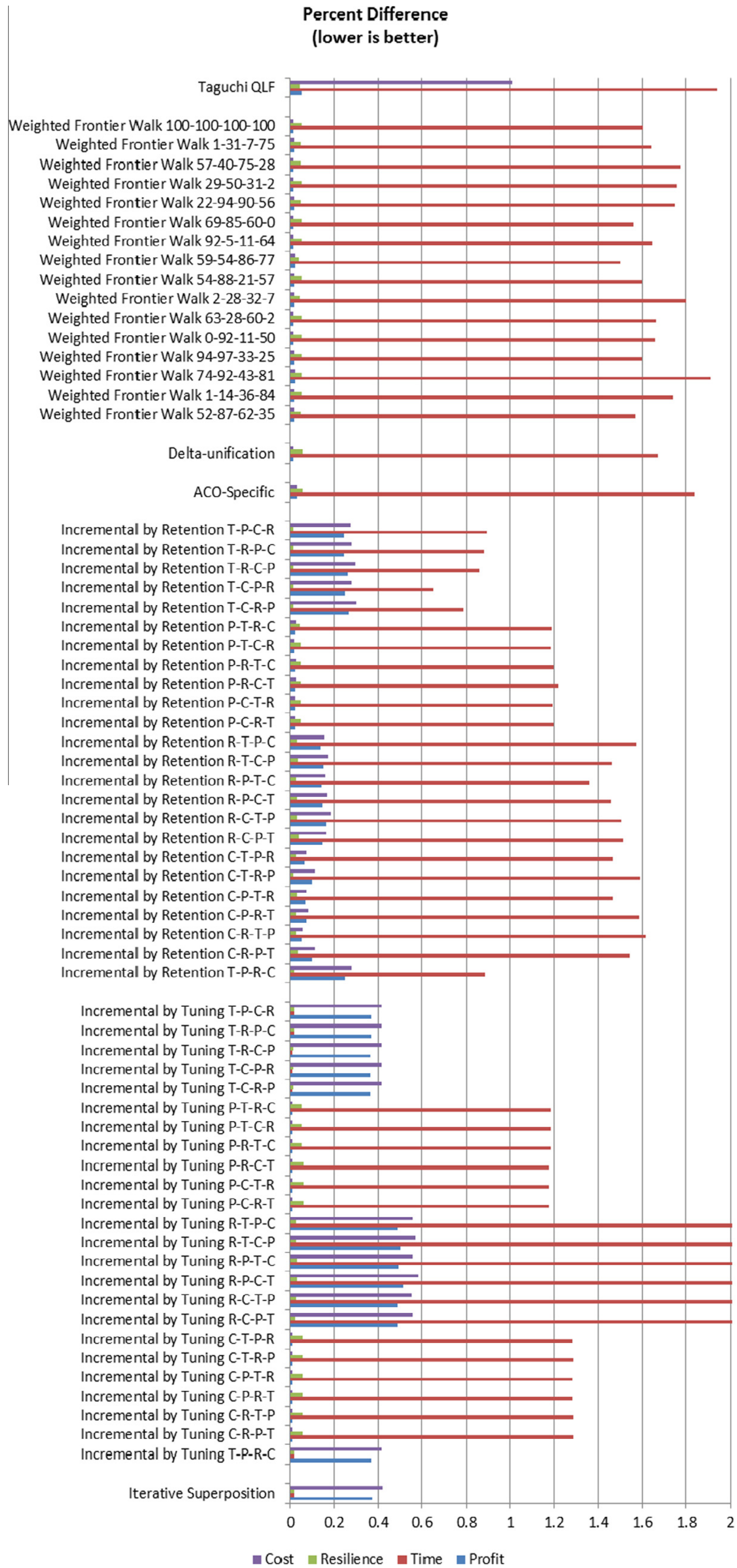

Fig. 2. Visual representation of percentage difference between single-goal problems and composite goal methods performances for the original problem.

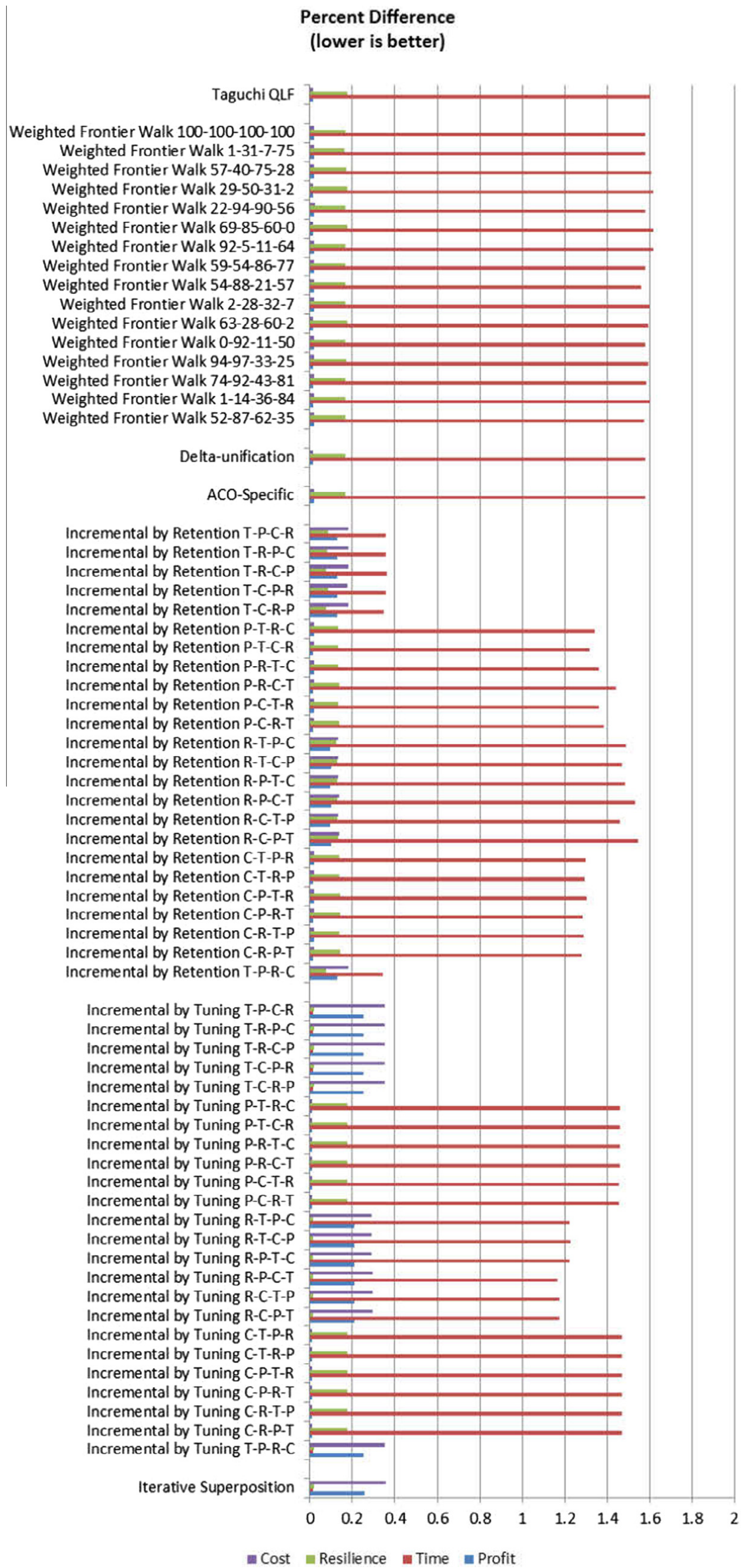

Fig. 3. Visual representation of percentage difference between single-goal problems and composite goal methods performances. The problem is randomly generated according to a normal distribution with mean and standard deviation as in the original data set.

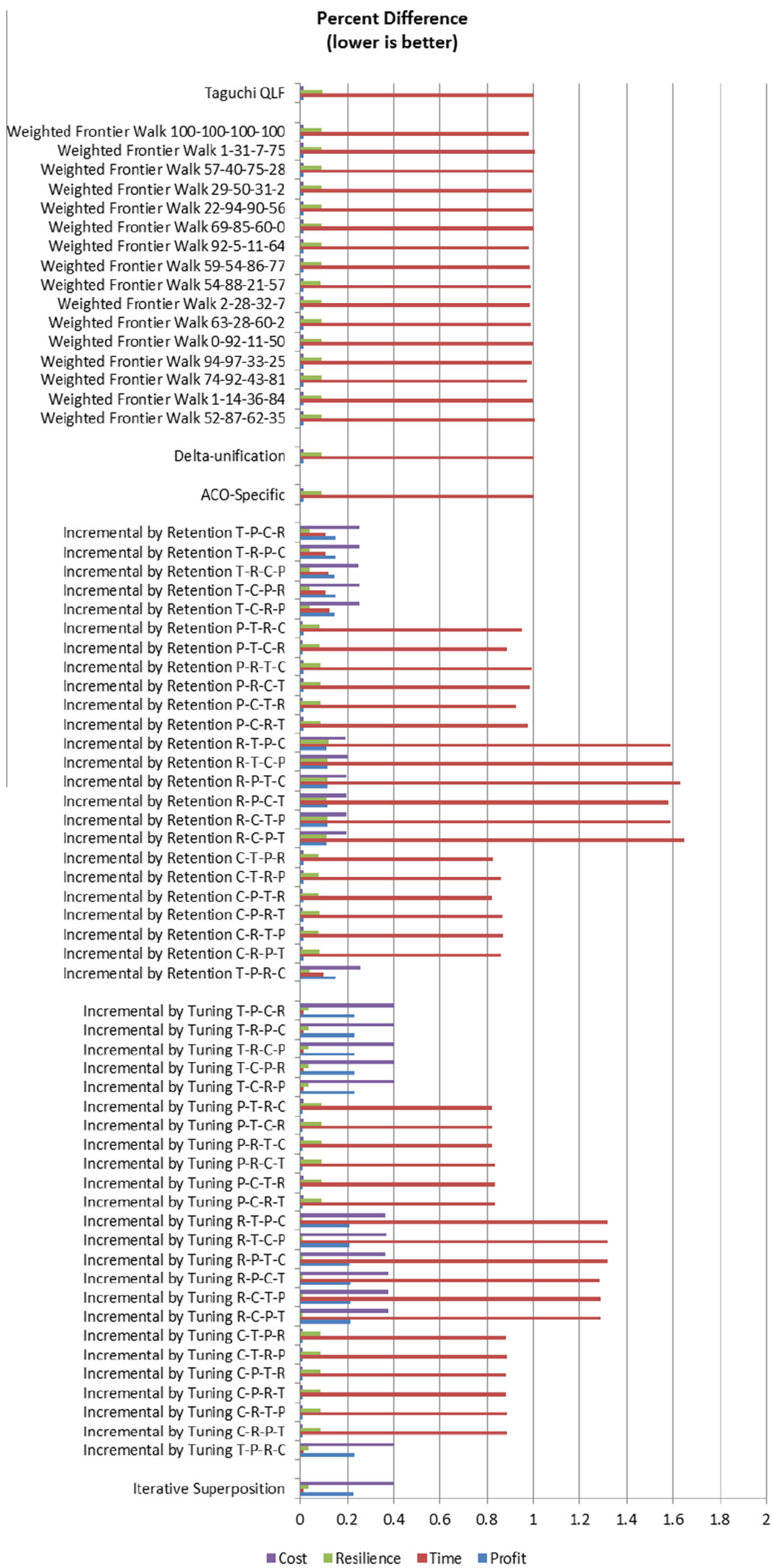

Fig. 4. Visual representation of percentage difference between single-goal problems and composite goal methods performances. The problem is randomly generated where the figures are an interval between 0 and an upper limit which is a random increase over the maximum value in the original data, according to a negative exponential distribution.

(where all the percent differences are below 40%) and the gap that exists between single-goal performances is not as large in comparison to the other methods. On occasions, the Incremental Solving by Tuning and Incremental Solving by Retention methods produce similar results, but they exhibit the drawback of having a dependency on the order used to solve the single-goal problems and on the number of elements to be eliminated or retained.

The methods described in this paper also have been tested on two randomly generated problems. In both instances, the number of dealers, production facilities and shipping ports is the same as in the original problem; it is only the demand figures, the production capacities, the transportation times and costs and the sale prices that have been randomly generated. In the first problem, the figures have been generated according to a normal distribution with the same mean and standard deviation as in the original data set (e.g. the demand figures have the same mean and standard deviation as those found in the original problem). The figures for the second problem are randomly generated in an interval between 0 and an upper limit which is a random increase over the maximum value in the original data, according to a negative exponential distribution.

Figs. 3 and 4 show the percent difference between the performance of the single-goal problems and the performance of the combination methods for the two randomly generated problems. Fig. 3 shows the percentage difference for the first randomly generated problem, whereas Fig. 4 shows them for the second.

The Iterative Superposition method has again proved to have the best performance for both randomly generated problems.

The network used for testing presents a very large number of possible paths. However, Caterpillar's business is characterized by having a relative low monthly demand for any given type of product. On average, over a period of 24 months, 12% of all possible routes are used each month to satisfy the demand (see Fig. 9). It is reasonable to expect that due to the large number of possible routes, the problem of maximizing the network resilience is the easiest to solve. The demand may be spread evenly on the network. Moreover, solution for the problems of profit maximization and transportation costs minimization may be expected to overlap as they involve similar economic aspects. This may not be the case for the problem of traveled time minimization. It is possible that even the most expensive production source may be chosen to satisfy a portion of the demand, provided the production facility is closely located in proximity to the dealers. Since we consider ocean lane discounts for lanes with high shipping commitments, the most expensive production source is unlikely to be considered for both the profit maximization and costs minimization goals. The solver would choose an inexpensive production facility, even if its location is not the closet to the dealership, thereby increasing transportation cost.

The results from running the multi-goal optimization methods on the three datasets confirm these expectations. All solutions had low differences with respect to resilience, for profit maximization and cost minimization the differences were similar. The quality of the solution in term of time minimization is inversely proportional to the goals for profit maximization and cost minimization.

As previously discussed, the Iterative Superposition strategy was shown to be the only method capable of finding high quality solutions for all four goals simultaneously, and independently of the initial configuration. One of the main differences between the four multi-goal strategies relates to the information used to find a combined solution. The goal synthesis strategy only uses the heuristic information from the set of single-goal problems; the solution is a combination of this heuristic information. In this context, the heuristic information refers to the information held on the routes in the network which guides the solver in building the distribution plan. For instance, if the goal is to maximize the profit arising from a distribution plan, then the heuristic information is most likely to be the transportation cost for the given routes. Figs. 5 and 6 show the heuristic information matrix associated with the distribution network for the problems of cost minimization and traveled time minimization respectively.

The incremental strategy makes use of heuristic information combined with a partial distribution plan that is optimal for one of the single-goal problems. Figs. 7 and 8 show a visualization of the distribution plan for the problems of cost minimization and traveled time minimization respectively. Recall that the incremental strategy consists of building a solution for one objective, retain-

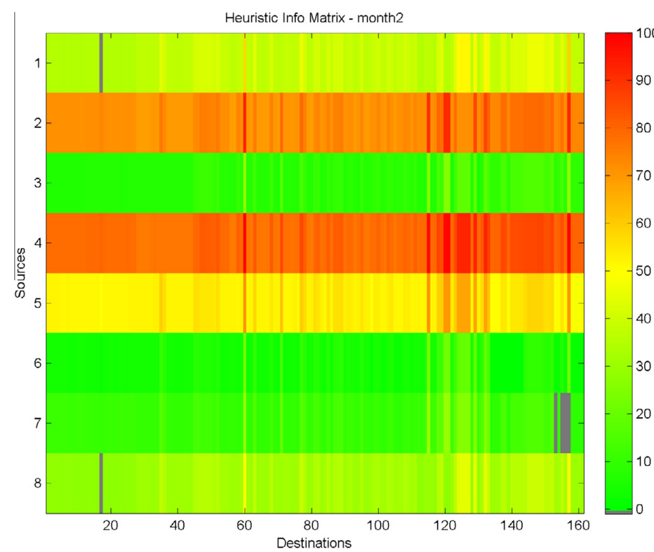

Fig. 5. Heuristic information matrix for the problem of minimization of transportation cost for all possible routes from sources to destinations. The color scale goes from green as most profitable route to red as least profitable route. Gray routes are non-connected routes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

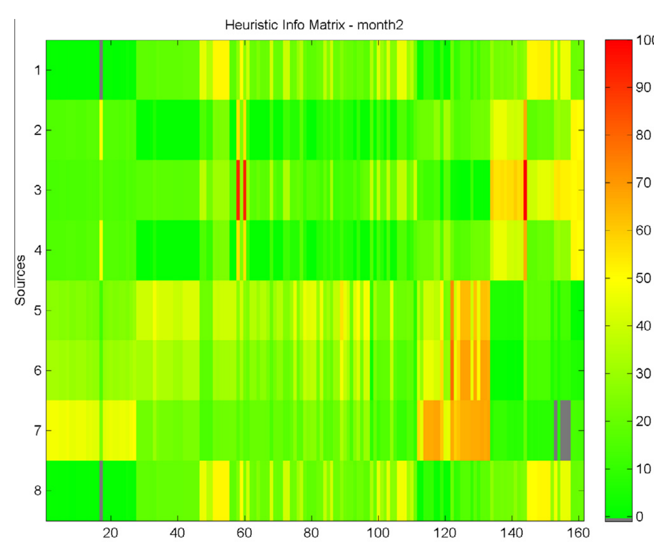

Fig. 6. Heuristic information matrix for the problem of minimization of traveled time for all possible routes from sources to destinations. The color scale goes from green as most profitable route to red as least profitable route. Gray routes are non-connected routes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

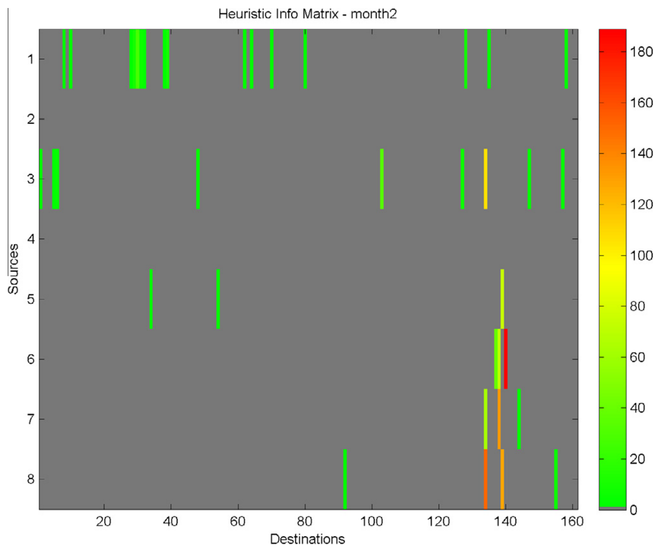

Fig. 7. Distribution plan for the problem of transportation cost minimization. The color scale goes from green as route with only one machine sent through, to red for highly used routes. Gray routes are not used. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

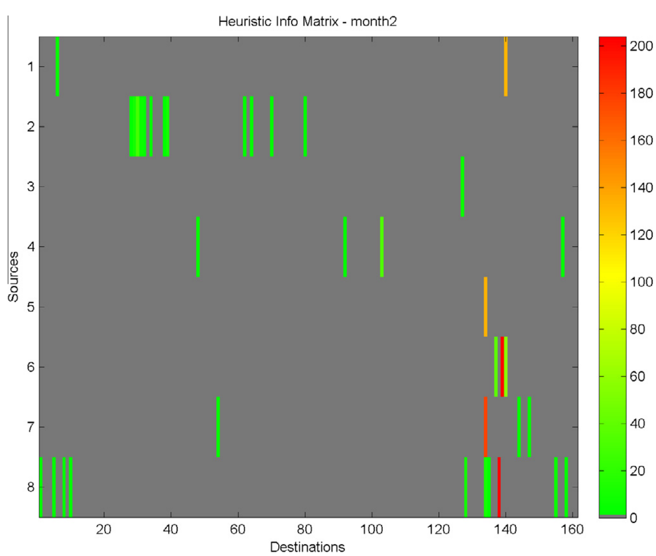

Fig. 8. Distribution plan for the problem of traveled time minimization. The color scale goes from green as route with only one machine sent through, to red for highly used routes. Gray routes are not used. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

ing or removing part of it, and then solving the remaining part in accordance with the next objective.

Whereas the superposition strategy makes only use of the optimal solution for each of the single-goal problems, the solution consists of finding the best solution for each single-goal problem separately, and then using the optimal distribution plans to build a solution for the multi-goal problem. Having the optimal solutions, then the complete heuristic information matrices may appear to be 'noisy' and it stands to reason that combining such solutions may for the majority of cases be the better approach. From a search space perspective, a strategy that works with optimal solutions as its inputs can be expected to produce a multi-goal solution that is closer to all single-goal ones. When starting from the intersection of the heuristic information matrices there is no guaranties that the solution which is the closest to all the single-goal ones will be the one resulting from the intersection of the

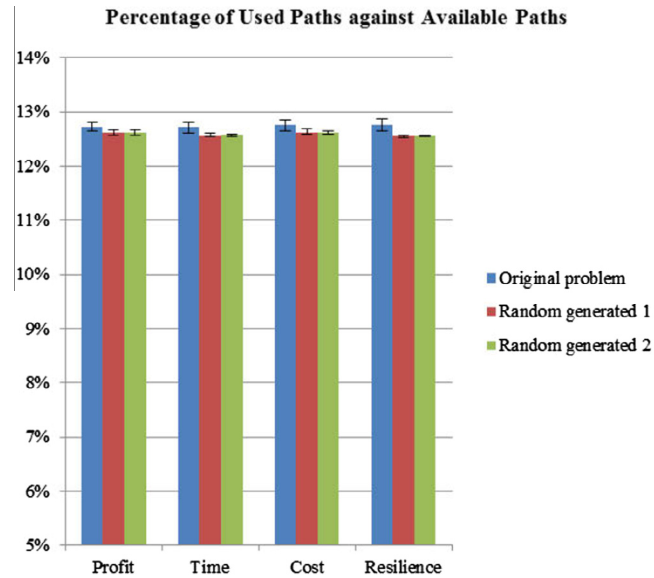

Fig. 9. Mean percentage of used paths against all available paths for all data sets and for all single-goal problems. The data are averaged through a period of 12 months, which each month presents a different dealer demand.

search spaces and could be significantly different from the search spaces of the single-goal problems.

Future work may consist of a theoretical and formal analysis of the different characteristics and behaviors of the discussed strategies to confirm the previous hypothesis.

6. Discussions and future research

The motivation and rational for undertaking this work was to highlight and provide a better understanding of the body of work in the literature relating to multi-goal analysis of transportation network optimization. Clearly from the review undertaken it was evident that more work could be done to enhance knowledge and foster understanding in certain areas of the topic.

Firstly, the literature review of work relating to multi-goal optimization could be extended. While the current problem focuses on transportation network optimization, it is apparent that such work is applicable to and could be extended into others areas of operational research. This would provide an opportunity to better understand which multi-goal optimization methods are preferred and why. It is possible that the current bias towards one specific method for multi-goal optimization is the result of existing software availability. If the current generations of optimization tools do not provide implementation that address multi-goal strategies, practically it is advantageous to adopt the goal synthesis approach and define a multi-goal model, which may be the input to the optimization tool. As future work, the literature could be extended to include information relating to optimization tool implementations and their capabilities.

Moreover, the methods presented here should be tested using a different optimization algorithm and on a differing dataset. While the Ant Colony System (ACS) is a very well established and accepted optimization algorithm, there remains the possibility that our results could be biased by undefined behaviors particular to the ACS algorithm. While testing the hypothesis on two randomly generated problems strengthens the result, it would nevertheless be interesting to test the outcomes on independent datasets, or even a different problem.

Finally, a theoretical analysis of the different characteristics and behaviors that pertain to the discussed strategies is needed to better understand the reasons why and under what circumstances some strategies consistently perform better than others.

7. Conclusion

The aim of this work was to identify the most promising multi-objective optimization techniques available for solving real-world 'industrial' transportation network optimization problems. We have reviewed the state of art for multi-objective optimization and have identified four generic strategies, which we refer to as *goal synthesis*, *superposition*, *incremental solving* and *exploration*. We have implemented seven instances of these four strategies. The preferred approach from analysis and review of the current literature would appear to be the construction of a model that combines single optimization goals. However, our experiment using goal combination methods produced low quality solutions in comparison to those produced by other strategies. In particular, the *superposition* strategy proved to be the most promising solution found, performing well across all single-goal problems and having the additional advantage that it is not dependent on the solution ordering or on the weightings assigned to individual single objectives.

The work presented here has aided in the development of a more accurate optimization model for the business of our industrial partner and has helped in the identification of optimization methods that are capable of producing high quality distribution plans. This work will serve as a reference on multi-objective methods for real-world 'industrial' transportation network optimization problems.

Acknowledgement

Authors would like to thank the Logistics Research & Innovation team at Caterpillar Logistics Inc. for their support of this work.

References

Alaya, I., Solnon, C., & Ghedira, K. (2007). Ant colony optimization for multi-objective optimization problems. *19th IEEE international conference on tools with artificial intelligence (ICTAI 2007)* (Vol. 1, pp. 450–457). IEEE. http://dx.doi.org/10.1109/ICTAI.2007.108.

Altiparmak, F., Gen, M., Lin, L., & Paksoy, T. (2006). A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & Industrial Engineering, 51*(1), 196–215. http://dx.doi.org/10.1016/j.cie.2006.07.011.

Amid, A., Ghodsypour, S. H., & O'Brien, C. (2011). A weighted max–min model for fuzzy multi-objective supplier selection in a supply chain. *International Journal of Production Economics, 131*(1), 139–145. http://dx.doi.org/10.1016/j.ijpe.2010.04.044.

Aslam, T., & Ng, A. H. C. (2010). Multi-objective optimization for supply chain management: A literature review and new development. In *Supply chain management and information systems (SCMIS), 2010 8th international conference on* (pp. 1–8). Retrieved from <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5681724>.

Bevilacqua, V., Costantino, N., Dotoli, M., Falagario, M., Sciancalepore, F., Taylor, P., et al. (2012). Strategic design and multi-objective optimisation of distribution networks based on genetic algorithms. *International Journal of Computer Integrated Manufacturing, 25*(12), 1139–1150. http://dx.doi.org/10.1080/0951192X.2012.684719.

Boudahri, F., Sari, Z., Maliki, F., & Bennekrouf, M. (2011). Design and optimization of the supply chain of agri-foods: Application distribution network of chicken meat. In *2011 International conference on communications, computing and control applications (CCCA)* (pp. 1–6). doi: http://dx.doi.org/10.1109/CCCA.2011.6031424.

Bravo, J. J., & Vidal, C. J. (2013). Freight transportation function in supply chain optimization models: A critical review of recent trends. *Expert Systems with Applications, 40*(17), 6742–6757. http://dx.doi.org/10.1016/j.eswa.2013.06.015.

Cardona-Valdés, Y., Álvarez, A., & Ozdemir, D. (2011). A bi-objective supply chain design problem with uncertainty. *Transportation Research Part C: Emerging Technologies, 19*(5), 821–832. http://dx.doi.org/10.1016/j.trc.2010.04.003.

Chaharsooghi, S. K., & Meimand Kermani, A. H. (2008). An effective ant colony optimization algorithm (ACO) for multi-objective resource allocation problem (MORAP). *Applied Mathematics and Computation, 200*(1), 167–177. http://dx.doi.org/10.1016/j.amc.2007.09.070.

Che, Z. H. (2012). A particle swarm optimization algorithm for solving unbalanced supply chain planning problems. *Applied Soft Computing, 12*(4), 1279–1287. http://dx.doi.org/10.1016/j.asoc.2011.12.006.

Che, Z. H., & Chiang, C. J. (2010). A modified Pareto genetic algorithm for multi-objective build-to-order supply chain planning with product assembly. *Advances in Engineering Software, 41*(7–8), 1011–1022. http://dx.doi.org/10.1016/j.advengsoft.2010.04.001.

Chen, C.-L., & Lee, W.-C. (2004). Multi-objective optimization of multi-echelon supply chain networks with uncertain product demands and prices. *Computers & Chemical Engineering, 28*(6–7), 1131–1144. http://dx.doi.org/10.1016/j.compchemeng.2003.09.014.

Chen, C.-L., Yuan, T.-W., & Lee, W.-C. (2007). Multi-criteria fuzzy optimization for locating warehouses and distribution centers in a supply chain network. *Journal of the Chinese Institute of Chemical Engineers, 38*(5–6), 393–407. http://dx.doi.org/10.1016/j.jcice.2007.08.001.

Cintron, A., Ravindran, A. R., & Ventura, J. A. (2010). Multi-criteria mathematical model for designing the distribution network of a consumer goods company. *Computers & Industrial Engineering, 58*(4), 584–593. http://dx.doi.org/10.1016/j.cie.2009.12.006.

Das, I., & Dennis, J. E. (1998). Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization, 8*(3), 631–657. http://dx.doi.org/10.1137/S1052623496307510.

Ding, H., Benyoucef, L., & Xie, X. (2009). Stochastic multi-objective production-distribution network design using simulation-based optimization. *International Journal of Production Research, 47*(2), 479–505. http://dx.doi.org/10.1080/00207540802426540.

Ding, H., Benyoucef, L., & Xie, X. (2004). A simulation-based optimization method for production-distribution network design. In *Systems, man and cybernetics, 2004 IEEE international conference* (pp. 4521–4526). Retrieved from <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1401244>.

Dorigo, M., & Gambardella, L. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation, 1*(1), 53–66. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=585892.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics. Part B, Cybernetics: A Publication of the IEEE Systems, Man, and Cybernetics Society, 26*(1), 29–41. http://dx.doi.org/10.1109/3477.484436.

García-Martínez, C., Cordón, O., & Herrera, F. (2007). A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP. *European Journal of Operational Research, 180*(1), 116–148. http://dx.doi.org/10.1016/j.ejor.2006.03.041.

Ghoseiri, K., & Nadjari, B. (2010). An ant colony optimization algorithm for the bi-objective shortest path problem. *Applied Soft Computing, 10*(4), 1237–1246. http://dx.doi.org/10.1016/j.asoc.2009.09.014.

Huang, M., Li, R., & Wang, X. (2011). Network construction for fourth-party logistics based on resilience with using particle swarm optimization. *Chinese Control and Decision Conference (CCDC), 2011*, 3924–3929. http://dx.doi.org/10.1109/CCDC.2011.5968907.

Joines, J. A., King, R. E., Kay, M. G., & Gupta, D. (2002). Supply chain multi-objective simulation optimization. *Proceedings of the winter simulation conference* (Vol. 2, pp. 1306–1314). IEEE. http://dx.doi.org/10.1109/WSC.2002.1166395.

Kamali, A., Fatemi Ghomi, S. M. T., & Jolai, F. (2011). A multi-objective quantity discount and joint optimization model for coordination of a single-buyer multi-vendor supply chain. *Computers & Mathematics with Applications, 62*(8), 3251–3269. http://dx.doi.org/10.1016/j.camwa.2011.08.040.

Konak, A., Coit, D. W., & Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety, 91*(9), 992–1007. http://dx.doi.org/10.1016/j.ress.2005.11.018.

Kumar, M., Vrat, P., & Shankar, R. (2004). A fuzzy goal programming approach for vendor selection problem in a supply chain. *Computers & Industrial Engineering, 46*(1), 69–85. http://dx.doi.org/10.1016/j.cie.2003.09.010.

Leung, S. C. H., Tsang, S. O. S., Ng, W. L., & Wu, Y. (2007). A robust optimization model for multi-site production planning problem in an uncertain environment. *European Journal of Operational Research, 181*(1), 224–238. http://dx.doi.org/10.1016/j.ejor.2006.06.011.

Liang, T.-F. (2008). Fuzzy multi-objective production/distribution planning decisions with multi-product and multi-time period in a supply chain. *Computers & Industrial Engineering, 55*(3), 676–694. http://dx.doi.org/10.1016/j.cie.2008.02.008.

Lin, Z., & Wang, L. (2008). Multi-stage partner selection based on genetic-ant colony algorithm in agile supply chain network. In *2008 The 9th international conference for young computer scientists* (pp. 1884–1889). doi: http://dx.doi.org/10.1109/ICYCS.2008.400.

Liu, J., Ding, F., & Lall, V. (2000). Using data envelopment analysis to compare suppliers for supplier selection and performance improvement. *Supply Chain Management: An International Journal, 5*(3), 143–150.

McMullen, P. R., & Tarasewich, P. (2006). Multi-objective assembly line balancing via a modified ant colony optimization technique. *International Journal of Production Research, 44*(1), 27–42. http://dx.doi.org/10.1080/00207540500216748.

Ogunbanwo, A., Williamson, A., Veluscek, M., Izsak, R., Kalganova, T., & Broomhead, P. (2014). Transportation network optimization. *Encyclopedia of Business Analytics and Optimization*, 2570–2583. http://dx.doi.org/10.4018/978-1-4666-5202-6.

Sabri, E. H., & Beamon, B. M. (2000). A multi-objective approach to simultaneous strategic and operational planning in supply chain design. *Omega, 28*(5), 581–598. http://dx.doi.org/10.1016/S0305-0483(99)00080-8.

Sadjady, H., & Davoudpour, H. (2012). Two-echelon, multi-commodity supply chain network design with mode selection, lead-times and inventory costs. *Computers & Operations Research, 39*(7), 1345–1354. http://dx.doi.org/10.1016/j.cor.2011.08.003.

Samuel, A. E., & Venkatachalapathy, M. (2011). Modified Vogel's approximation method for fuzzy transportation problems, Vol. 5, No. 28, pp. 1367–1372.

Taguchi, G., Elsayed, E. A., & Hsiang, T. C. (1988). *Quality engineering in production systems (Mcgraw hill series in industrial engineering and management science)*. Mcgraw-Hill College. Retrieved from <http://www.amazon.com/Quality-Engineering-Production-Industrial-Management/dp/0070628300>, p. 192.

Torabi, S. A., & Hassini, E. (2008). An interactive possibilistic programming approach for multiple objective supply chain master planning. *Fuzzy Sets and Systems, 159*(2), 193–214. http://dx.doi.org/10.1016/j.fss.2007.08.010.

Utama, D. N. D., Djatna, T., Hambali, E., Marimin, & Kusdiana, D. (2011). Multi objectives fuzzy ant colony optimization of palm oil based bioenergy supply path searching. In *Advanced computer science and information system (ICACSIS), 2011 international conference* (pp. 177–182). Retrieved from <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6140746>.

Veluscek, M., Ogunbanwo, A., Williamson, A., Kalganova, T., Broomhead, P., & Grichnik, A. J. (2014). Benchmarking of meta-heuristic algorithms for real-world transportation network optimization, Vol. 10.

Wang, H. S. (2009). A two-phase ant colony algorithm for multi-echelon defective supply chain network design. *European Journal of Operational Research, 192*(1), 243–252. http://dx.doi.org/10.1016/j.ejor.2007.08.037.

Wang, G., Huang, S. H., & Dismukes, J. P. (2004). Product-driven supply chain selection using integrated multi-criteria decision-making methodology. *International Journal of Production Economics, 91*(1), 1–15. http://dx.doi.org/10.1016/S0925-5273(03)00221-4.

Wang, F., Lai, X., & Shi, N. (2011). A multi-objective optimization for green supply chain network design. *Decision Support Systems, 51*(2), 262–269. http://dx.doi.org/10.1016/j.dss.2010.11.020.

Weber, C. A., & Current, J. R. (1993). A multiobjective approach to vendor selection. *European Journal of Operational Research, 68*, 173–184.

Yagmahan, B., & Yenisey, M. M. (2008). Ant colony optimization for multi-objective flow shop scheduling problem. *Computers & Industrial Engineering, 54*(3), 411–420. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0360835207001933>.

Yeh, W.-C., & Chuang, M.-C. (2011). Using multi-objective genetic algorithm for partner selection in green supply chain problems. *Expert Systems with Applications, 38*(4), 4244–4253. http://dx.doi.org/10.1016/j.eswa.2010.09.091.

Yıldız, A. R. (2009). An effective hybrid immune-hill climbing optimization approach for solving design and manufacturing optimization problems in industry. *Journal of Materials Processing Technology, 209*(6), 2773–2780. http://dx.doi.org/10.1016/j.jmatprotec.2008.06.028.