CrossMark

ORIGINAL PAPER

An advanced tabu search approach to the dynamic airlift loading problem

August G. Roesener[1,3] · J. Wesley Barnes[2]

Abstract This research effort presents a tabu search algorithm to solve the dynamic airlift loading problem. Given a set of palletized cargo items which require transportation from an aerial port of embarkation to an aerial port of debarkation within a pre-specified time frame, the dynamic airlift loading problem seeks to partition the pallets into aircraft loads, select an efficient and effective subset of aircraft from available aircraft, and assign the pallets to allowable positions on those aircraft. The dynamic airlift loading problem differs from many partitioning and packing problems described in the literature because, in addition to spatial constraints, factors such as allowable cabin load, balance restrictions, and temporal restrictions on cargo and aircraft are included. The algorithm developed in this research, the dynamic airlift loading problem-tabu search, was tested on a variety of problem instances. Since real-world solutions are hand generated by subject matter experts and no previous research effort has solved this specific problem, the algorithmic results are compared to compute lower bounds on the number of aircraft trips required.

Abbreviations

AALPS	Automated airlift load planning system
ACL	Allowable cabin load
APOD	Aerial port of debarkation
APOE	Aerial port of embarkation
C-5	C-5 galaxy
C-17	C-17 globemaster III
CB	Center of balance
DALP	Dynamic airlift loading problem
DALP-TS	Dynamic airlift loading problem-tabu search
TPFDD	Time-phased force deployment document
US	United States

✉ August G. Roesener
August.Roesener@gmail.com

J. Wesley Barnes
wbarnes@mail.utexas.edu

1 Department of Operational Sciences, Air Force Institute of Technology, 2950 Hobson Way, Wright Patterson Afb, OH 45433-7765, USA

2 Department of Mechanical Engineering, The University of Texas, 1 University Station, C2200, Austin, TX 78712, USA

3 Present Address: Headquarters Air Mobility Command, 1 Soldier Way, Building 1900 West, Scott Air Force Base, IL 62225, USA

1 Introduction

Power projection, which is the capability to transport military power in an expeditionary manner, represents a large portion of United States (US) military activities. Transportation of personnel and equipment in this endeavor is the responsibility of the US Transportation Command, whose airlift component is air mobility command [40]. In an average week, US Transportation Command operates in 75 % of the world's countries, conducting over 1900 air missions [40].

For several years, a consortium between the Air Force Office of Scientific Research, the University of Texas at Austin, and the Air Force Institute of Technology

addressed various aspects of the overarching Military Mobility Problem which involves transporting personnel and equipment to any point on the planet. Previous research efforts addressed several aspects of this Military Mobility Problem using advanced tabu search methods:

- The aerial fleet refueling problem [3] addressed locating global refueling aircraft flight tracks and joining them with aircraft requiring fuel;
- The theater distribution vehicle routing and scheduling problem [10] evaluated the routing and scheduling of multimodal (i.e., air, sea, rail, truck, etc.) theater transportation assets;
- The strategic airlift problem [21] examined routing cargo-loaded strategic airlift aircraft through a global network of embarkation and debarkation ports;
- The strategic mobility mode selection problem [26] analyzed the proper mode (i.e., air, sea, rail, truck, etc.) to transport palletized and non-palletized cargo items;
- The static airlift loading problem [34] minimized the total strategic airlift aircraft required to transport palletized cargo between a fixed port of embarkation and debarkation pair; and
- The mixed payload airlift loading problem [28] expanded on the static airlift loading problem to include non-palletized cargo items such as vehicles, helicopters, or small boats.

Airlift provides military forces the global reach capability to quickly apply strategic power to various crisis situations worldwide [1]. Airlift has a history spanning over 80 years [6, 14]. While numerous advances have been achieved in both aircraft capabilities and airlift procedures, many areas for improvement still exist. As noted in [34], an area of airlift not yet adequately addressed involves (1) *packing* cargo items onto pallets, (2) *partitioning* the pallets into aircraft loads, (3) *selecting* an efficient and effective subset of aircraft from an available pool of aircraft, and (4) feasibly *placing* the pallets in the best allowable positions. These four tasks are interdependent and their combination, augmented with temporal considerations (e.g., earliest cargo available load date at its aerial port of embarkation (APOE) and earliest and latest delivery dates to its designated aerial port of debarkation (APOD)), defines the dynamic airlift loading problem (DALP).

The analysis directorate of headquarters air mobility command expressed a need for an analytic tool to quickly and effectively generate excellent solutions to DALP tasks 2, 3, and 4. Task 1 was excluded because, in deployment situations, it is usually performed by elements of the US Army.

To describe and solve this problem, this research effort first briefly discusses previous research efforts on similar types of problems. Next, this endeavor presents background information on a previously described problem, the static airlift loading problem followed by a detailed description of the DALP. The explanation of this effort proceeds with a detailed description of the algorithm developed to solve it, the dynamic airlift loading problem-tabu search (DALP-TS), including mathematical comparisons to the current technique as well as a theoretical lower bound. The description of this methodology concludes with a brief summary and recommendations for future research.

2 Airlift loading: brief literature review

Previous research efforts have approached the airlift loading problem from different perspectives rendering meaningful algorithmic comparisons difficult. Feng et al. [12] present a detailed literature review of commercial air cargo operations. They note that "most problems, real-world problems in particular, remain unsatisfactorily solved, partly because of the complexities of air cargo operations" [12]. Most commercial airlift problems differ from military airlift problems in terms of efficiency and effectiveness. Commercial airlift cargo problems attempt to balance efficiency (i.e., generating revenue) and effectiveness (maintaining customer satisfaction); however, their clear goal is to earn a profit. Military airlift problems ensure effectiveness first (i.e., deliver needed goods and equipment when needed), while maintaining the greatest degree of efficiency as possible (i.e., wisely spending taxpayer's money). These different goals lead to varying approaches in formulating and solving commercial and military airlift problems.

The literature described herein is categorized as military specific research efforts, bin packing problems, pickup and delivery problems, and cost/revenue problems. Although many research efforts could be categorized in multiple categories, they are categorized herein by their main emphasis area.

2.1 Military loading research efforts

Cochard and Yost [7] developed the deployable mobility execution system for use by the US Air Force. Their model used a modified cutting stock heuristic which only generated *feasible* loads for the aircraft. An upgraded version of this algorithm, called Computer Aided Load Manifesting, became the US Air Force standard; however, it was subsequently replaced due to its inadequacies in generating solutions to large-scale airlift loading problems.

Ng presented a multicriteria optimization algorithm (using goal programming) for aircraft loading [29]. He demonstrated a 9 % reduction in required aircraft compared to traditional, manual methods for generating aircraft

loads [29]. His algorithm was limited to the "initial planning stages of an airlift exercise, when most of the data are not very accurate;" thereby limiting its applicability for an actual airlift loading problem [29].

The Air Force Studies and Analyses Agency (now designated as Headquarters Air Force/A9) contracted the development of the airlift loading model as a research and evaluation tool for analysis of loadability for military combat and support units on airlift [8]. This model determined the number of sorties required to move military units into and within a theater of operations.

Baker et al. [2] detailed an algorithm, called NPS/RAND Mobility Optimizer, to optimize the utility of military airlift in a deployment setting with time windows as well as analyze the US Air Force fleet modernization and allocation of resources. Their methodology expanded upon the airlift loading problem by including air refueling; however, it only loads cargo items according to weight [2]. The algorithm did not assign cargo items to specific positions within an aircraft; therefore, this technique is useful for estimating total aircraft required for planning and budgeting purposes but not for implementation in an actual deployment setting.

Gueret et al. [17] proposed a method for loading military aircraft (specifically, the French military) for airlift operations. They used a two-phased solution method consisting of heuristics that quickly compute "good" initial solutions and a local search algorithm to improve upon the initial solution, while ignoring the aircraft center of balance (CB). The research effort presented herein employs a similar construct, using an extension of the current methodology to quickly generate quality initial solutions and then a search heuristic to improve on this solution.

Hilliard et al. [18] described the airlift deployment analysis system which facilitated troop and cargo movement in support of operation DESERT STORM. The operation was a large-scale implementation of an airlift loading problem with time windows. Unfortunately, the authors only explained the algorithm in general terms and the transported goods in bulk; the algorithm does not assign cargo to specific locations within the aircraft.

2.2 Bin packing problems

Modeling the airlift loading problem as a 2-dimensional bin packing problem is an attractive but limited approach [4, 24, 32]. Heidelberg et al. [20] approached the airlift loading problem as a 2-dimensional bin packing problem (ignoring height constraints) using length and width of the cargo items and of the aircraft's cargo hold. They indicated that classical methods of bin packing are inadequate in aircraft loading because they ignore aircraft CB concerns and item spatial restrictions.

Mongeau and Bes [27] described an algorithm to optimize aircraft loading by treating the cargo holds as containers. They demonstrated the efficacy of the algorithm on Airbus aircraft; however, their algorithm does not incorporate cargo time windows. Thomas et al. [37] presented an algorithm used at Federal Express to pack cargo containers into Airbus aircraft. Their algorithm attempts to find a *feasible* (in terms of CB constraints) packing scheme for aircraft in which the cargo load has already been selected.

Other efforts have modeled cargo loading as a 3-dimensional bin packing problem [9, 30, 35]. Paquay et al. [30] discuss the problem of optimizing loading a set of strongly heterogeneous boxes into commercial aircraft containers with the goal of minimizing the unused volume within the container. They include center of gravity considerations in their formulation. While this concept could be applied to pallet buildup (maximizing the usable space under cargo netting), the pallet buildup step is excluded from consideration in this research effort since it is often outside the control of the strategic airlift personnel (i.e., airlift customers pack their own pallets). Roesener and Hall [35] formulated a 3-dimensional packing problem for constructing pallets for specific positions within aircraft.

2.3 Pickup and delivery problems

Pollaris et al. [31] present a review of vehicle routing problems, some of which include vehicle balance and/or pickup and delivery constraints. While focusing on ground-based vehicles, many of the concepts they discuss could easily apply to aircraft as well.

Solanki and Southworth [36] detailed an algorithm to create a schedule for moving cargo and personnel. They modeled the problem as a pickup and delivery problem with time windows and developed an insertion heuristic to build an airlift schedule by sequentially "adding movement requirements [i.e., cargo items or personnel] to the schedule one at a time" [36]. This technique does not adjust aircraft loads or the overall schedule after completion; it simply creates the best possible schedule (that may not be feasible) given the movement requirements.

Lurkin and Schyns [25] model an airline container loading problem with pickup and delivery. They consider both aircraft center of balance constraints as well as cargo temporal restrictions [25]. They show that this problem is NP-hard and therefore compare their results to current manually generated solutions vice optimal (or even "near" optimal) solutions [25].

2.4 Cost and revenue problems

Many other research efforts examine airlift loading from cost aspects [5, 22, 33, 41]. Reiman et al. [33] discuss

methods to airlift aircraft fuel efficiency through increased utilization of aircraft cargo capacity. Li et al. [22] present a compromised large-scale search neighborhood to aid freight forwarders in minimizing total freight costs given a limited number of rented containers.

Bookbinder et al. [5] present four exact solution methodologies for solving the air cargo consolidation problem under the pivot-weight scheme. In this problem, customer's individual unit load devices (ULDs) are charged a penalty cost if the loaded weight fails to fall within an acceptable range; this penalty incentivizes proper aircraft balancing by creating a financial advantage to the customer to adhere to these rules [5]. A similar construct is presented in Sects. 4.5.5 and 5.1 in application to this research effort: Aircraft are penalized if their allowable cabin load is outside of an acceptable range.

Vancroonenburg et al. [41] attempted to optimize profit in the selection of cargo for commercial transportation, while minimizing "the deviation between an aircraft's center of gravity and a known target value so as to reduce fuel consumption and improve stability" in flight. While these airlift cost and revenue problems advance aspects of airlift loading problems, none of these efforts include temporal restrictions in their problems.

2.5 Literature review summary

While the research efforts discussed in these four categories of airlift problems examined aspects of the DALP, none of them holistically consider the entire problem. The research effort described herein expands upon the current literature by combining the four interdependent subproblems of packing, partitioning, selecting, and placing cargo pallets. Additionally, this effort proposes a new heuristic based algorithm to solve the DALP and demonstrates the efficacy and efficiency of the algorithm in solving real-sized problem instances.

3 DALP problem description

3.1 General overview

The DALP involves assigning palletized cargo to specific pallet positions within available aircraft to minimize deviations from preferred aircraft load requirements while satisfying, to the maximum extent possible, the temporal constraints on the pallets. The temporal constraints define the available "window" of acceptable days within which each pallet should reach the destination. Although preferable for a pallet to arrive within its specified time window, DALP-TS considers solutions in which pallets arrive outside the desired window (either early or late). These

solutions present decision makers with courses of action (some of which will require a waiver to implement) from which they can select their preferred option. From an effectiveness standpoint, the DALP attempts to deliver all cargo items within the specified time frame; however, efficiency might preclude sending a strategic airlift aircraft with a single pallet.

The temporal constraints in the DALP enable aircraft to conduct multiple trips. The required travel time from the APOE to APOD, down time at the APOD (for cargo off-load, crew rest, and aircraft refueling), and travel time from APOD to APOE are known a priori. Each aircraft returns to the APOE (home station) from which it originated.

The overall DALP goal includes minimizing not only the number of flights required to transport all pallets, but also the total number of aircraft required, while at the same time minimizing aircraft allowable cabin load (ACL) violations and pallet temporal violations. An aircraft available for a DALP scenario but not required in the final solution can be utilized for other logistical transportation needs. ACL violations occur when the assigned cargo load weight surpasses the aircraft's ACL; pallet temporal violations occur when a pallet's arrival date is outside of its acceptable arrival window. By considering feasible and infeasible solutions (i.e., those with violations), senior level decision makers can choose a *preferred* solution from a set of solutions in order to balance the effectiveness and efficiency of airlift aircraft.

3.2 Contingency versus sustainment versus planning

The DALP describes the *contingency* or *deployment* phase of military involvement in an area, while [34] addressed the *sustaining* phase in the static airlift loading problem and [2] examined the *planning and budgeting* phase in their optimization of military airlift. The contingency phase considers the flow of items and equipment into a region in which the US military presence is not yet established or just beginning; the US Military has a preferred sequencing for the delivery of items. These can broadly be described as security items and equipment (e.g., weapons and ammunition), necessity items (e.g., shelter, water purification, and food), and lastly comfort items (e.g., cots and pillows). To ensure that items arrive as required (not early or late), proper sequencing is necessary. A Time-Phased Force Deployment Document (TPFDD) details earliest and latest arrival dates for items coinciding with the expected or actual progression of the operation.

3.3 Computational complexity

The general 2-dimensional bin packing problem can be defined as follows:

Given a set of M bins of capacity C and a set of N items of sizes $D_1, \ldots D_N$, minimize the number of bins required to pack all N items.

A Bin packing problem can be single, dual, or multi-dimensional (i.e., 1-dimensional, 2-dimensional, or $\geq$3-dimensional); [11] presents a general Bin packing problem formulation. The 2-dimensional bin packing problem, which [13] proved is NP-Hard, is a special case of the static airlift loading problem with the aircraft and knapsack constraints relaxed. Furthermore, the static airlift loading problem [34] is a special case of the DALP with relaxed temporal constraints. A polynomial reduction from the 2-dimensional Bin Packing Problem to the DALP is possible; thus, the DALP is assumed to be NP-Hard.

4 DALP-TS description

The following sections describe the DALP-TS algorithm developed to solve DALP instances. A complete description of general tabu search algorithms is beyond the scope of this effort; [15] and [16] present a detailed overview and provide background information on tabu search algorithms.

4.1 DALP-TS inputs

DALP-TS requires two pre-defined object sets which contain both physical and temporal data fields for the available aircraft and pallets. Since the TPFDD specifies the temporal dates as whole days (i.e., integers) from a starting reference, all dates for DALP-TS are also given in whole days. Table 1 presents a summary of these objects and their data fields which expand upon those presented developed in [34] by including temporal constraints.

The *aircraft input object* data fields describe the physical and temporal aspects of an aircraft. The aircraft identification number distinguishes individual aircraft, while the aircraft type details the particular type of aircraft (e.g., C-17 or C-5) of an aircraft object. The number of available pallet positions indicates the total number of pallets which an aircraft can hold. The ACL is the maximum total weight loadable into the aircraft; ACL depends upon the take-off environmental conditions, flight length, flight path

Table 1 DALP-TS inputs and their data fields

Aircraft	Identification number, aircraft type, number of available pallet positions, planning ACL, maximum ACL, CB lower bound, CB upper bound, optimal CB, ready to load date, required travel time, and required crew rest time
Pallet	Identification number, loaded weight, loaded height, available load date, latest arrival date, and required delivery date

Table 2 Aircraft type with planning and maximum ACLs

Aircraft type #	Description	Planning ACL (pounds)	Maximum ACL (pounds)
1	C-130 (6 pallets)	25,000	40,000
2	C-17 (logistics SYSTEM–18 pallets)	90,000	175,000
3	C-17 (air drop system– 11 pallets)	90,000	175,000
4	C-5 (36 pallets)	150,000	291,000
5	KC-10 (17 pallets)	80,000	150,000
6	KC-10 (23 pallets)	80,000	150,000
7	C-141 (13 pallets)	46,000	70,000
8	KC-135 (6 pallets)	30,000	40,000

conditions, landing environmental conditions, and available refueling aircraft. An aircraft type has two different ACLs: planning ACL and maximum ACL. The planning ACL is a *conservative guideline* for the aircraft's total load based on average flight conditions; this value is used by military planning personnel to determine a rough estimate for the number of aircraft required for an operation. The maximum ACL is the *structural load limitation* of an aircraft type; the maximum ACL is the highest value for ACL under "perfect" flight conditions. Attempting to fly an aircraft with a load exceeding, the maximum ACL will cause physical damage to the aircraft. Table 2 presents the US Air Force transportation aircraft types including their planning and maximum ACLs; these are the same aircraft examined in the static airlift loading problem by [34].

A loaded aircraft's CB is the distance (measured in inches) of the cargo load CB from an associated reference line. For an aircraft's longitudinal CB, the reference line is at or near the front of the aircraft (varies for each aircraft type); for aircraft with two pallet rows, the reference line for the lateral CB follows the center of the aircraft from nose to tail. An aircraft's CB is computed as the sum of the pallets' moments divided by the total weight of the cargo, where a pallet's moment is the product of the pallet weight and the distance from the associated reference line. For a given cargo load's total weight, each aircraft type stipulates a lower and upper bound CB which enables safe flight; additionally, each aircraft type specifies an optimal or target CB location for the best fuel consumption rate at a given cargo load's total weight. The ready to load date specifies the *earliest* date the aircraft is available for use. The required travel time indicates the amount of time required for the aircraft to travel from the APOE to APOD. The crew rest time specifies the amount of time required for the crew to rest at the APOD prior to returning to the APOE. Since the DALP's initial focus is to be effective first and efficient second, the aircraft set is sufficiently

large to transport all pallets (i.e., available aircraft is not a limitation in solving the DALP). Senior decision makers may modify the set of aircraft to coincide with their preferences.

The *pallet input object* presented in Table 1 contains three physical and four temporal data fields. The pallet object physical data fields detail an identification number and the height and weight of the loaded pallet which indicate whether a pallet feasibly fits into an aircraft's pallet position. Utilization of the military standard 463L pallet system removes the necessity of including length and width dimensions for a loaded pallet, which are $108'' \times 88''$. The pallet's available load date corresponds to the earliest date available for loading on an aircraft; as such, it is strictly enforced by DALP-TS. Violations of the remaining three temporal fields are allowed but penalized. Pallets reaching the APOD prior to the earliest arrival date may arrive before the necessary equipment and/or personnel required for cargo unloading are in place and may also cause port congestion issues (i.e., exceeding airfield storage capacity) at the APOD. The pallet latest arrival date and pallet required delivery date are very similar and often identical. The latest arrival date is the latest date by which the pallet *should* arrive at the APOD to enable proper unloading and assembly; the required delivery date is the latest date by which the pallet *must* arrive at the APOD prior to the deploying force departing the APOD.

4.2 DALP-TS objects

While exploring the solution space, DALP-TS creates two sets of objects: an *aircraft trip* corresponding to each APOE/APOD round trip and a *solution* object corresponding to each solution visited. Table 3 lists these objects and their data fields which expand upon those developed in [34] by including the temporal constraints. Note that the aircraft trip object only changes when a tabu search move modifies the aircraft's load; the solution object changes for each new solution examined.

The aircraft trip object augments the fields of the aircraft input object with data fields relating to the cargo load, objective function contribution, and trip temporal information. The trip number indexes a specific aircraft trip and

Table 3 DALP-TS objects and their data fields

Aircraft trip	Trip index, loaded cargo weight, number of loaded pallets, longitudinal CB, lateral CB, objective function weight usage, objective function space usage, objective function longitudinal CB, objective function lateral CB, ready to load date, departure date, arrival date, and trip number
Solution	Number of aircraft used, number of flights flown, solution array, and objective function value

details the total number of flights flown by each aircraft. The loaded cargo weight and number of loaded pallets details the cargo assigned to the aircraft; the longitudinal and lateral CBs correspond to these values for the given cargo load. To accelerate solution updating, DALP-TS maintains aspects of the aircraft trip's contribution to the objective function: weight usage, space usage, longitudinal CB, and lateral CB.

The aircraft ready to load date is the *earliest* date upon which an aircraft can be loaded and depart the APOE. After selecting a pallet cargo load, DALP-TS computes an appropriate aircraft departure date *from the APOE* that considers pallet available to load date. The aircraft arrival date describes the date upon which the aircraft arrives *at the APOD*, incorporating the departure date and the required travel time from APOE to APOD. The aircraft departure date *can* be the same as the ready to load date; however, delaying the departure date to a later date may prevent or reduce the amount of pallet temporal violations. The ready to load date for each aircraft's first trip is day 1. The ready to load date of each subsequent trip for a particular aircraft considers the previous trip's departure date and the travel, down, and return time.

The *solution* object details the required aspect of a solution. The number of aircraft used and number of flights flown are simply counts for aircraft and trips utilized in a solution. The solution array indicates the assignment of pallets to aircraft trips, while the objective function value denotes a single value for solution comparison.

4.3 DALP-TS solution array

The DALP-TS solution representation assigns pallets to specific positions in available aircraft. By the US Air Force designation, each aircraft's pallet position has a specific reference number. In aircraft with a single row of pallet positions, the numbering is sequential from nose to tail; in aircraft with parallel rows of pallet positions, the pallet position designation is 1L, 1R, 2L, 2R, etc. For example, in a C-17 aircraft in the logistics configuration, 18 pallet positions exist; Fig. 2 graphically depicts this configuration. The pallet position closest to the nose of the aircraft on the left (port) side is designated 1L, and the pallet adjacent to it is 1R; the pallet positions at the tail of the aircraft are 9L and 9R. DALP-TS uses the US Air Force designation for single row aircraft and a modified version for double row aircraft (i.e., the designation changes from 1L, 1R, 2L, …, 9R to 1, 2, 3, …, 18).

The structure used in the DALP-TS solution representation prepends the aircraft's index and trip index to the array; this is an expansion of the structure used in [34] with the addition of the trip index. As a clarifying example, consider two C-130 aircraft available to transport 15

pallets, with indices given by: 1, 2, …, 15. A possible solution for this problem is:

$$(1, 1, 4, 5, 6, 1, 2, 3)(2, 1, 7, 8, 9, 0, 10, 11)$$
$$(2, 2, 12\ 13, 0, 0, 14, 15)(1, 2, 0, 0, 0, 0, 0, 0)(3, 0)$$

In each of the first four parenthetical subsets, the initial two numbers correspond to the aircraft's index and trip number, respectively; the remaining numbers indicate loaded pallet indices. A pallet index of zero corresponds to an empty pallet position. In this example problem, aircraft 1, trip 1 has pallet 4 in position 1, pallet 5 in position 2, etc. Aircraft 1, trip 2 has no pallets loaded. The final parenthetical subset represents a "storage aircraft" (with capacity equivalent to the total number of pallets) that is used during neighborhood searches. This concept is analogous to the "Big Bin" previously used in tabu search algorithms; in reality, it corresponds to an APOE's hangars or tarmac storage area [19]. The index for this "storage aircraft" is one greater than the total number of aircraft without a trip index. In this example, the *0* in *(3, 0)* indicates that this "aircraft" is empty (i.e., all pallets are loaded on aircraft).

4.4 DALP-TS initial solution generator

DALP-TS attempts to efficiently produce a quality initial solution by inserting pallets into aircraft trips with arrival dates within the pallet's acceptable window and with sufficient remaining space and weight capacity. At the time of this research, no software or methodology was used by the Department of Defense to generate solutions to a DALP; however, a software program called the automated air load planning system (AALPS) was used to solve static airlift loading problem instances (i.e., those problems without temporal restrictions). As a result, DALP-TS utilizes an analogous extension of the AALPS loading procedure to generate an initial solution to the DALP.

To generate an initial solution, DALP-TS sequentially sorts the pallets by increasing available load date, earliest arrival date, latest arrival date, and required delivery date, and then by decreasing weight, at this point, all pallets are theoretically "loaded" in the "storage aircraft." The DALP-TS initial solution generator selects the pallet at the head of the sorted pallet list. Next, the best suitable aircraft is selected. Three states of *feasible* (i.e., no temporal or ACL violations) aircraft trip availability exist for every pallet: (1) At least one non-empty aircraft (i.e., with one or more pallets loaded) is scheduled to be at the APOE with feasible temporal constraints and sufficient weight and space capacity remaining to support the pallet. In this case, DALP-TS selects the aircraft with the largest ratio of ACL to number of pallet positions and the smallest weight capacity remaining. (2) An available aircraft has returned from a trip and has a ready to load date which enables the pallet to arrive

on or before its latest arrival date. In state 2, DALP-TS the aircraft with the largest ratio of ACL to number of pallet positions and the largest trip index. (3) No additional aircraft trips satisfy the pallet's temporal window requirements. For state 3, DALP-TS utilizes the next unused aircraft from the user-provided list of available aircraft.

After selecting the aircraft, DALP-TS removes the pallet from the "storage aircraft" and assigns it to the first available position in the aircraft. If the aircraft's trip is in state 1, no changes to the aircraft departure date or solution object are necessary; however, updates are required for states 2 or 3. First, DALP-TS increments the number of flights flown in the solution object. DALP-TS also updates the aircraft's departure date and arrival date to reflect the new temporal constraints. DALP-TS sets the aircraft arrival date identical to the pallet latest arrival date and adjusts the aircraft's departure date to incorporate travel time; this also causes the insertion of an additional aircraft trip in the solution object. The ready to load date of this trip incorporates the previous trip's departure date and total travel time. For this trip, the aircraft is empty but available for future loading. If the initial solution generator is in state 3, the aircraft is being used for the first time; the aircraft departure date is updated such that the aircraft arrival date corresponds with the pallet's earliest arrival date.

The DALP-TS initial solution generation process ignores CB constraints; it only considers weight and temporal constraints. DALP-TS adjusts the lateral and longitudinal CB as the first step in its dynamic neighborhood selection process.

While feasibility is not an initial solution requirement for TS, the DALP-TS initial solution generator produces a feasible solution (with respect to ACL and temporal violations) for a given set of pallets and aircraft. The proper assignment of a pallet to an aircraft trip usually yields quality (i.e., fewer aircraft and aircraft trips required) initial solutions. Selecting the best arrival date (and correspondingly the departure date) for an aircraft trip receiving its first pallet assignment is imperative in producing quality initial solutions.

4.5 DALP-TS formulation

The DALP-TS formulation is a relaxation of constraints into the objective function subject to integrality restrictions on the decision variables. The objective function to be minimized is a combination of penalties and usage fees. The components of the objective function are: (1) aircraft usage fee, (2) aircraft load penalty, (3) lateral center of balance penalty, (4) longitudinal center of balance penalty, and (5) temporal violation penalty. These penalties consider $i = 1,…,I$ pallets and $j = 1,…,J$ available aircraft which can make up to $k = 1,…,K$ trips.

4.5.1 Aircraft usage fee

DALP-TS incorporates a usage fee, which varies with aircraft type, for each aircraft trip employed. DALP-TS charges an *increased* usage fee for the first aircraft trip; therefore, it is preferred to use an additional aircraft trip than to use an aircraft for the first time. One goal of DALP-TS is to transport all pallets with both the fewest trips *and* the fewest aircraft employed.

Equation (1) presents the computation of this penalty:

$$\sum_{j=1}^{J+1}\sum_{k=1}^{K} C_{jk}A_{jk} \tag{1}$$
$$A_{jk} \in \{0,1\} \quad \forall j = 1,\ldots,J+1; \quad k = 1,\ldots,K$$

where C_{jk} is the usage fee (cost) associated with trip k of aircraft j; A_{jk} is 1 if trip k of aircraft j is used and 0 otherwise; $J + 1$ is the number of aircraft available (including the "storage aircraft"). The cost of using the storage aircraft is significantly higher than that for any aircraft trip, denoted by $C_{(J+1)1} - C_{jk}\forall j = 1, \ldots, J; k = 1, \ldots, K$, thereby driving the search to regions where the storage aircraft is empty. Additionally, the cost of using an aircraft for the first time is higher than using an additional aircraft trip, denoted by $C_{j1} \gg C_{jk-}$ $\forall j = 1, \ldots, J; k = 2, \ldots, K$, thereby enabling DALP-TS to encourage the use of fewer aircraft.

4.5.2 Aircraft load penalty

DALP-TS uses the aircraft load penalty to encourage maximum usage of the aircraft's planning ACL. Because the planning ACL represents a soft constraint, DALP-TS evaluates aircraft loads which exceed this value; however, a penalty applies to this violation. To enable a broad search of the solution space, DALP-TS explores, but does not save, solutions which exceed the maximum ACL. The computation for these loaded aircraft penalties is:

$$\sum_{j=1}^{J}\sum_{k=1}^{K}\left(\frac{\left(\mathrm{ACL}_j - \sum_{i=1}^{n} W_{ijk}\right)}{\mathrm{ACL}_j}\right)X_{jk}$$
$$+ \sum_{j=1}^{J}\sum_{k=1}^{K}\left(\frac{\left(\mathrm{ACL}_j - \sum_{i=1}^{n} W_{ijk}\right)}{\mathrm{ACL}_j}\right)(1 - X_{jk}) \tag{2}$$
$$X_{jk} \in \{0,1\} \quad \forall j = 1,\ldots,J; \quad k = 1,\ldots,K$$

where ACL_j is the planning ACL for aircraft j; W_{ijk} is the weight of pallet i loaded on trip k of aircraft j; and X_{jk} is 1 if the total pallet weight on trip k of aircraft j is less than its planning ACL. Although these equations will provide an identical value with either $X_{jk} = 0$ or 1, they have different scaling factors in the final version of the objective function, which is detailed in Sect. 4.5.5.

4.5.3 Lateral and longitudinal center of balance penalties

The lateral and longitudinal CB penalties force DALP-TS to explore areas of the solution space with preferred assignments of pallets to specific positions within an aircraft. DALP-TS computes an aircraft's CB as the sum of the pallets' moments divided by the total weight of the cargo, where a pallet's moment is the product of the pallet weight and the distance from an associated reference line. DALP-TS only computes the lateral CB for aircraft with two rows of pallets; for these aircraft, the reference line traverses the center of the aircraft from nose to tail. The reference line for the longitudinal CB differs for all aircraft types; however, for most aircraft, it is at or near the nose of the aircraft. The general CB computation for the kth trip of aircraft j is:

$$\mathrm{CB}_{jk} = \frac{\sum_{i=1}^{n}\left(W_{ijk} * D_{ij}\right)}{\sum_{i=1}^{n}\left(W_{ijk}\right)} \tag{3}$$

where n is the total number of pallets loaded on trip k of aircraft j; W_{ijk} is the weight of pallet i loaded on trip k of aircraft j; and D_{ij} is the distance of pallet i from the associated reference line on aircraft j.

DALP-TS computes a CB penalty based on the squared difference between the aircraft's target CB value (which produces the best fuel consumption rate) for the given cargo weight and the aircraft's loaded CB value. DALP-TS computes the lateral CB penalty as a deviation from the aircraft's centerline:

$$\sum_{j=1}^{J}\sum_{k=1}^{K}\left(\frac{\sum_{i=1}^{n}\left(W_{ijk} * D_{i_{(\mathrm{Lat})}j}\right)}{\sum_{i=1}^{n}\left(W_{ijk}\right)}\right)^{2}A_{jk} \tag{4}$$
$$A_{jk} \in \{0,1\} \quad \forall j = 1,\ldots,J; \quad k = 1,\ldots,K$$

where n is the total number of pallets loaded on trip k of aircraft j; W_{ijk} is the weight of pallet i loaded on trip k of aircraft j; $D_{i_{(\mathrm{Lat})}j}$ is the distance of pallet i from the associated reference line on aircraft j (i.e., the aircraft's centerline); and A_{jk} is 1 if trip k of aircraft j is used and 0 otherwise. Note that for the aircraft considered herein that utilize the 463L pallet, $D_{i_{(\mathrm{Lat})}j}$ is constant for a specific aircraft type j; the value will be negative for pallets to the left of centerline and positive for those on the right.

The longitudinal CB is computed from the reference datum line, which is at or near the nose of the aircraft and varies for different types of aircraft. The longitudinal CB computation considers situations where the loaded aircraft CB is within and outside the upper and/or lower bound values. Because the algorithm does not constrain the search to feasible solutions and prohibits returning to previous solutions for a specified number of iterations, DALP-TS explores larger portions of the solution space, thereby

avoiding traps of local optimal solutions; however, solutions that are outside the acceptable range produce dangerous flight conditions and are not saved. The longitudinal CB computation is:

$$\sum_{j=1}^{J}\sum_{k=1}^{K}\left(\text{TCB}_j - \frac{\sum_{i=1}^{n}\left(W_{ijk} * D_{i_{(\text{Long})}j}\right)}{\sum_{i=1}^{n}\left(W_{ijk}\right)}\right)^2 A_{jk}Y_{jk}$$

$$+ \sum_{j=1}^{J}\sum_{k=1}^{K}\left(\text{TCB}_j - \frac{\sum_{i=1}^{n}\left(W_{ijk} * D_{i_{(\text{Long})}j}\right)}{\sum_{i=1}^{n}\left(W_{ijk}\right)}\right)^2 A_{jk}\left(1 - Y_{jk}\right)$$

$$A_{jk}, Y_{jk} \in \{0,1\} \quad \forall j = 1,\ldots,J; \quad k = 1,\ldots,K$$

$$(5)$$

where n is the total number of pallets loaded on trip k of aircraft j; TCB_j is the target center of balance point for the given cargo load (this value indicates the CB at which an aircraft experiences the best fuel consumption rate and varies by aircraft type and loaded cargo weight); W_{ijk} is the weight of pallet i loaded on trip k of aircraft j; $D_{i_{(\text{Long})}j}$ is the longitudinal distance of pallet i from the associated reference line of aircraft j; A_{jk} is 1 if trip k of aircraft j is used and 0 otherwise; and Y_{jk} is 1 if the aircraft's CB is within the upper and lower CB bounds and 0 otherwise. Note that the two parts of this equation will have the same value if $Y_{jk} = 1$ or 0; the contribution to the objective function varies according to their associated penalty multiplier, which reflects their relative importance and is explained in Sect. 4.5.5.

Figure 1 presents the US Department of Defense form utilized by load planners to create aircraft load plans and perform hand-calculations on CB on C-5 aircraft [38], while Fig. 2 illustrates the form used for C-17 load plans and CB calculations [39]. These forms present the distance (in inches) of the center position for each pallet from the reference data line (i.e., $D_{i_{(\text{Long})}j}$); these values are indicated below the black arrows that are above the aircraft diagram. These forms also illustrate the complexity required to manually configure an airlift aircraft flight; the second sheet for each form (not included) allows for the manifesting of additional cargo items. Furthermore, the target center of balance specification is not explicitly stated on these forms; however, this value is included in the technical manuals utilized by loadmasters in training for and maintaining their air load planner status. Figure 2 demonstrates that two configurations exist for pallets within the C-17: the center loaded 11 pallet positions correspond to the Airdrop System (ADS), while the two column 18 pallet positions align with the logistics configuration. As Fig. 1 indicates, the

Fig. 1 DD Form 2130-1, C-5 load plan (w/cargo pallet positions) [38]

Fig. 2 DD Form 2130-13, C-17 load plan (w/cargo pallet positions) [39]

C-5 aircraft is not rated to conduct airdrop missions, so it only contains a logistics configuration.

4.5.4 Temporal violation penalty

Each individual pallet has an acceptable window of arrival dates and a strict lower bound on its departure date. It is preferable for a pallet to arrive after its earliest arrival date and before its latest arrival date; no specific date within the acceptable window is preferable to any other. DALP-TS allows exploration of solutions with arrival dates outside the acceptable region by applying penalties to these violations. The approach used by [21] and [26] for temporal violations involved penalizing the objective function by a product of the number of days the item arrives after the required delivery date and the item's weight. DALP-TS utilizes a similar approach by penalizing both early and late arrivals; however, rather than utilizing the required delivery date, DALP-TS considers the more restrictive latest arrival date. Pallets arriving prior to their earliest arrival date incur a penalty equal to the difference between the earliest arrival date and arrival date multiplied by the pallet weight and a scaling factor, which is detailed in Sect. 4.5.5. Likewise, pallets arriving after their latest arrival date incur a penalty equal to the difference between the arrival date

and latest arrival date multiplied by the pallet weight and a scaling factor.

The computation for penalty is:

$$\sum_{k=1}^{K}\sum_{j=1}^{J}\sum_{i=1}^{I}(t_{\text{EAD}} - t_{\text{AD}})W_{ijk}A_{jk}Z_{ijk}^1$$
$$+ \sum_{k=1}^{K}\sum_{j=1}^{J}\sum_{i=1}^{I}(t_{\text{AD}} - t_{\text{LAD}})W_{ijk}A_{jk}Z_{ijk}^2$$
$$A_{jk} \in \{0,1\} \quad \forall j = 1,\ldots,J; \quad k = 1,\ldots,K$$
$$Z_{ijk}^1, Z_{ijk}^2 \in \{0,1\} \quad \forall i = 1,\ldots,I; \quad j = 1,\ldots,J; \quad k = 1,\ldots,K$$

$$(6)$$

where t_{EAD} and t_{LAD} are the pallet's earliest and latest arrival dates, respectively; t_{AD} is the pallet's actual arrival date; A_{jk} is 1 if aircraft j flies trip k, 0 otherwise; Z_{ijk}^1 is 1 if the arrival date of trip k of aircraft j is before pallet i's earliest arrival date, 0 otherwise; and Z_{ijk}^2 is 1 if the arrival date of trip k of aircraft j is after pallet i's latest arrival date, 0 otherwise.

4.5.5 DALP-TS formulation

The DALP-TS formulation additively combines the previously described penalties as constraints relaxed into the objective function to be minimized. Each constraint relaxation is multiplied by a scaling factor (λ) which

ensures that each fee or penalty receives the appropriate level of consideration. Since DALP-TS is a decision-making aid, the determination of the scaling factor values enables decision maker inputs prior to initiating the algorithm. Thus, the scaling factors are not fixed values but rather relate the relative importance of the penalties: a relatively higher scaling factor indicates a penalty (i.e., an allowable violation) that is more important relative to the other penalties. As an example, a decision maker may decide that airlift effectiveness (i.e., on time delivery) is more important than efficiency (i.e., minimizing the required number and fully utilizing all aircraft); the scaling factor associated with Eq. (6), cargo temporal violations would be increased relative to Eqs. (1) and (2), aircraft usage and aircraft loading, respectively. Combining these concepts with the previously defined equations, the complete DALP-TS formulation is:

$$\min f\left(A_{jk}, X_{jk}, Y_{jk}, Z_{ijk}^1, Z_{ijk}^2\right)$$

$$+ \lambda_2 \sum_{j=1}^{J} \sum_{k=1}^{K} \left(\frac{\left(\mathrm{ACL}_j - \sum_{i=1}^{n} W_{ijk}\right)}{\mathrm{ACL}_j}\right) X_{jk}$$

$$+ \lambda_3 \sum_{j=1}^{J} \sum_{k=1}^{K} \left(\frac{\left(\mathrm{ACL}_j - \sum_{i=1}^{n} W_{ijk}\right)}{\mathrm{ACL}_j}\right)(1 - X_{jk})$$

$$+ \lambda_4 \sum_{j=1}^{J} \sum_{k=1}^{K} \left(\frac{\sum_{i=1}^{I}\left(W_{ijk} * D_{i_{\mathrm{Lat}}}\right)}{\sum_{i=1}^{I}\left(W_{ijk}\right)}\right) A_{jk}$$

$$+ \lambda_5 \sum_{j=1}^{J} \sum_{k=1}^{K} \left(\mathrm{TCB}_j - \frac{\sum_{i=1}^{I}\left(W_{ijk} * D_{i_{(\mathrm{Long})j}}\right)}{\sum_{i=1}^{I}\left(W_{ijk}\right)}\right)^2 A_{jk} Y_{jk}$$

$$+ \lambda_6 \sum_{j=1}^{J} \sum_{k=1}^{K} \left(\mathrm{TCB}_j - \frac{\sum_{i=1}^{I}\left(W_{ijk} * D_{i_{(\mathrm{Long})j}}\right)}{\sum_{i=1}^{I}\left(W_{ijk}\right)}\right)^2 A_{jk}(1 - Y_{jk})$$

$$+ \lambda_7 \sum_{k=1}^{K} \sum_{j=1}^{J} \sum_{i=1}^{I} (t_{\mathrm{EAD}} - t_{\mathrm{AD}}) W_{ijk} A_{jk} Z_{ijk}^1$$

$$+ \lambda_8 \sum_{k=1}^{K} \sum_{j=1}^{J} \sum_{i=1}^{I} (t_{\mathrm{AD}} - t_{\mathrm{LAD}}) W_{ijk} A_{jk} Z_{ijk}^2$$

Subject to:

$$A_{jk}, X_{jk}, Y_{jk}, \in \{0,1\} \quad \forall j = 1, \ldots, J; \quad k = 1, \ldots, K$$
$$Z_{ijk}^1, Z_{ijk}^2 \in \{0,1\} \quad \forall i = 1, \ldots, I; \quad j = 1, \ldots, J; \quad k = 1, \ldots, K$$

Several aspects of this formulation are nonlinear, thereby enhancing the selection of a heuristic methodology (i.e., tabu search) in determining high quality solutions. Minimizing the objective function encourages DALP-TS to search for solutions utilizing a minimal number of aircraft and aircraft trips, with pallets arriving within their acceptable arrival window and ideally positioned within

the aircraft. To accomplish this, DALP-TS uses three search neighborhoods to locate quality solutions near the current solution.

4.6 Search neighborhoods

DALP-TS utilizes three search neighborhoods to traverse the solution space: (1) unload entire aircraft, (2) intra-aircraft pallet swap, and (3) inter-aircraft pallet swap. DALP-TS calls each neighborhood under specific strategic circumstances.

The *unload entire aircraft* neighborhood removes pallets from a selected aircraft trip and strategically inserts them onto non-empty aircraft trips. This neighborhood swaps all pallets from a single aircraft with empty positions on other aircraft, resulting in an unloaded aircraft. DALP-TS transfers the pallets in order of decreasing weight into non-empty aircraft with at least one empty pallet position *and* an acceptable arrival date. If multiple aircraft contain acceptable arrival dates, DALP-TS selects the aircraft with the greatest weight capacity remaining. If multiple pallet positions are empty in an aircraft, DALP-TS assigns the pallet to the empty position closest to the front of the aircraft. For this move neighborhood, DALP-TS does not consider aircraft CB; the algorithm utilizes the two other move neighborhoods to correct violations of aircraft CB. If no aircraft has an acceptable arrival window, DALP-TS chooses the aircraft with the smallest amount of pallet temporal violation penalty. The unloaded aircraft trip is no longer considered available for pallet transportation. This neighborhood endeavors to decrease the total number of trips required.

The *intra-aircraft pallet swap* neighborhood evaluates every possible two-tuple (swapping two pallets or inserting a pallet into an empty position) *within* a single aircraft; this follows the structure developed by [34]. DALP-TS ignores swaps of pallets with identical weights and temporal constraints due to the null objective function effect. DALP-TS selects the best (in terms of objective function value) non-tabu move. Unfortunately, the best move could worsen (i.e., increase) this value. The lowest indexed pallet may not return to its previous position for a specified number (i.e., *tabu tenure*) of search iterations.

The *inter-aircraft pallet swap* neighborhood evaluates moves between two non-empty aircraft. DALP-TS evaluates every possible two-tuple *between* two aircraft, ignoring swaps of pallets with identical weights and temporal constraints. DALP-TS chooses the best non-tabu move and prevents returning a pallet to its donor aircraft for tabu tenure search iterations.

Numerous changes to an aircraft trip's temporal information results from the unload entire aircraft and inter-aircraft pallet swap neighborhoods. First, an aircraft trip

departure date might require modification to minimize pallet temporal violations. If the current departure date of a trip is later than the ready to load date, DALP-TS can decrease the departure date; however, the algorithm does not schedule an aircraft trip's departure date prior to its ready to load date. Decreasing a departure date only occurs if doing so improves the solution. Changing the departure date to a date which causes infeasibilities in the majority of the loaded pallets is unproductive. Decreasing the departure date of an aircraft trip creates a series of ready to load date decreases in subsequent aircraft trips; DALP-TS modifies the departure date of the succeeding aircraft trips if doing so improves the overall objective function value. Note that the trip ready to load date is the earliest date upon which an aircraft *may* depart; this date does not necessarily correspond to the best departure date for a given pallet load. By increasing a trip's departure date, DALP-TS could trigger a series of updates to the subsequent aircraft trips. This occurs when a trip's altered departure date causes the ready to load date of the succeeding trip to be *after* its departure date; DALP-TS then equates the departure date with the ready to load date.

DALP-TS utilizes these search neighborhoods to traverse both the feasible and infeasible solution space. The algorithm incorporates varying levels of tabu tenure and a tabu memory structure to escape local optimal solutions thereby exploring diverse solutions.

4.7 DALP-TS tabu memory structure

DALP-TS utilizes a tabu memory structure similar to that developed by [34] to guide the search to differing neighborhoods in the solution space by prohibiting the return to a solution with certain attributes for a number of iterations. The DALP-TS tabu memory structure is a 3-dimensional array of integers of size:

$$(\text{Number of pallets} \times \text{Max number of aircraft pallet positions} \times (\text{number of aircraft trips} + 1))$$

The value *Max number of pallet positions* indicates the largest total number of pallet positions in any of the available aircraft. The "*+1*" on "*number of aircraft trips +1*" accounts for the previously described storage aircraft.

After DALP-TS performs a move previously described in the search neighborhood section, returning the pallet(s) to its/their previous position on the aircraft trip is made tabu for a certain number of iterations specified by the *tabu tenure*. As previous research indicates, an adaptive tabu search mechanism improves search effectiveness [19]. Thus, DALP-TS adaptively modifies the tabu tenure value according to the level of objective function's improvement

resulting from the neighborhood move. A major improving move applies a unit decrease in the tabu tenure value, while a dis-improving move results in a unit increase in this value. If the improving move provides an objective function improvement below a predetermined minimum threshold level, no change is applied to the tabu tenure. Additional details on the levels of objective function improvements are presented in the next section.

The iteration count (i.e., tabu tenure + current iteration value) at which a return move is allowed is inserted into the tabu memory structure array in the cell corresponding to the pallet's identification number, pallet position number within the aircraft, and aircraft trip number. Return moves are not allowed until the iteration count reaches this value. This helps DALP-TS to avoid local optimal by forcing the search to progress away from a specific solution.

4.8 DALP-TS stopping criteria

DALP-TS first generates an initial solution from the given pallets and aircraft. Since the initial solution generator ignores CB, DALP-TS performs a series of CB improving moves within an aircraft using the *intra-aircraft swap* neighborhood; however, the algorithm halts after reaching a predetermined maximum number of these iterations (Max Fix). DALP-TS then initiates the dynamic neighborhood selection portion of the search. Limiting computation time is not a novel concept as demonstrated by [25]. Many companies determine that it is not fiscally productive to expend the additional time required to find the best possible solution; the small amount of cost savings does not justify the delay in operations.

DALP-TS categorizes the intra-aircraft swap moves as major improving, minor improving or non-improving. A major improving move decreases the overall objective function value by at least a predetermined improvement percentage, while a minor improving move decreases the overall objective function value less than or equal to this value. A non-improving move increases the objective function value.

DALP-TS utilizes two counters (both initialized to zero) in the neighborhood selection process: *non-improving move count* and *minor improving move count*. DALP-TS increments the non-improving move count value after selecting a non-improving move and resets the value to zero after selecting either type of improving move. Likewise, DALP-TS increments the minor improving move count value after selecting a minor improving move and resets the value to zero after selecting a major improving move; however, after selecting a non-improving move, the minor improving move count *does not change*. This logic

prevents DALP-TS from indefinitely cycling between non-improving and minor improving moves. DALP-TS halts when either of these counters reach a predetermined cessation value (maximum value).

Due to large memory requirements, cycling detection is not explicitly included in DALP-TS; however, cycling is prevented by use of the tabu memory structure, Adaptive tabu tenure, unload entire aircraft neighborhood, and the non-improving and minor improving move counters. The rationale for this choice is that the tabu memory structure is based on an individual pallet being placed upon a specific pallet location within a specific aircraft on a specific trip. The relatively large number of pallets undergoing placement and the refined granularity of the adaptive tabu tenure precludes the necessity of explicit cycling detection. Additionally, as is indicated in Sect. 4.9.2, DALP-TS incorporates the unload entire aircraft neighborhood after the occurrence of a pre-specified number of minor or non-improving moves, both of which are less than the initial tabu tenure value. Thus, if the search has stabilized in an unproductive area of the search region, DALP-TS will unload an aircraft before it will allow a pallet to return to its previous position.

4.9 Local search procedure

DALP-TS strategically employs the three previously described search neighborhoods during the algorithmic progression. DALP-TS incorporates a dynamic neighborhood selection process to select the most appropriate neighborhood given one of three algorithmic search states, using predetermined Max fix, Max minor improving move, and Max non-improving move values:

1. (Aircraft CB violation) $\cap$ (Fix CB count $\leq$ Max fix value)
2. (Minor improving move count $=$ Max minor move value) $\cup$ (non-improving move count $=$ Max non-improving move value)
3. Neither case 1 or 2 applies.

4.9.1 Search state 1

An aircraft longitudinal CB window violation often can be corrected with the *Intra-Aircraft Swap* neighborhood. There are three ways to escape state 1: (1) the CB window is satisfied, (2) the CB window is unsatisfied for a predetermined number of iterations (Max Fix Value) which indicates that timely progress is not present, or (3) the CB window violation worsens. These moves result in relatively small changes to the objective function value by changing the CB of a *single aircraft*.

4.9.2 Search state 2

A state 2 situation invokes the *unload entire aircraft* neighborhood in the following circumstances:

1. If minor improving moves proceed for Max minor improving move value iterations, DALP-TS has stabilized in a local plateau. Unloading an entire aircraft constitutes a diversification strategy which is an effort to navigate toward a region with better solutions.
2. If DALP-TS produces Max non-improving move value consecutive non-improving moves, diversification aids in escaping from a nonproductive region of the solution space.

Because DALP-TS imposes temporal and weight restrictions upon aircraft trips, the algorithm does not consider unloading lightly loaded aircraft; temporal constraints may necessitate such a loading schema. "Lightly loaded" aircraft are those with a cargo weight less than 25 % of the ACL. This search neighborhood selects the lightest loaded aircraft that has a cargo load greater than 25 % of the ACL and results in relatively large changes to the objective function value by removing an entire aircraft from future consideration.

4.9.3 Search state 3

If neither state 1 nor state 2 applies, DALP-TS uses the *Inter-Aircraft Swap* neighborhood, resulting in mid-sized objective function value changes.

4.10 Solution output

DALP-TS is a *decision-making aid* and, as such, allows decision makers to select a preferred solution for a given situation. Senior military and civilian leaders evaluate the risks and rewards associated with different options and make decisions based on their knowledge of the overall situation; thus, they receive multiple options, called "courses of action," from which they choose their preferred solution. To accommodate this selection system, DALP-TS reports up to four solutions which fall into four group types: (1) Feasible, (2) ACL Violations, (3) Temporal Violations, and (4) Both ACL and Temporal Violations.

If DALP-TS encounters solutions from a specific solution type, the algorithm stores the best solution (i.e., a solution with a lower objective function value will replace the current best solution). The first solution type, "Feasible," satisfies all constraints—there are no ACL violations and all pallets have an arrival date that is after the earliest arrival date but before the latest arrival date. The next three solution types are technically infeasible, but the benefits of these violations may enable the decision makers to choose

them over a strictly feasible solution. The second solution type allows the total loaded pallet weight of one or more aircraft to exceed the aircraft's planning ACL by no more than a predetermined ACL percentage (Planning ACL Max Violation). The third solution type includes those solutions which have violations in at least one pallet arrival date (either early or late). The last solution type encompasses solutions which both ACL violations *and* pallet temporal violations at the previously described levels.

5 Computational results

DALP-TS demonstrates its efficacy against 100 distinct problem instances for each of six realistic scenarios. At the time of this research, no previously developed baseline method or solution methodology to the DALP existed. Thus, the DALP-TS results are compared against a computed lower bound on the number of aircraft trips required. The technique of comparing computed results to non-optimal solutions is not without precedent in airlift loading; [23, 25, 29], and [41] compared their results to manually computed values from a human subject matter expert (i.e., "loadmaster") rather than computed optimal solutions. The DALP-TS initial solution extends the greedy algorithm used by AALPS to include temporal constraints. Thus, any reduction in the required number of trips from the initial solution to the final reported solution is an improvement over the current methodology.

5.1 Constant values

Prior to executing DALP-TS on the six scenarios, extensive testing focused on generating quality solutions across the solution spaces for a variety of problem instances produced the scaling factor and constant values utilized by DALP-TS. To accomplish this testing, values for the penalty factors, usage fees, and the search/move values were varied. The goal of this testing was to determine the levels which produced the best results in terms of objective function value and required aircraft trips for the 100 instances of the six problem scenarios. Since DALP-TS is a decision-making aid, the scaling factors and constant values are designed such that they can be adjusted as additional senior leader inputs regarding the relative importance of constraints are received.

The resulting values for the *scaling factors* are:

$\lambda_1 = 1; \lambda_2 = 1; \lambda_3 = 30; \lambda_4 = 1; \lambda_5 = 1; \lambda_6 = 800; \lambda_7 = 1; \lambda_8 = 1$

In the absence of actual decision maker inputs for their relative importance, the majority of the scaling factors

were fixed to a value of 1 to allow each penalty to have a similar impact on the objective function value. The only values not initialized to 1 are those for aircraft overloading ($\lambda_3 = 30$) and aircraft longitudinal CB penalty ($\lambda_6 = 800$). Although DALP-TS searches through solutions with cargo load total weights exceeding the Planning ACL values, these solutions are not the preferred type of solutions. Thus, there is a larger scaling factor associated with this penalty. An aircraft trip with a longitudinal CB that is outside the allowable limits is not properly balanced for stable flight. As a result, DALP-TS has a very large scaling factor associated with this penalty function. DALP-TS explores violations of the remaining penalty functions in order to achieve the algorithm's main purpose: reduce the total number of required aircraft trips. Thus, these values are relatively smaller and equivalent; no type of relaxation in the constraints receives priority over the others. The aircraft usage fee for the first trip of any aircraft is $C_{j1} = 50,000$; subsequent trips impose a usage fee of $C_{jk} = 5,000 \ \forall \ k > 1$.

The improvement value is 5 % of the overall objective function value. The cessation value is 20 iterations, while the Max fix value is 5 iterations. The Max minor move value and Max non-improving move value are both 15. The planning ACL Max violation is 2.5 % greater than the planning ACL value, and the initial tabu tenure value is 20.

5.2 Scenario description

DALP-TS demonstrate its effectiveness and efficiency on 100 instances of six scenarios that vary two factors: Pallets and Aircraft. At the time of this research, no actual data sets were available at the DALP level. As a result, two types of pallet data sets were generated from line item information contained in an unclassified version of the January 2005 TPFDD from the Air Expeditionary Force cycle 9/10. Unfortunately, a TPFDD line item only provides the combined overall weight for all items as well as their associated temporal constraints; individual transportable good dimensions (i.e., weight, height, length, and depth) are not available.

To generate the pallet data from the information contained in a TPFDD line item, DALP-TS computes upper and lower bounds on the required number of pallets. The bounds indicate the smallest and largest number of pallets that could be created from the given cargo weight. In order to be a non-trivial load, pallets were assumed to have a lower weight bound of 2500 lb. The structural limitations of the pallet base, as well as most cargo aircraft, restrict the pallet upper weight limit to 10,000 lb. Thus, if a TPFDD line item specified 63,800 lb of cargo, the minimum number of possible pallets is: 63,800/10,000 = 7, and the

maximum number of pallets is: 63,800/2500 = 25. DALP-TS randomly selects an integer between these bounds (inclusive) as the total number of pallets and assigns a random weight between 2500 and 10,000 lb to each pallet such that the sum of the pallet weights equals the specified weight by the TPFDD line item. Each pallet associated with a given TPFDD line item has identical temporal constraints.

The first generated pallet data set utilized 20 distinct line items from the TPFDD with their associated weights and temporal constraints. The 20 line items totaled 2,160,950 lb of transportable goods with associated temporal constraints ranging between 1 and 29 days for available to load and required delivery dates, respectively. The second pallet data set extended the first to 40 line items (inclusive of the original 20) and contained 4,331,900 lb of transportable goods with associated temporal constraints ranging between 1 and 56 days for the available to load and required delivery dates, respectively. Although the second data set includes the line items in the first data set, the pallets associated with the line items were generated independently, i.e., the second data set does not contain the same pallet configurations generated for the first data set.

Although DALP-TS accommodates nine configurations of six different military airlift aircraft, the scenarios considered herein only incorporate the principle military strategic airlifters in the US Air Force inventory: the C-17 and C-5. Table 2 indicates that the Planning ACL for C-17 and C-5 aircraft are 90,000 and 150,000 lb, respectively. Additionally, Table 2 and Figs. 1, 2 denote that the C-17 in the logistics system configuration and C-5 aircraft can hold a maximum of 18 and 36 pallets, respectively.

5.3 Lower bound computation

By relaxing all constraints except aircraft cargo weight and volume capacities, a reasonable lower bound on the number of aircraft is determined. Consider a scenario instance using C-17 aircraft to transport the first 20 line items from the TPFDD which total 2,160,950 lb that have been divided into 500 pallets; the lower bound is computed as the maximum value of 2, 160, 950/90, 000 = 25 and 500/18 = 28. Likewise, consider a scenario instance using C-5 aircraft to transport the first 40 line items from the TPFDD which total 4,331,900 lb that have been divided 1000 pallets; the lower bound is computed as the maximum value of 4, 331, 900/ 150, 000 = 29 and 1000/36 = 28. Finally, consider the scenario using both aircraft types with alternating availability (starting with the C-17); if an odd number of aircraft is required in these scenarios, DALP-TS uses one additional C-17 aircraft. The lower bound for a scenario instance requiring transportation of 500 pallets is the maximum of 2 * (2, 160, 950/(90, 000 + 150, 000)) = 19 and 2 * (500/(18 + 36)) = 19; while the lower bound for an

Table 4 Example instance used for DALP-TS algorithmic testing

Days in timeline	Type of aircraft	Number of pallets	Lower bound on required number of aircraft trips
29	C-17	500	28
56	C-17	1000	56
29	C-5	500	15
56	C-5	1000	29
29	C-17 and C-5	500	19
56	C-17 and C-5	1000	38

instance requiring transportation of 1000 pallets is the maximum of 2 * (4, 331, 900/(90, 000 + 150, 000)) = 37 and 2 * (1000/(18 + 36)) = 38. In these two scenarios, the value inside the ceiling function is doubled to account for multiple aircraft types. Table 4 presents the six problem scenario and their associated lower bounds on the number of aircraft trips for these two example instances.

To properly demonstrate the efficacy of DALP-TS, two sets of 100 instances of pallets with temporal restrictions (i.e., a short and long timeline) were randomly generated using the previously described methodology. Table 5 presents a description of the instances and their associated lower bounds.

5.4 Results summary

Running on an HP Pavilion desktop computer with an AMD Phenom II X4 960T processor (CPU speed of 3.00 GHz) and 8GM of DDR SDRAM, DALP-TS generated solutions for each of the 100 instances of the six problem scenarios. To determine the quality of the solutions, the resulting required number of aircraft trips was compared to the lower bounds for each of the 100 instances. The DALP-TS solution percentage from the lower bound is computed as:
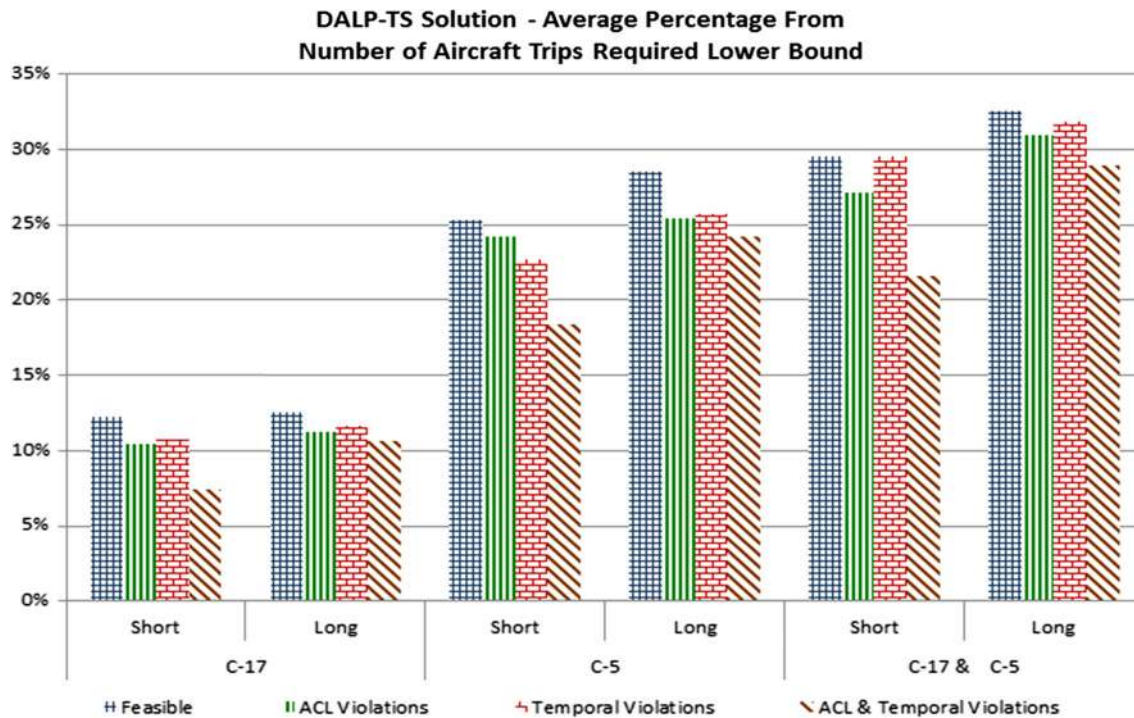
Percentage from lower bound

$$= \frac{(\text{Solution value} - \text{Lower bound value})}{\text{Lower bound value}}$$

The resulting values for the 100 instances were averaged for each of the six problem scenarios. Figure 3 graphically depicts the results for the comparison of the required number of aircraft trips with the lower bound.

Figure 3 demonstrates that for the C-17 aircraft, DALP-TS produced feasible solutions within an average of 12.5 % of the lower bound value on the required number of aircraft trips for both the short and long timelines. Additionally, DALP-TS produced average solutions with either (but not both) ACL Violations and Temporal Violations that demonstrated minor (<2 %) average improvements over the feasible solutions. Finally, DALP-TS produced

Table 5 Instance values for DALP-TS algorithmic testing

Days in timeline	Type of aircraft	Number of pallets			Lower bound on required number of aircraft trips		
		MIN	AVG	MAX	MIN	AVG	MAX
29	C-17	426	539.36	653	25	30.41	37
56	C-17	853	1076.99	1271	49	60.27	71
29	C-5	426	539.36	653	15	15.83	19
56	C-5	853	1076.99	1271	29	30.72	36
29	C-17 and C-5	426	539.36	653	19	20.68	25
56	C-17 and C-5	853	1076.99	1271	37	40.64	48

Fig. 3 DALP-TS required aircraft trips percentage from lower bound value

infeasible solutions with both ACL and Temporal Violations that averaged within 7.4 and 10.7 % of the lower bound for the short and long timelines, respectively.

Figure 3 also conveys that, for C-5 aircraft, DALP-TS produced feasible solutions within 25.33 and 28.6 % of the lower bound for the short and long timelines, respectively. This jump is approximately double that of the C-17 aircraft, which naturally poses the question: Why does a large difference exist between the two aircraft types? The answer is based on the ratio of the planning ACL to the number of pallet positions for an aircraft type. The C-17 can hold an average of 5000 lb per pallet position (i.e., 90,000 lb/18 pallet positions), while the C-5 can only hold an average of 4166.67 lb per pallet position (i.e., 150,000 lb/36 pallet positions). The smaller average weight per pallet position

implies that a C-5 aircraft is more likely to reach its ACL (weight) constraint before its pallet load (volume) constraint than a C-17 aircraft. This also translates to the scenarios with a combination of C-5 and C-17 aircraft; in these cases, the average weight per pallet position is 4444.44 (i.e., [150,000 + 90,000]/[36 + 18]).

The rationale for the difference in solution quality for differing aircraft types is further demonstrated after analyzing which characteristic of the pallet data sets drives the lower bound calculation for the 100 instances each problem scenario. Since the total weight of the pallet sets are fixed at 2,160,950 and 4,331,900 lb for the short and long scenarios, the lower bound on the required number of aircraft based on total pallet weight is fixed for a given aircraft type. For the C-17 aircraft, this lower bound is 2,160,950/

Table 6 Count of binding characteristic for lower bound computation from the 100 problem instances

Aircraft type	Timeline length	Pallet set characteristic	
		Binding by number	Binding by weight
C-17	Short	99	4
	Long	99	2
C-5	Short	70	52
	Long	82	35
C-17 and C-5	Short	82	37
	Long	89	16

Table 7 DALP-TS average time to achieve a solution

DALP-TS solution—average time (seconds)			
Solution type	C-5	C-17	Combo
Short problem			
Feasible	166.23	46.86	78.52
ACL violations	144.81	41.30	70.82
Temporal violations	137.17	38.82	73.72
ACL and temporal violations	358.54	120.78	238.08
Long problem			
Feasible	773.59	243.28	419.34
ACL violations	682.08	211.92	377.29
Temporal violations	644.04	200.54	392.75
ACL and temporal violations	1663.41	624.20	938.30

$90,000 = 26$ for the short timeline pallet sets and $4,331,900/90,000 = 49$ for the long timeline pallet sets. Thus, the lower bound on the number of aircraft required will never fall below these values; the lower bound will be greater if the total number of pallets results in a value greater than 26 or 49 for the short and long timelines, respectively. Table 6 presents a count of the total number of instances in which the lower bound is binding from either the number of pallets or the total weight of all pallets. The combined value of both of these cases is greater than 100 because some problem instances were generated such that both characteristics are simultaneously binding against the lower bound.

The average time (in seconds) to produce a solution for each of the 100 instances of the six problem scenarios are presented in Table 7. Note that the longest average time to locate a solution is less than 30 min (1663.41 s–27 min, 45 s). This is for a scenario requiring 56 days for cargo delivery; a 30-min time to produce a solution which is, on average, ~ 24 % from the theoretical lower bound should be acceptable to a decision maker.

6 Summary

Prior to this research, the DALP had not been formulated or modeled. More general models exist, but these models do not assign items to specific positions in an aircraft and/or consider the temporal restrictions. Instead, these models assign items based upon weight to aircraft and assume that the aircraft has the spatial capacity to accommodate the items.

The DALP-TS algorithm expands upon the solution representation used in the static airlift loading problem-tabu search by including departure and arrival dates for aircraft as well as multiple trips for aircraft [34]. As with the previous research, the representation lends itself to the TS methodology, specifically inserting and swapping of pallets and adding and removing aircraft. The neighborhood definition in DALP-TS exploits the structure of the solution representation.

DALP-TS developed a unique dynamic neighborhood selection process. The dynamic neighborhood selection process incorporates problem specific knowledge as well as understanding of the TS search process. This process enables DALP-TS to effectively and efficiently search areas of quality solutions by ensuring proper neighborhood selection.

DALP-TS produces an array of solutions from which the decision maker can select the preferred solution. If encountered during the search process, DALP-TS reports the best solution from four different types of solutions. The four different types of solutions are feasible, those with only ACL violations, those with temporal only violations, and those with both ACL and temporal violations.

DALP-TS demonstrated its efficacy and efficiency on a series of 100 problem instances generated from information contained in a TPFDD. No previous baseline or comparison results were available for the DALP prior to this research. As a result, the DALP-TS results were compared against lower bounds on the number of aircraft trips required. These lower bound values were computed by relaxing the aircraft loading and cargo temporal constraints.

References

1. Air Force Doctrine Document 1 (2003) Air Force Basic Doctrine, Complement of Joint Publication 1: Joint Warfare of the Armed Forces of the United States
2. Baker SF, Morton DP, Rosenthal RE, Williams LM (2002) Optimizing military airlift. Oper Res 50(4):582–602
3. Barnes JW, Wiley V, Moore JT, Ryer D (2004) Solving the aerial fleet refueling problem using group theoretic tabu search. Math Comput Model 39(6–8):617–640
4. Bhatia AK, Basu SK (2004) Packing bins using multi-chromosomal genetic representation and better-fit heuristic. Lect Notes Comput Sci 3316:181–186
5. Bookbinder JH, Elhedhli S, Li Z (2015) The air-cargo consolidation problem with pivot weight: models and solution methods. Comput Oper Res 59:22–32
6. Callander BD (1998) The evolution of air mobility. Air Force Mag Online J Air Force Assoc 81(2):68–73
7. Cochard DD, Yost KA (1985) Improving utilization of air force cargo aircraft. Interfaces 15(1):53–68
8. Computer Sciences Corporation (1997) Unit type code development, tailoring, and optimization (UTC-DTO) phase 2 final report, contract DCA100-94-D-00144. Defense Enterprises Integration Services, Falls Church, VA
9. Costa MDG, Captivo ME (2016) Weight distribution in container loading: a case study. Int Trans Oper Res 23(1–2):239–263
10. Crino J, Moore JT, Barnes JW, Nanry WP (2004) Solving the military theater distribution vehicle routing and scheduling problem using group theoretic tabu search, invited paper for a special issue. Math Comput Model 39(68):599–616
11. Elhedhli A (2003) Ranking lower bounds for the bin-packing problem. Eur J Oper Res 160:24–46
12. Feng B, Li Y, Shen ZJM (2015) Air cargo operations: literature review and comparison with practices. Transp Res Part C Emerg Technol 56:263–280
13. Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-completeness. W.H. Freeman, San Francisco, CA
14. Giangreco DM, Griffin RE (1988) Airbridge to Berlin—The Berlin Crisis of 1948, its Origins and Aftermath. Presidio Press, Novato, CA
15. Glover F (1989) Tabu search—part I. ORSA J Comput 1(3):190–206
16. Glover F (1990) Tabu search—part II. ORSA J Comput 2(1):4–32
17. Guéret C, Jussien N, Lhomme O, Pavageau C, Prins C (2003) Loading aircraft for military operations. J Oper Res Soc 54:458–465
18. Hilliard MR, Solanki RS, Liu C, Busch IK, Harrison G, Kraemer RD (1992) Scheduling the operation desert storm airlift: an advanced automated scheduling support system. Interfaces 22(1):131–146
19. Harwig J, Barnes JW, Moore J (2006) An adaptive tabu search approach to cutting and packing problems. Mil Oper Res 11(2):5–26
20. Heidelberg KR, Parnell GS, Ames JE IV (1998) Automated air load planning. Nav Res Logist 45:751–768
21. Lambert GR, Barnes JW, Van Velhuizen D (2007) A tabu search approach to the strategic airlift problem. Mil Oper Res J 12(3):59–79
22. Li Y, Tao Y, Wang F (2009) A compromised large-scale neighborhood search heuristic for capacitated air cargo loading planning. Eur J Oper Res 199(2):553–560
23. Limbourg S, Schyns M, Laporte G (2012) Automatic aircraft cargo load planning. J Oper Res Soc 63(9):1271–1283
24. Lodi A, Martello S, Vigo D (2002) Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. INFORMS J Comput 11(4):345–357
25. Lurkin V, Schyns M (2015) The airline container loading problem with pickup and delivery. Eur J Oper Res 244(3):955–965
26. McKinzie K, Barnes JW (2006) A tabu search approach to strategic mobility mode selection problem. Air Force J Logist XXX(3):51–61
27. Mongeau M, Bes C (2003) Optimization of aircraft container loading. IEEE Trans Aerosp Electron Syst 39(1):140–150
28. Nance RL, Roesener AG, Moore JT (2010) An advanced tabu search approach for solving the mixed payload airlift loading problem. J Oper Res Soc 62:337–347
29. Ng KYKA (1992) Multicriteria optimization approach to aircraft loading. Oper Res 40(6):1200–1205
30. Paquay C, Schyns M, Limbourg S (2016) A mixed integer programming formulation for the three-dimensional bin packing problem deriving from an air cargo application. Int Trans Oper Res 23(1–2):187–213
31. Pollaris H, Braekers K, Caris A, Janssens GK, Limbourg S (2015) Vehicle routing problems with loading constraints: state-of-the-art and future directions. OR Spectr 37(2):297–330
32. Raidl GR, Kodydek G (1998) Genetic algorithms for the multiple container packing problem. Lect Notes Comput Sci 1498:875–884
33. Reiman AD, Main BD, Anderson BE (2015) Enhancing airlift fuel efficiency through increased utilization of cargo capacity. J Def Model Simul Appl Method Technol 12:19–29
34. Roesener AG, Barnes JW, Moore JT, Van Veldhuizen D (2010) An advanced tabu search approach to the static airlift loading problem. Mil Oper Res J 15(1):5–29
35. Roesener AG, Hall SN (2014) A nonlinear integer programming formulation for the airlift loading problem with insufficient aircraft. J Nonlinear Anal Optim 5(1):125–141
36. Solanki RS, Southworth F (1991) An execution planning algorithm for military airlift. Interfaces 21(4):121–131
37. Thomas C, Campbell K, Hines G, Racer M (1998) Airbus packing at federal express. Interfaces 28(4):21–30
38. United States Department of Defense Form 2130-1 (DD Form 2130-1): "C-5 Load Plan" (SEP 1998). Available on Defense Technical Information Center (DTIC). http://www.dtic.mil/whs/directives/forms/eforms/dd2130-1.pdf. Accessed 7 Mar 2016
39. United States Department of Defense Form 2130-13 (DD Form 2130-13): "C-17 Load Plan" (SEP 1998). Available on Defense Technical Information Center (DTIC). http://www.dtic.mil/whs/directives/forms/eforms/dd213013.pdf. Accessed 7 Mar 2016
40. United States Transportation Command Website. "About USTRANSCOM." http://www.ustranscom.mil/cmd/aboutustc.cfm. Accessed 12 Mar 2016
41. Vancroonenburg W, Verstichel J, Tavernier K, Vanden Berghe G (2014) Automatic air cargo selection and weight balancing: a mixed integer programming approach. Transp Res Part E Logist Transp Rev 65:70–83