Bi-criteria Convoy Movement Problem

P.N. Ram Kumar, T.T Narendran and A.I Sivakumar

The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology published online 15 September 2009
DOI: 10.1177/1548512909348233

The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology OnlineFirst, published on September 15, 2009 as doi:10.1177/1548512909348233

Original article

Journal of Defense Modeling and Simulation: Applications, Methodology, Technology
XX(X) 1–14

DOI: 10.1177/1548512909348233
http://dms.sagepub.com
SAGE

Bi-criteria Convoy Movement Problem

P. N. Ram Kumar[1], **T. T. Narendran**[2] **and A. I. Sivakumar**[1]

Abstract

The problem of routing and scheduling military convoys adhering to certain constraints is known as the convoy movement problem. This work addresses a bi-criteria model for the peacetime version of the convoy movement problem with minimizing total travel time and travel span as objectives. Two approaches, one based on goal programming and the other based on simulated annealing, are proposed. Encouraging results have been obtained.

Keywords

military convoys, conflicts, peacetime, mathematical models, meta-heuristics

1. Introduction

Defense establishments often need to move large numbers of personnel and equipment from their home bases to regions of conflict, threat, or crisis as rapidly as possible. The process of rapidly moving arms and ammunition safely and securely is, perhaps, as old as the term logistics itself. In the process of movement, each military unit must be deployed as a convoy and certain precedence relationships must be followed among different units. Armored fighting vehicles (AFVs) such as tanks and armored personnel carriers are transported using specially designed trucks called *transporters*. During wartime or emergency situations, convoys move non-stop to their destinations except for minor routine halts for food and rest. During peacetime, convoys are allowed to halt and resume their journey. Convoys having to cross each other along the same road (edge) is known as *conflict*;[1] this is avoided since the edges along which convoys travel may not have the load bearing ability and/or width to accommodate two convoys at a time. Besides, when two convoys travel along the same edge in the same direction, a minimum headway time must be maintained between them. This problem of routing and scheduling military convoys adhering to said constraints is known as the convoy movement problem (CMP).[1] Problems analogous to the CMP include routing of automated guided vehicles (AGVs) in flexible manufacturing systems (FMSs) environment,[2] movement of luggage from different flights along a common automated transportation system to various pickup points,[3] strategic routing of hazardous materials,[4] and scheduling of trains on a single track.[5]

During peacetime, military convoys may have to be moved in a manner that addresses several concerns relating to time and traffic risk. These can be stated as multiple objectives, such as minimization of the overall time needed for the deployment of convoys, minimization of the total travel time of convoys, minimization of the cost of transportation, minimization of disruption to civilian traffic, and maximization of the utilization of transportation assets. This work addresses a bi-criteria model for the peacetime version of the CMP with the following objectives:

- Minimization of the total travel time, i.e. the sum of arrival times of convoys at their respective destinations.
- Minimization of the travel span, i.e. the arrival time of the last convoy.

For a given set of military convoys and origin–destination pairs associated with each, the problem is to determine the routes and schedules for all the convoys such that no two convoys cross or overtake each other along the roads and a minimum headway time is always maintained.

[1] Systems and Engineering Management Division, School of MAE, Nanyang Technological University, Singapore
[2] Department of Management Studies, Indian Institute of Technology Madras, Chennai, India

Corresponding author:
P. N. Ram Kumar Systems and Engineering Management Division, School of MAE, Nanyang Technological University, 50 Nanyang Avenue, 639798 Singapore
E-mail: pnramkumar@ntu.edu.sg

Section 2 presents a brief literature review pertaining to the CMP. In Section 3, the mathematical model for the bi-criteria CMP is presented. Section 4 gives a description of the proposed modified lexicographic approach, followed by a description of the multi-objective simulated annealing approach in Section 5. The characteristics of the generated datasets are explained in Section 6. Sections 7 and 8 present our results and conclusions, respectively.

2. Literature Review

Models that address the problem of military convoy routing and scheduling are categorized under the name of military mobility models. Schank et al.[6] and McKinzie and Barnes[7] reviewed a number of strategic mobility models. They presented an overview of all the models, their operating characteristics, and advantages and disadvantages for mobility modeling applications. They observed the following:

(i) There has been a concerted effort over the last 20 years to improve the fidelity of these models and to improve their interconnectivity.
(ii) Complexity of military logistics requires the use of advanced computer models for analysis.
(iii) Current models use either cumbersome and ineffective classical optimization algorithms or simplistic and ineffective greedy approaches to find solutions. The existing models do not use advanced optimization techniques.

Bovet et al.[3] considered the problem of scheduling the movements of a collection of convoys along the same road to reach different destinations. The movements were subjected to two key constraints: (i) each convoy has to depart from its origin during a pre-specified time window and (ii) convoys must not pass or cross each other along the road. Two formulations were proposed for this problem: one, based on integer programming and the other, based on graphs. Tabu search was used to solve the problem and was compared with a mixed integer programming package. Lee et al.[8] introduced the convoy movement problem and described three approaches for minimizing the total travel time and the travel span of convoys: (i) a branch-and-bound (B&B) algorithm for solving a CMP with delays, (ii) a hybrid approach based on genetic algorithms (GAs) and B&B algorithms (the GA algorithm computes the delays and the B&B algorithm computes the paths), and (iii) a pure GA-based approach to compute the delays as well as paths associated with convoys. All three approaches were tested on hypothetical and real-life scenarios. The authors concluded that incorporating start delays improves the quality of the convoy routes. Chardaire et al.[1] gave a formal specification for the CMP, and established that the corresponding decision version of the problem is NP complete. They introduced an integer programming model based on the concept of a time–space network for minimizing total travel time. Lagrangian relaxation was applied and the dual function was evaluated using a modified version of Dijkstra's algorithm.[9] The authors acknowledged the existence of constraints against crossing and overtaking, but simplified their model by dropping them. Tuson and Harrison[10] demonstrated that a straightforward reformulation of the model proposed by Chardaire et al.[1] renders the real-world-like instances amenable to solution by simple heuristics. They stated that the NP hardness is a worst-case measure of the problem's time complexity and real-world problems may not necessarily be hard. It was shown that the start delays had a positive and significant effect on solution quality. Barring Lee et al.,[8] none of the models in the literature considered more than one objective for the CMP.

With this literature gap as motivation, we augment the mathematical model proposed by Ram Kumar and Narendran[11,12] for the single-objective CMP with a second objective, i.e. minimizing travel span. The resultant model is solved iteratively using the proposed modified lexicographic approach to generate non-dominated solutions for small and medium problems (up to 40 cities and 10 convoys). For larger problems, we propose a simulated annealing based meta-heuristic approach.

3. Mathematical Model

3.1. Assumptions

1. Convoys are assumed to travel at constant speed across the entire network.
2. All convoys have equal priority.
3. All the nodes have sufficient infrastructure for halting more than one convoy.
4. Adequate number of transporters is always available.
5. All convoys are ready at their respective home bases at time 0.

3.2. Sets and Indices

Q set of all the nodes ($S \cup D$)
A set of arcs
S set of source nodes
D set of destination nodes
C set of military convoys needing to be moved, $c = 1, 2, 3, \ldots, C$
M a large integer

3.3. Input Parameters

q_s^c Source node of convoy c, $c \in C$, $q_s^c \in S$
q_d^c Destination node of convoy c, $c \in C$, $q_d^c \in D$
$T_{qq'}$ No. of time units required to traverse between nodes q and q', $(q, q') \in A$

h Minimum headway distance between two convoys (in time units)

3.4. Decision Variables

AT_q^c Arrival time of convoy c at node q

DT_q^c Departure time of convoy c at node q

$$A_{qq'}^c = \begin{cases} 1, \text{ if convoy } c \text{ traverses between nodes } q \text{ and } q' \\ 0, \text{ otherwise} \end{cases}$$

$$X_{qq'}^{cc'} = \begin{cases} 1, \text{ if convoy } c \text{ traverses between nodes } q \text{ and } q' \\ \quad \text{before convoy } c' \text{ traverses between node } q \text{ and } q' \\ 0, \text{ otherwise} \end{cases}$$

$$Y_{qq'}^{cc'} = \begin{cases} 1, \text{ if convoy } c \text{ traverses between nodes } q \text{ and } q' \\ \quad \text{before convoy } c' \text{ traverses between node } q' \text{ and } q \\ 0, \text{ otherwise} \end{cases}$$

Of the five decision variables, AT_q^c and DT_q^c have to be integers. The rest are binary variables.

$$\text{Minimize } \left\{ Z_1 = \sum_c AT_{q_d^c}^c,\ Z_2 \right\} \tag{1}$$

subject to the following constraints:

$$Z_2 \geq AT_{q_d^c}^c \ \forall\ c \in C \tag{2}$$

$$\sum_{q'} A_{qq'}^c - \sum_{q''} A_{q''q}^c = \begin{Bmatrix} 1, \text{ if } q = q_s^c \\ 0, \text{ if } q \neq q_s^c \text{ or } q_d^c \\ -1, \text{ if } q = q_d^c \end{Bmatrix} \forall\ c \in C, q \neq q', q \neq q'', \text{ and } \{(q,q'),(q'',q)\} \in A \tag{3}$$

$$\left. \begin{array}{l} AT_{q'}^c + M(1 - A_{qq'}^c) \geq T_{qq'} A_{qq'}^c + DT_q^c \\ AT_{q'}^c - M(1 - A_{qq'}^c) \leq T_{qq'} A_{qq'}^c + DT_q^c \end{array} \right\} \forall\ c \in C\ ,\ q \neq q', \text{and } (q,q') \in A \tag{4}$$

$$\begin{array}{l} X_{qq'}^{cc'} \leq A_{qq'}^c \ \ \forall (c,c') \in C, (q,q') \in A, c \neq c', q \neq q', \text{ and } c' > c \\ X_{qq'}^{cc'} \leq A_{qq'}^{c'} \ \ \forall (c,c') \in C, (q,q') \in A, c \neq c', q \neq q', \text{ and } c' > c \end{array} \tag{5}$$

$$\begin{array}{l} Y_{qq'}^{cc'} \leq A_{qq'}^c \ \ \forall (c,c') \in C, (q,q') \in A, c \neq c', q \neq q', \text{ and } c' > c \\ Y_{qq'}^{cc'} \leq A_{q'q}^{c'} \ \ \forall (c,c') \in C, (q,q') \in A, c \neq c', q \neq q', \text{ and } c' > c \end{array} \tag{6}$$

$$DT_q^c - AT_q^c \geq 0, \forall c \in C, q \notin S, \text{ and } q \notin D \tag{7}$$

$$\left. \begin{array}{l} MX_{qq'}^{cc'} + M(1 - A_{qq'}^c) + M(1 - A_{qq'}^{c'}) + AT_{q'}^c \geq AT_{q'}^{c'} + h(1 - X_{qq'}^{cc'}) \\ M(1 - X_{qq'}^{cc'}) + M(1 - A_{qq'}^c) + M(1 - A_{qq'}^{c'}) + AT_{q'}^{c'} \geq AT_{q'}^c + hX_{qq'}^{cc'} \end{array} \right\} \forall \begin{array}{l} (c,c') \in C,\ (q,q') \in A, \\ c \neq c', q \neq q', \text{ and } c' > c \end{array} \tag{8}$$

$$\left. \begin{array}{l} MY_{qq'}^{cc'} + M(1 - A_{qq'}^c) + M(1 - A_{q'q}^{c'}) + DT_q^c \geq AT_q^{c'} \\ M(1 - Y_{qq'}^{cc'}) + M(1 - A_{qq'}^c) + M(1 - A_{q'q}^{c'}) + DT_{q'}^{c'} \geq AT_{q'}^c \end{array} \right\} \forall \begin{array}{l} (c,c') \in C,\ (q,q') \in A, \\ c \neq c', q \neq q', \text{and } c' > c \end{array} \tag{9}$$

$$AT_{q_s^c}^c = 0, \forall c \in C \text{ and } q_s^c \in S \tag{10}$$

$$DT_{q_d^c}^c = M, \forall c \in C \text{ and } q_d^c \in D \tag{11}$$

The objective functions minimizing the sum of arrival times of convoys at their respective destinations (Z_1) and minimizing the arrival time of the last convoy (Z_2) are captured in (1) and (2). Constraint set (3) represents the network flow conservation constraints. Constraint set (4) ensures that the speed restrictions are not violated. Constraint sets (5) and (6) check if an arc is common to the path of two convoys. Variables $X_{qq'}^{cc'}$, $Y_{qq'}^{cc'}$ are activated when it is common and two convoys compete to traverse along that arc. Constraint (7) facilitates crossing at node junctions in the network. Constraint set (8) ensures that a minimum headway time of h units is maintained between any pair of convoys that move along an arc in the same direction. Constraint set (9) prohibits convoys from crossing. Constraints (10) and (11) ensure that all convoys are ready to move at their home bases at time 0 and stop after reaching their destinations.

4. Modified Lexicographic Approach (MLA)

In order to explore all the available non-dominated solutions, a few modifications to the basic lexicographic approach are proposed. Suppose that Z_1 and Z_2 represent the two objective functions for the problem P. If S and S' are two solutions to the problem, solution S is said to dominate solution S' if

$$Z_x(S) \leq Z_x(S') \ \forall_x, \text{ and } Z_x(S) < Z_x(S') \text{ for at least one } x, x = \{1, 2, \ldots m\} \quad (12)$$

The proposed approach is implemented in two stages. Bounds are determined for both the objectives within which non-dominated solutions can be found. Thereafter, non-dominated solutions are generated iteratively by solving a series of integer programming (IP) problems. Stage 1 refines the search space for uncovering non-dominated solutions, by tightening the bounds on individual objective functions. A numerical example is presented below to explain this statement.

Example: Suppose that there are three convoys. Consider two situations. If the arrival times of convoys at their respective destinations are 50, 100, and 150, the sum of the arrival times is 300 and the travel span is 150. If the arrival times are 95, 105, and 150, the travel span is still 150 but the sum of the arrival times is 350. For the same value of travel span, there exist two different values of total travel time. Of these, (300, 150) dominates (350, 150). Hence, the search space between 300 and 350 can be ignored for further search. Thus, tightening the bounds of individual objective functions results in effective exploration of the search space. The step-by-step procedure of stage 1 is presented below.

4.1. Stage 1 – Determination of Bounds

1. Solve the problem P by considering Z_1 alone in the objective function and compute the value of Z_2 associated with that solution. Label the obtained values as (Z_1^1, Z_2^1) and assign $Z_2^{lower} = Z_1^1$.
2. Solve the problem P by considering Z_2 alone and compute the value of Z_1 associated with that solution. Label the values obtained as (Z_1^2, Z_2^2) and assign $Z_2^{lower} = Z_2^2$.
3. To find the best value of Z_2 corresponding to the minimum value of Z_1, solve the problem P with Z_2 as the only objective and with the constraint $Z_1 \leq Z_1^1$. Label the solution as Z_2^{temp}.
4. If $Z_2^{temp} \leqslant Z_2^1$, then assign $Z_2^{upper} = Z_2^{temp}$. Otherwise, $Z_2^{upper} = Z_1^2$. Go to next step.
5. To find the best value of Z_1 corresponding to the minimum value of Z_2, solve the problem P with Z_1 as the only objective with the constraint $Z_2 \leqslant Z_2^2$. Label the solution as Z_2^{temp} and go to next step.
6. If $Z_2^{temp} \leqslant Z_1^2$, then assign $Z_1^{upper} = Z_1^{temp}$. Otherwise, $Z_1^{upper} = Z_1^2$.
7. Bounds obtained for objective Z_1 are $(Z_1^{lower}, Z_1^{upper})$ and for Z_2 are $(Z_2^{lower}, Z_2^{upper})$.

4.2. Stage 2 – Generation of Non-dominated Solutions

In stage 2, the modified lexicographic approach is implemented in order to generate the non-dominated solutions.

The underlying principle of this approach is as follows: In Figure 1, points A and B correspond to the bounds on the objective functions Z_1 and Z_2. The set of non-dominated solutions must lie between these two points. In multi-objective optimization problems, it is often impossible to improve an objective without sacrificing others. In the current problem, it may not be possible to improve travel span without compromising total travel time and vice versa. On this premise, a bi-directional iterative search method is used to identify the Pareto-optimal solutions. Applicable only to bi-criteria problems, the key advantage of this approach is the elimination of weights for objective functions. Both of the objective functions are given equal importance. Directional searches from A to B as well as from B to A are performed iteratively to exploit the presence of non-dominated solutions thoroughly (see Figure 2). The number of iterations to be performed along each directional iterative search is based on the range of objective function values $(Z^{upper} - Z^{lower})$. Based on a pilot study, the values for the number of iterations were chosen as shown in Table 1. For each objective function, interval size, the ratio of the range of the function to the number of iterations, is determined using Table 1. The interval size of the objective function IS_j is defined as $(Z_j^{upper} - Z_j^{lower})/n_j \ \forall \ j = 1, 2, \ldots, m$, where n_j is the number of iterations corresponding to objective j and m is the number of objectives.

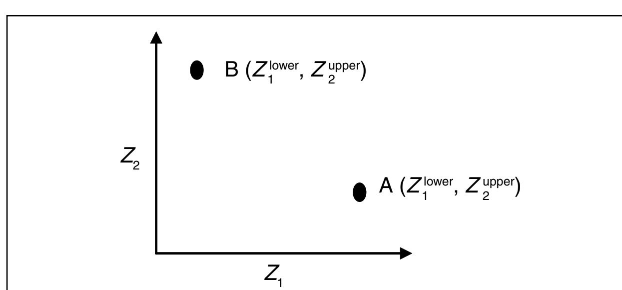

Figure 1. Bounds on objective functions

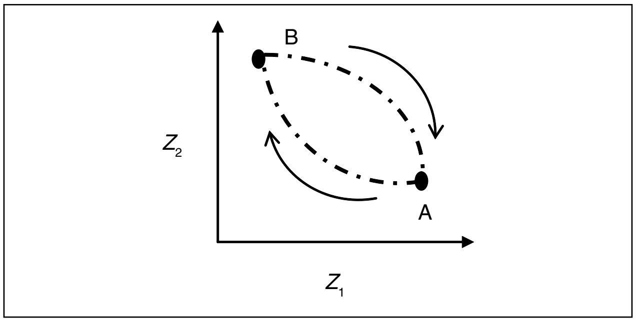

Figure 2. Bi-directional iterative search

Table 1. Objective function range vs number of iterations

Range of objective function values (R)	Number of iterations performed (n)
$R \leqslant 10$	R
$10 < R \leqslant 25$	$R/2$
$25 < R \leqslant 50$	$R/3$
$50 < R \leqslant 100$	$R/4$
$100 < R \leqslant 250$	$R/5$
$250 < R \leqslant 500$	$R/8$
$R > 500$	$R/10$

Let Ω denote the set of non-dominated solutions. If $Z_1^{lower} \neq Z_1^{upper}$ and $Z_2^{lower} \neq Z_2^{upper}$, initialize Ω with solutions $(Z_1^{lower}, Z_2^{upper})$ and $(Z_1^{upper}, Z_2^{lower})$. The bi-directional iterative search looks for non-dominated solutions at uniform intervals from A to B and from B to A. The procedure is as follows:

1. Increment Z_2^{lower} by IS_{Z_2} units, i.e. $Z_2^{lower} = Z_2^{lower} + IS_{Z_2}$.
2. If $(Z_2^{lower} > Z_2^{upper})$, go to step 5. Otherwise, continue.
3. Solve the following optimization problem:
   minimize Z_1
   subject to constraints (2) to (11) and $Z_2 \leqslant Z_2^{lower}$.
4. Assume that the solution obtained by solving the problem in step 3 is (Z_1^{new}, Z_2^{lower}). If $Z_1^{new} < Z_1^{upper}$, then add the obtained solution (Z_1^{new}, Z_2^{lower}) to Ω. Go to step 1.
5. Terminate the directional iterative search procedure from A to B and go to step 6.
6. Increment Z_1^{lower} by IS_{Z_1} units, i.e. $Z_1^{lower} = Z_1^{lower} + IS_{Z_1}$.
7. If $Z_1^{lower} < Z_1^{upper}$, go to step 10.
8. Solve the following optimization problem:
   minimize Z_2
   subject to constraints (2) to (11) and $Z_1 \leqslant Z_1^{lower}$.
9. Assume that the solution obtained by solving the problem in step 8 is (Z_1^{lower}, Z_2^{new}). If $Z_2^{new} < Z_2^{upper}$, then add the obtained solution (Z_1^{lower}, Z_2^{new}) to the non-dominating set Ω. Go to step 6.
10. Terminate the directional iterative search procedure from B to A and record the final Ω as the set of non-dominating solutions generated by the algorithm.

By making a few modifications to the existing lexicographic approach, a set of non-dominated solutions can be generated in an iterative manner.

5. Multi-objective Simulated Annealing (MOSA)

The modified lexicographic approach generates non-dominated solutions by solving the integer program iteratively. Since commercial solvers are unable to solve large problems within a set computational time of 8 h, the proposed modified lexicographic approach may not be computationally viable. For such problems, meta-heuristics can be used. In this section, we describe an adaptation of a simulated annealing approach proposed by Varadharajan and Rajendran[13] to solve the bi-criteria CMP. This technique requires the following parameters to be specified: initial temperature, epoch length (number of iterations at a given temperature), cooling rate, final temperature, and termination criterion. In addition, a perturbation scheme to generate a solution S' in the neighborhood of a given solution S has to be incorporated.

Acceptance-probability function and initial temperature

The acceptance-probability function is defined as

$$F(S, S', T, x) = \exp\{-((Z_x(S') - Z_x(S)) * 100/Z_x(S))/T\}, \quad x \in \{1, 2\} \tag{13}$$

For problem sizes up to 50 cities and 10 convoys, application of the perturbation scheme facilitates acceptance of solutions that are 30% inferior to the initial solution with a probability around 0.95. For large problems, solutions inferior by 50% are also accepted. By substituting these values in Equation (13), the initial temperature T is found to be 575 for smaller and medium problems and 1000 for large problems.

Perturbation scheme

The solution for the CMP comprises individual paths of all the convoys. These are represented as

$$|S_{c_1} - \xi_1^{c1} - \xi_2^{c1} - \xi_3^{c1} \ldots\ldots D_{c_1}|S_{c_2} - \xi_1^{c2} - \xi_2^{c2} - \xi_3^{c2} \ldots\ldots D_{c_2}| \ldots\ldots |S_{c_n} - \xi_1^{cn} - \xi_2^{cn} - \xi_3^{cn} \ldots\ldots D_{c_n}$$

where S_{c_j} denotes the source node of convoy c_j, D_{c_j} denotes the destination node of convoy c_j, and ξ_i^{cj} denotes the ith intermediate node on the path of convoy c_j
The perturbation scheme is as follows:

(i) For each route component, generate a random number between 1 and n, where n is equal to the number of arcs en route from source to destination minus 1. Suppose that the generated number is k.
(ii) For that route component, delete the kth arc and generate a new route between the preceding node of arc k and the convoy's destination.
(iii) Repeat steps (i) and (ii) until all the convoy$S$' route components are perturbed.
(iv) Check for the feasibility of the newly obtained solution. If it is feasible, STOP. Else, go to step (i).

Other parameters of SA

The cooling rate is chosen as 0.80 and the number of iterations at each temperature as 25. The algorithm is terminated when the temperature reaches a value of 10. The parameter settings are chosen on the basis of pilot runs.

Principle of MOSA

Initially, two random solutions are generated. The perturbation scheme is applied 50 times on both solutions with the objectives of minimizing total travel time and travel span. The resultant solutions are Σ^{TTT} and Σ^{SPAN}. Each solution corresponds to a possible Pareto-optimal solution containing the minimum value of total travel time/travel span of convoys. These are given as the starting sequences to the MOSA, which seeks to obtain non-dominated solutions through the implementation of a simple probability function. The function is varied in such a way that the entire search space is covered uniformly.[13] The direction of search is controlled by two parameters x_1 and x_2 and the algorithm is implemented in two phases. In the first phase, x_1 corresponds to travel span and x_2 corresponds to total travel time. S is initialized to Σ^{SPAN}, and x_1 and x_2 are initialized to 0 and 1. After performing the search for the pre-defined number of iterations, the process is repeated for $x_2 = x_2 - 0.1$ and $x_1 = 1 - x_2$. Likewise, x_2 is decreased by 0.1 and x_1 is increased by 0.1, until x_2 is $\geqslant 0.5$. The temperature is progressively decreased by 20% until the final temperature reaches a value of 25. In the second phase, x_1 corresponds to total travel time and x_2 corresponds to travel span, S is initialized to Σ^{TTT}, and x_1 and x_2 are initialized to 1 and 0. The procedure described for phase 1 is repeated. The aim at this stage is to obtain as many Pareto-optimal solutions as possible by searching the entire objective function space

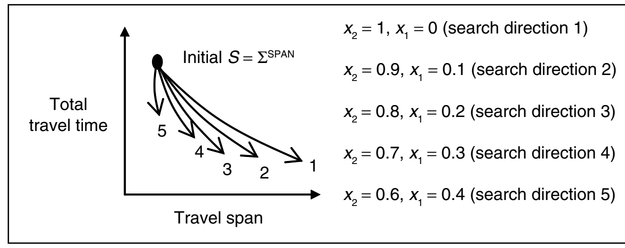

Figure 3. Foci of directions of search in phase 1 with initial $S = \Sigma^{SPAN}$

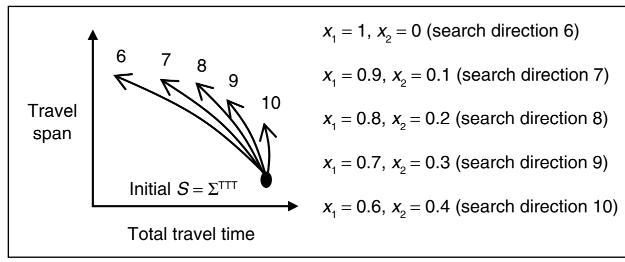

Figure 4. Foci of directions of search in phase 2 with initial $S = \Sigma^{TTT}$

uniformly. The sequences of directions of search corresponding to phases 1 and 2 are shown in Figures 3 and 4. The flow chart describing the working procedure of the MOSA is presented in Figure 5.

The multi-objective simulated annealing algorithm is described below:

1. Generate two random solutions R1 and R2.
2. Apply the perturbation scheme 50 times on R1 and call the resultant solution Σ^{TTT}.
3. Apply the perturbation scheme 50 times on R2 and call the resultant solution Σ^{SPAN}.
4. Initialize the non-dominated set with solutions Σ^{TTT} and Σ^{SPAN}. Call the set Ω and set $K = 0$.
5. Increment K by 1, i.e. set $K = K + 1$.
6. If $K > 2$, go to step 14.
7. If $K = 1$, initialize x_1 (corresponding to travel span) = 0, x_2 (corresponding to total travel time) = 1, and $S = \Sigma^{SPAN}$, go to step 9.
8. If $K = 2$, initialize x_1 (corresponding to total travel time) = 1, x_2 (corresponding to travel span) = 0, and $S = \Sigma^{TTT}$, go to step 9.
9. /* Core operation of simulated annealing algorithm */
   9.1. Initialize temperature T.
   9.2. Do the following 50 times:
   {
   Generate a uniform random number, u, in the range [0, 1];
   If $K = 1$

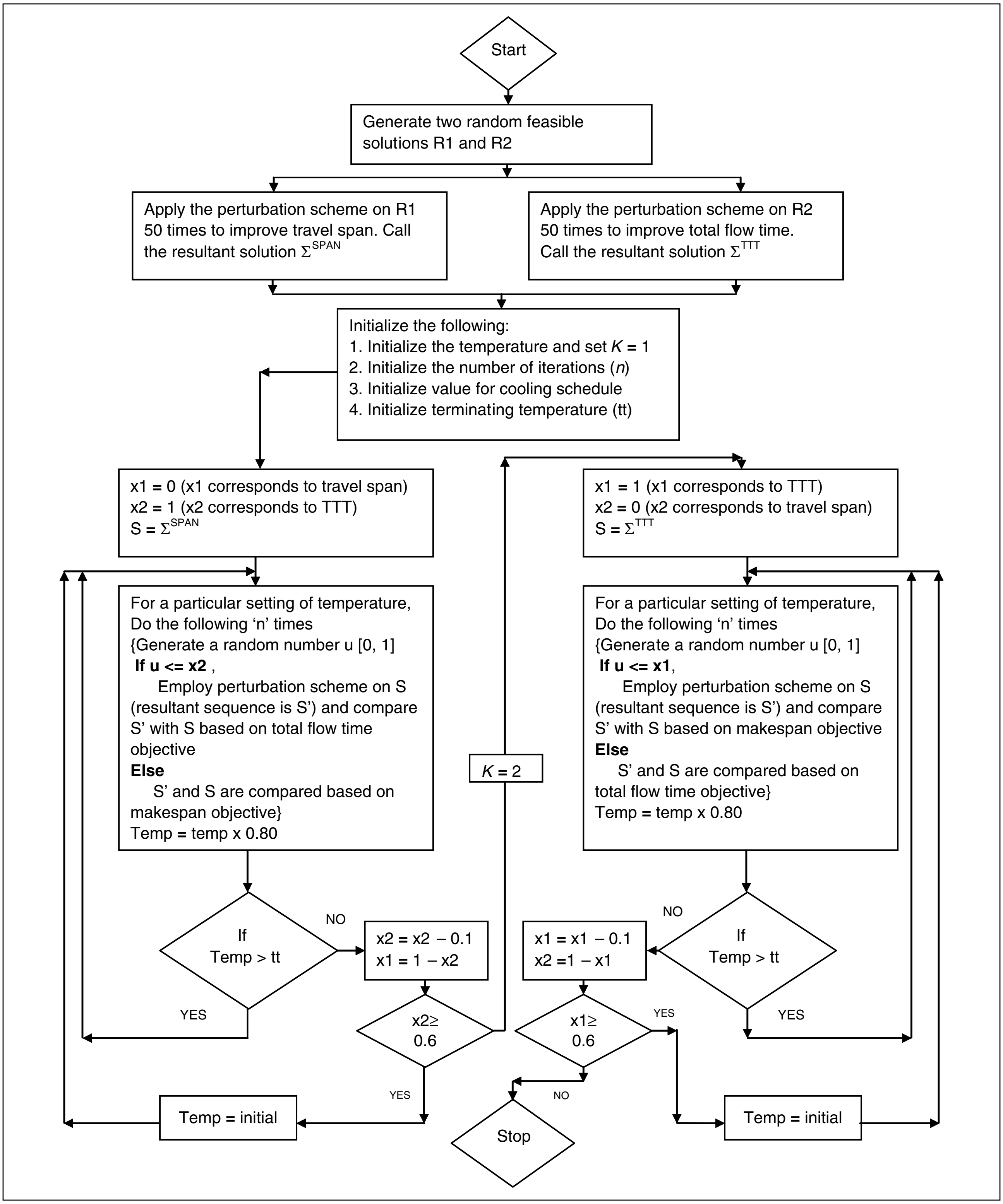

Figure 5. Flow chart for MOSA

{
If $u \leqslant x_2$,
Then, employ the perturbation scheme on S (resulting solution is S') and compare total travel times of S' and S.
Else,
compare travel spans of S' and S
}
If $K = 2$
{
If $u \leqslant x_1$,
Then, employ the perturbation scheme on S (resulting solution is S') and compare travel spans of S' and S.
Else,
compare total travel times of S' and S
}
If S' is better than S, it qualifies for entry, and update Ω accordingly;
}
9.3 Set $T = T * 0.8$
If $T > 25$, go to step 9.2
Else
{
If $K = 1$, proceed to step 10.
If $K = 2$, proceed to step 11.
}

10. Set $x_2 = x_2 - 0.1$ and $x_1 = 1 - x_2$; proceed to step 12.
11. Set $x_1 = x_1 - 0.1$ and $x_2 = 1 - x_1$; proceed to step 13.
12. If $x_2 \geqslant 0.6$, set $S = \Sigma^{SPAN}$ and go to step 9.1. Else, proceed to step 5.
13. If $x_1 \geqslant 0.6$, set $S = \Sigma^{TTT}$ and go to step 9.1. Else, proceed to step 5.
14. STOP and report the solutions present in set Ω as the non-dominated solutions generated by the algorithm.

6. Test Problem Instances

We generate hypothetical test problem instances along the lines of Lee et al.[8] to evaluate the proposed approaches, since benchmark problems are not available for CMP. We increase the possibility of occurrence of conflicts by generating datasets using the following procedure.

1. Nodes with the minimum degree (2) in the network are identified; one of them is chosen as the destination node for a fraction of the total number of convoys.
2. The number of convoys to be routed towards the identical destination node is determined by introducing a factor called the *identical destination factor*, which is the ratio of the number of identical destination nodes to the total number of convoys:

$$\text{identical destination factor } (\theta) = \frac{\text{number of identical destination nodes}}{\text{total number of convoys}}$$

Table 2. Network characteristics of the generated test problems

Problem size	Arc travel times (in time units)		
	Minimum	Maximum	Average
Up to 25 cities, 2–7 convoys	16	240	112.5
30–50 cities, 6–15 convoys	8	120	72.9
60–100 cities, 12–30 convoys	8	96	58.5

An identical destination factor of zero implies that every convoy is destined towards a different destination; a value of 0.5 implies that 50% of the convoys are destined towards a common destination and a value of one implies that all the convoys are destined towards a single destination. Using different combinations of arc-density factor values (0.15, 0.20, and 0.25) and identical destination factors (0 and 0.5), a total of 216 problem instances are generated. The characteristics of the generated datasets are summarized in Table 2.

7. Results and Discussion

The mathematical models were solved using the ILOG CPLEX 9.0 optimizer by allowing them to run until the desired optimal criterion is attained or for 8 h, whichever is earlier. The meta-heuristic was coded in VC++ language and solved using a machine with an Intel Pentium 4.0, 3 GHz processor, and 1 GB of RAM. The MOSA consumed negligible computational time compared to the MLA. Even the largest problem was solved by the MOSA in less than 5 min. The results in terms of number of non-dominated solutions uncovered by each approach are summarized in Tables 3 and 4.

The results indicate that the MLA generates non-dominated solutions for problem sizes up to 40 cities and 10 convoys. The MOSA is able to uncover non-dominated solutions for problem sizes up to 100 cities and 30 convoys. This study allows both algorithms to run up to their respective termination conditions and their performance is compared on the basis of the number of non-dominated solutions.

While applying the MLA for various problem instances, unique non-dominated solutions emerged for about 65% of the problem instances. After the first two steps of stage 1, both the solutions obtained are equal, at least for one objective, implying that there is only a single non-dominated solution for the problem. This often eliminates the use of iterative procedures for subsequent exploration, and indicates that non-dominated solutions are often few in number.

Tables 3 and 4 present the results pertaining to multi-objective datasets with identical-destination factors of 0 and 0.5. The number of intermediate non-dominated

Table 3. Results pertaining to multi-objective datasets for an identical-destination factor (θ) of 0.0

Problem ID (number of cities× number of convoys_arc density factor)	Number of non-dominated solutions obtained			Difference in number of solutions
	MLA		MOSA	
	Stage 1	Final result		
10×2_0.15	1	1	1	0
10×2_0.20	1	1	1	0
10×2_0.25	2	2	2	0
10×3_0.15	1	1	1	0
10×3_0.20	1	1	1	0
10×3_0.25	1	1	1	0
10×4_0.15	1	1	1	0
10×4_0.20	1	1	1	0
10×4_0.25	1	1	1	0
15×3_0.15	1	1	1	0
15×3_0.20	1	1	1	0
15×3_0.25	2	2	3	1
15×4_0.15	1	1	1	0
15×4_0.20	1	1	1	0
15×4_0.25	1	1	1	0
15×5_0.15	1	1	1	0
15×5_0.20	1	1	1	0
15×5_0.25	1	1	1	0
20×4_0.15	2	3	3	0
20×4_0.20	1	1	1	0
20×4_0.25	2	4	3	1
20×5_0.15	1	1	1	0
20×5_0.20	2	2	2	0
20×5_0.25	1	1	1	0
20×6_0.15	1	1	1	0
20×6_0.20	2	2	3	1
20×6_0.25	1	1	1	0
25×5_0.15	1	1	1	0
25×5_0.20	1	1	1	0
25×5_0.25	1	1	1	0
25×6_0.15	2	4	3	1
25×6_0.20	1	1	1	0
25×6_0.25	1	1	1	0
25×7_0.15	1	1	1	0
25×7_0.20	1	1	1	0
25×7_0.25	1	1	1	0
30×6_0.15	1	1	1	0
30×6_0.20	1	1	1	0
30×6_0.25	1	1	1	0
30×7_0.15	1	1	1	0
30×7_0.20	1	1	1	0
30×7_0.25	1	1	1	0
30×8_0.15	1	1	1	0
30×8_0.20	1	1	1	0
30×8_0.25	1	1	1	0
35×7_0.15	1	1	1	0
35×7_0.20	1	1	1	0
35×7_0.25	1	1	1	0
35×8_0.15	1	1	1	0
35×8_0.20	1	1	1	0
35×8_0.25	1	1	1	0
35×9_0.15	1	1	1	0
35×9_0.20	1	1	2	0
35×9_0.25	2	2	2	0

(*Continued*)

Table 3. (Continued)

Problem ID (number of cities× number of convoys_arc density factor)	Number of non-dominated solutions obtained			Difference in number of solutions
	MLA		MOSA	
	Stage 1	Final result		
40×8_0.15	1	1	1	0
40×8_0.20	1	1	1	0
40×8_0.25	1	1	1	0
40×9_0.15	1	1	1	0
40×9_0.20	1	1	1	0
40×9_0.25	1	1	1	0
40×10_0.15	2	2	1	0
40×10_0.20	1	1	1	0
40×10_0.25	1	1	1	0
50×10_0.15	–	–	2	NA
50×10_0.20	–	–	1	NA
50×10_0.25	–	–	2	NA
50×12_0.15	–	–	3	NA
50×12_0.20	–	–	2	NA
50×12_0.25	–	–	1	NA
50×15_0.15	–	–	1	NA
50×15_0.20	–	–	2	NA
50×15_0.25	–	–	1	NA
60×12_0.15	–	–	2	NA
60×12_0.20	–	–	1	NA
60×12_0.25	–	–	1	NA
60×15_0.15	–	–	2	NA
60×15_0.20	–	–	1	NA
60×15_0.25	–	–	3	NA
60×18_0.15	–	–	1	NA
60×18_0.20	–	–	1	NA
60×18_0.25	–	–	1	NA
75×15_0.15	–	–	2	NA
75×15_0.20	–	–	1	NA
75×15_0.25	–	–	3	NA
75×18_0.15	–	–	1	NA
75×18_0.20	–	–	2	NA
75×18_0.25	–	–	2	NA
75×20_0.15	–	–	1	NA
75×20_0.20	–	–	1	NA
75×20_0.25	–	–	3	NA
85×18_0.15	–	–	1	NA
85×18_0.20	–	–	2	NA
85×18_0.25	–	–	1	NA
85×20_0.15	–	–	4	NA
85×20_0.20	–	–	1	NA
85×20_0.25	–	–	2	NA
85×25_0.15	–	–	1	NA
85×25_0.20	–	–	3	NA
85×25_0.25	–	–	1	NA
100×20_0.15	–	–	1	NA
100×20_0.20	–	–	1	NA
100×20_0.25	–	–	2	NA
100×25_0.15	–	–	3	NA
100×25_0.20	–	–	2	NA
100×25_0.25	–	–	1	NA
100×30_0.15	–	–	2	NA
100×30_0.20	–	–	1	NA
1-00×30_0.25	–	–	3	NA

Table 4. Results pertaining to multi-objective datasets for an identical-destination factor (θ) of 0.5

Problem ID (number of cities×number of convoys_ identical destination Factor)	Number of non-dominated solutions obtained			Difference in no. of solutions
	MLA		MOSA	
	Stage 1	Final result		
10×2_0.15	1	1	1	0
10×2_0.20	1	1	1	0
10×2_0.25	1	1	1	0
10×3_0.15	1	1	1	0
10×3_0.20	2	2	2	0
10×3_0.25	1	1	1	0
10×4_0.15	1	1	1	0
10×4_0.20	1	1	1	0
10×4_0.25	1	1	1	0
15×3_0.15	1	1	1	0
15×3_0.20	1	1	1	0
15×3_0.25	2	2	2	0
15×4_0.15	1	1	1	0
15×4_0.20	1	1	1	0
15×4_0.25	1	1	1	0
15×5_0.15	1	1	1	0
15×5_0.20	1	1	1	0
15×5_0.25	1	1	1	0
20×4_0.15	2	2	2	0
20×4_0.20	1	1	1	0
20×4_0.25	2	2	2	0
20×5_0.15	1	1	1	0
20×5_0.20	2	2	2	0
20×5_0.25	1	1	1	0
20×6_0.15	1	1	1	0
20×6_0.20	2	2	3	1
20×6_0.25	1	1	1	0
25×5_0.15	2	3	3	0
25×5_0.20	1	1	1	0
25×5_0.25	2	2	2	0
25×6_0.15	2	2	2	0
25×6_0.20	1	1	1	0
25×6_0.25	1	1	1	0
25×7_0.15	1	1	1	0
25×7_0.20	1	1	1	0
25×7_0.25	1	1	1	0
30×6_0.15	1	1	1	0
30×6_0.20	1	1	1	0
30×6_0.25	1	1	1	0
30×7_0.15	2	2	2	0
30×7_0.20	1	1	1	0
30×7_0.25	1	1	1	0
30×8_0.15	1	1	1	0
30×8_0.20	1	1	1	0
30×8_0.25	1	1	1	0
35×7_0.15	1	1	1	0
35×7_0.20	1	1	1	0
35×7_0.25	1	1	1	0
35×8_0.15	1	1	1	0
35×8_0.20	1	1	1	0
35×8_0.25	1	1	1	0
35×9_0.15	1	1	1	0
35×9_0.20	2	3	3	0
35×9_0.25	2	4	3	1

(Continued)

Table 4. (Continued)

Problem ID (number of cities×number of convoys_ identical destination Factor)	Number of non-dominated solutions obtained			Difference in no. of solutions
	MLA		MOSA	
	Stage 1	Final result		
40×8_0.15	1	1	1	0
40×8_0.20	1	1	1	0
40×8_0.25	1	1	1	0
40×9_0.15	1	1	1	0
40×9_0.20	1	1	1	0
40×9_0.25	1	1	1	0
40×10_0.15	1	1	1	0
40×10_0.20	1	1	1	0
40×10_0.25	1	1	1	0
50×10_0.15	–	–	2	NA
50×10_0.20	–	–	1	NA
50×10_0.25	–	–	2	NA
50×12_0.15	–	–	2	NA
50×12_0.20	–	–	1	NA
50×12_0.25	–	–	1	NA
50×15_0.15	–	–	3	NA
50×15_0.20	–	–	1	NA
50×15_0.25	–	–	2	NA
60×12_0.15	–	–	1	NA
60×12_0.20	–	–	2	NA
60×12_0.25	–	–	2	NA
60×15_0.15	–	–	2	NA
60×15_0.20	–	–	1	NA
60×15_0.25	–	–	2	NA
60×18_0.15	–	–	1	NA
60×18_0.20	–	–	2	NA
60×18_0.25	–	–	3	NA
75×15_0.15	–	–	1	NA
75×15_0.20	–	–	1	NA
75×15_0.25	–	–	2	NA
75×18_0.15	–	–	3	NA
75×18_0.20	–	–	3	NA
75×18_0.25	–	–	2	NA
75×20_0.15	–	–	1	NA
75×20_0.20	–	–	2	NA
75×20_0.25	–	–	1	NA
85×18_0.15	–	–	1	NA
85×18_0.20	–	–	2	NA
85×18_0.25	–	–	2	NA
85×20_0.15	–	–	1	NA
85×20_0.20	–	–	3	NA
85×20_0.25	–	–	2	NA
85×25_0.15	–	–	3	NA
85×25_0.20	–	–	1	NA
85×25_0.25	–	–	2	NA
100×20_0.15	–	–	2	NA
100×20_0.20	–	–	1	NA
100×20_0.25	–	–	2	NA
100×25_0.15	–	–	2	NA
100×25_0.20	–	–	2	NA
100×25_0.25	–	–	1	NA
100×30_0.15	–	–	3	NA
100×30_0.20	–	–	2	NA
100×30_0.25	–	–	3	NA

NA – not applicable

solutions obtained at the end of stage 1 of the MLA is given in column 2. Column 3 indicates the final number of non-dominated solutions generated by the MLA. Column 4 shows the final number of non-dominated solutions generated by the MOSA. The difference between the numbers of non-dominated solutions obtained by both approaches is shown in column 5. Hardware limitations prevented application of the MLA to problem sizes beyond 40 cities and 10 convoys. The following inferences are drawn from the results:

1. For 60% of the problem instances, the number of non-dominated solutions generated is 1. The highest number of non-dominated solutions obtained is 4. More non-dominated solutions are found for larger problems. An increased number of non-dominated solutions is more likely to occur when objectives are more conflicting with one another. Since the objectives used in the CMP are not completely unrelated and conflicting, non-dominated solutions are very small in number.
2. For the same reason, the MLA rarely found additional non-dominated solutions in its second stage.
3. While the MOSA, by its structure, does find more non-dominated solutions than the MLA for several problems, there have been a few cases where the MLA did better than the MOSA in finding solutions. This indicates that the MOSA does not dominate the MLA.
4. Within the datasets used, it is not clear if there is any influence of the identical-destination factor on the number of non-dominated solutions generated.

8. Conclusions and Scope for Further Work

This paper has addressed the bi-criteria convoy movement problem, an important problem in the domain of military logistics. Two methodologies, the modified lexicographic approach based on the theory of goal programming and the multi-objective simulated annealing, have been proposed. The performance of both approaches is evaluated based on the number of non-dominated solutions generated. Encouraging results have been obtained. For some problem instances, the MLA performed better than the MOSA. The reasons for this superior behavior of the MLA on only certain problem instances are yet to be investigated.

References

1 Chardaire P, McKeown G P, Harrison S A, Richardson S B. Solving a time–space network formulation for the convoy movement problem. *Operat. Res.* 2005; 53: 219–230

2 Batta R, Krishnamurthy N, Karwan M H. Developing conflict-free routes for automated guided vehicles. *Operat. Res.* 1993; 41: 1077–90

3 Bovet J, Constantin C, De Werra D. A convoy scheduling problem. *Discrete Appl. Math.* 1991; 30: 1–14

4 Iakovou E, Douligeris C, Li H, Yudhbir L. A maritime global route model for hazardous materials. *Transp. Sci.* 1999; 33: 34–48

5 Higgins A, Kozan E, Ferriera L. Optimal scheduling of trains on a single line track. *Transp. Res. Part B* 1996; 30: 147–61

6 Schank J, Mattock M, Sumner G, Greenberg I, Rothenberg J, Stucker J P. *A review of strategic mobility models and analysis*. Santa Monica, CA: Rand Corporation, 1991, Report R-3926-JS.

7 McKinzie K, Barnes J W. A review of strategic mobility models supporting the defense transportation system. *Math. Comput. Model.* 1993; 39: 839–68

8 Lee Y N, McKeown G P, Rayward-Smith, V J. The convoy movement problem with initial delays. In: Rayward-Smith V J, Osman I H, Reeves C R, Smith G D (eds) *Modern heuristic search methods*. Chichester, UK: John Wiley, 1996, pp. 213–33

9 Dijkstra E W. A note on two problems in connection with graphs. *Numer. Math.* 1959; 1: 269–71

10 Tuson A L, Harrison S A. Problem difficulty of real instances of convoy planning. *J. Operat. Res. Soc.* 2005; 56: 763–75

11 Ram Kumar P N, Narendran T T. A mathematical approach for variable speed convoy movement problem. *Def. Secur. Anal.* 2009; 25: 137–55

12 Ram Kumar P N, Narendran T T. Integer programming formulation for convoy movement problem. *Int. J. Intell. Def. Support Syst.* 2008; 1: 177–88

13 Varadharajan T K, Rajendran C. A multi-objective simulated-annealing algorithm for scheduling in flowshops to minimize the makespan and total flowtime of jobs. *Eur. J. Operat. Res.* 2005; 167: 772–95

Acknowledgements

The authors are grateful to Prof. C. Rajendran, IIT Madras, who contributed greatly to this work with his fruitful suggestions. The authors would also like to thank the two anonymous reviewers for their constructive comments and suggestions to improve the earlier version of this article.

Author biographies

P. N. Ram Kumar is currently working as a post-doctoral research fellow in the School of Mechanical and Aerospace Engineering at Nanyang Technological University, Singapore. He earned his Master's degree in industrial engineering from P.S.G. College of Technology, India and his Doctoral degree in operations management from the Indian Institute of Technology, Madras. His areas of interest are combinatorial modeling, network optimization, and military logistics.

T.T. Narendran is currently a professor in the Department of Management Studies, IIT Madras. He joined the institution in 1976 and has served for over 30 years. During his service, some of his administrative responsibilities include: Head of the Department of Humanities and Social Sciences, Advisor (Training and Placement), and Dean of Industrial Consultancy & Sponsored Research. He has done extensive research in the areas of cellular manufacturing systems and flexible manufacturing systems and has published several papers in leading international journals.

A. I. Sivakumar is an associate professor in the School of Mechanical and Aerospace Engineering (MAE) at the Nanyang Technological University, Singapore and Fellow of the Singapore–Massachusetts Institute of Technology (MIT) Alliance (SMA). He had many years of industrial and research experience in the UK prior to his position in Singapore in 1993. He has a number of international journal publications to his credit. He was the technical chair and co-edited the proceedings of the 3rd and 4th International Conferences on Computer Integrated Manufacturing (ICCIM'95 and ICCIM'97), Singapore.