INFOR, Vol. 50, No. 2, May 2012, pp. 88–94
ISSN 0315-5986 | EISSN 1916-0615

On the Computational Complexity of Peer-to-Peer Satellite Refueling Strategies

Sofie Coene and Frits C. R. Spieksma
Research Group Operations Research and Business Statistics (ORSTAT), Katholieke Universiteit Leuven, Naamsestraat 69, 3000 Leuven, Belgium, e-mail: sofie.coene@ecom.kuleuven.be; frits.spieksma@econ.kuleuven.be

Atri Dutta
Department of Mechanical and Aerospace Engineering, Princeton University, Olden Street, Princeton, NJ 08544-5263, USA, e-mail: atrid@princeton.edu

Panagiotis Tsiotras
Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, 270 Ferst Drive, NW Atlanta, GA 30332-0150, USA, e-mail: tsiotras@gatech.edu

Abstract—We revisit the peer-to-peer on-orbit satellite refueling problem, in which the maneuvering satellites are allowed to interchange their orbital positions. We show that the problem is computationally hard by reducing it to a special case of the three-index assignment problem. On the positive side, we show that the size of instances in practice is such that a state-of-the-art integer programming solver is able to find globally optimal solutions in little computing time.

Keywords On-orbit satellite refueling, three-index assignment problem, computational complexity, integer programming.

1. INTRODUCTION

Refueling is an important aspect of on-orbit servicing operations, because of the immense benefits that can be obtained by extending the useful lifetime of the spacecraft, by increasing the flexibility of space missions, and by reducing the overall cost of space operations. The conventional notion about on-orbit refueling of a system of multiple satellites is to have a dedicated refueling spacecraft visit the satellites, and impart fuel to them. Alternatively, in the absence of an external refueling spacecraft, satellites with a large amount of fuel may exchange fuel with fuel-deficient satellites, so that, at the end of the process, all satellites have at least the required amount of fuel. This is known as the Peer-to-Peer (P2P) refueling strategy, (Shen and Tsiotras, 2005). The P2P strategy is an integral part of the mixed refueling strategy, during which the dedicated refueling spacecraft only refuels part (at least half) of the satellites, which in turn distribute the fuel to the remaining satellites via P2P refueling. In terms of the fuel expended during all orbital transfers taking place during a refueling process, the mixed refueling strategy is better than the conventional strategy, particularly as the number of satellites in the constellation increases and the time to complete the refueling decreases (Dutta and Tsiotras, 2006).

In this paper, we focus on the P2P strategy, as described in Dutta and Tsiotras (2008). Given are n satellites distributed over n slots in a circular orbit. Each satellite is denoted as either *fuel-deficient*, or *fuel-sufficient*. This depends on whether the satellite has at least the minimum required amount of fuel. The idea put forward in Dutta and Tsiotras (2008) is that the satellites can redistribute the fuel among themselves by engaging in fuel exchange in pairs. For each pair, a satellite moves (performs an orbital transfer) to rendezvous with another satellite, exchanges fuel, and returns back to any available slot. We say that a pair of satellites is *feasible* when (i) the pair consists of a fuel-sufficient satellite and a fuel-deficient satellite, and (ii) the amount of fuel carried by the two satellites of the pair can be redistributed among them in a way

Received May 2011; Revision December 2011, Accepted March 2012

such that each satellite of the pair has at least the required amount of fuel at the end of the refueling process. We assume that the satellites can only engage in a *non-cooperative* rendezvous, that is, only one of the satellites in a feasible pair is active and initiates the orbital transfer necessary to rendezvous. Associated with a feasible pair of satellites is also a set of orbital slots where the active satellite can return to. The goal of the P2P refueling problem is to decide which satellites pair up for feasible fuel exchanges, and which positions the active satellites return to, in order to minimize fuel costs. Note that an active satellite can return to any vacant orbital slot, and not necessarily to its original slot. We call this problem the Egalitarian-P2P (E-P2P) refueling strategy, see Dutta and Tsiotras (2008).

The purpose of this note is twofold: First, to establish the computational complexity of the E-P2P refueling problem, and second, to show that the size of the instances that occur in practice allow using a state-of-the-art integer programming solver to solve these instances to optimality.

2. PROBLEM STATEMENT

Let us consider a system of n satellites, distributed over n slots in a circular orbit. For the sake of simplicity, we consider that satellite s_i occupies the slot ϕ_i, for all $i \in \{1, 2, \dots, n\}$. The task of redistributing fuel is now accomplished by considering maneuvers of the satellites that we represent by an ordered triple (i, j, k): this means that satellite s_i at position ϕ_i moves to rendezvous with satellite s_j at position ϕ_j, and after undergoing a fuel exchange, satellite s_i moves to position ϕ_k (while satellite s_j stays at position ϕ_j). Clearly, we only take into account those triples that contain a feasible pair of satellites. Observe that the maneuvering satellite can be fuel-sufficient (in which case this satellite will donate fuel to its companion) or fuel-deficient (in which case the satellite will receive fuel from its companion). There is a given cost-coefficient $c(i, j, k)$ associated with each triple. These cost-coefficients correspond to the amount of fuel expended during the orbital transfers. Notice that $c(i, j, k)$ depends on a number of aspects, such as the angle of separation of the satellites, the mass and engine of the maneuvering satellite, etc. Notice also that it might happen that the amount of fuel of the two satellites making up the feasible pair is not sufficient for the moving satellite to reach some position ϕ_k with enough fuel to be sufficient. In that case, the coefficient $c(i, j, k)$ simply takes on a large value. We come back to this issue of fuel cost extensively further in this section. The goal is to find triples (i, j, k) such that:

(i) each satellite is in at most one triple,
(ii) each satellite that is fuel-deficient is in at least one triple with a fuel-sufficient satellite,
(iii) at the end of the refueling process, each position is occupied by exactly one satellite, and
(iv) the total fuel costs are minimal.

The above is a mathematical statement of the E-P2P refueling problem, and a set of triples that satisfies (i), (ii), and (iii) is said to be a feasible solution to the problem. Clearly, for a solution to exist, the number of fuel-deficient satellites should not exceed the number of fuel-sufficient satellites.

Recall that we allow explicitly for a satellite to end its movement in a position that is different from its starting position. Clearly, one could restrict the solutions to be such that the starting and ending slots are identical, that is, $\phi_i = \phi_k$. Although this would make the problem easy to solve (see Section 3), it has been illustrated that this might lead to suboptimal solutions, i.e., solutions with a larger than necessary fuel cost (Dutta and Tsiotras, 2008).

Let us consider a triple (i, j, k) and the associated cost $c(i, j, k)$. The movement of the active satellite s_i consists of two trips: the forward trip, when it moves from its position ϕ_i to another position ϕ_j harboring satellite s_j, and the return trip, where satellite s_i travels from position ϕ_j to position ϕ_k. Let p^i_{ij} denote the fuel expended by the satellite s_i when it transfers from slot ϕ_i to slot ϕ_j. The fuel consumed by satellite i in the first part can then be expressed as (see Dutta and Tsiotras (2008)):

$$p^i_{ij} = (m_{\mathrm{si}} + f_i^-)(1 - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}), \tag{1}$$

where m_{si} refers to the mass of the permanent structure of satellite s_i $(1 \le i \le n)$ (fuel not included), f_i^- refers to the initial amount of fuel present in satellite s_i, I_{spi} refers to the specific impulse of the engine of satellite s_i, ΔV_{ij} is the velocity change required to transfer from orbital slot ϕ_i to slot ϕ_j, for each ordered pair of positions ϕ_i and ϕ_j, $(1 \le i, j \le n)$, and g_0 is a constant denoting the acceleration due to gravity at the Earth's surface.

In order to express the fuel consumption of an active satellite s_i in the second part of its maneuver, we need to know how much fuel is present in satellite s_i after the exchange. This quantity is chosen such that the amount of fuel present in the active satellite at the completion of its return to position ϕ_k is the minimum amount needed in order to keep itself operational for the remaining lifetime of the satellite. (This follows from the fact that there is no point in moving fuel (mass) which is not needed.) Now, by letting p^i_{jk} stand for the fuel consumption of satellite s_i when traveling from position ϕ_j to position ϕ_k, we can write down the following equation:

$$p^i_{jk} = (m_{\mathrm{si}} + \underline{f}_i + p^i_{jk})(1 - e^{-\Delta V_{jk}/g_0 I_{\mathrm{spi}}}), \tag{2}$$

where $\underline{f}_i$ refers to the amount of fuel required for satellite s_i after it has reached its position following the exchange.

Solving this equation for p^i_{jk} gives:

$$p^i_{jk} = (m_{\mathrm{si}} + \underline{f}_i)(e^{\Delta V_{jk}/g_0 I_{\mathrm{spi}}} - 1). \tag{3}$$

INFOR, Vol. 50, No. 2, May 2012, pp. 88–94
ISSN 0315-5986 | EISSN 1916-0615
DOI 10.3138/infor.50.2.088

Summing equations (1) and (3), we derive an explicit expression for the cost-coefficients $c(i, j, k)$:

$$c(i,j,k) = p^i_{ij} + p^i_{jk} = m_{\mathrm{si}}(e^{\Delta V_{jk}/g_0 I_{\mathrm{spi}}} - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}) + f^-_i(1 - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}) + \underline{f}_i(e^{\Delta V_{jk}/g_0 I_{\mathrm{spi}}} - 1). \quad (4)$$

The problem of E-P2P refueling seeks to find the set of feasible triples which minimize the total fuel cost. In order to analyze the computational complexity of the problem, we now describe precisely what constitutes the input to the E-P2P refueling decision problem. For each satellite s_i $(i \in \{1, 2, \ldots, n\})$ the associated data of the problem are $m_{\mathrm{si}}, f^-_i, I_{\mathrm{spi}}$, and $\underline{f}_i$. For each pair of positions ϕ_i, ϕ_j where $i, j \in \{1, 2, \ldots, n\}$, the associated input data is ΔV_{ij}, the total velocity change required to go from position ϕ_i to position ϕ_j. Finally, L is an integer, denoting an upper bound on the total fuel cost of a solution.

Given this input, the decision version of the E-P2P refueling problem seeks to find the answer to the following question:

Does there exist a feasible set of triples such that the summation of the total fuel costs (as expressed by (4)) does not exceed L?

3. THE COMPLEXITY OF SATELLITE PEER-TO-PEER REFUELING PROBLEM

Computational complexity is a field that attempts to establish the hardness of optimization problems. We refer to Garey and Johnson (1979) for a classical introduction.

In this section we prove that the E-P2P refueling problem is NP-hard, by exhibiting a reduction from a special 3-dimensional assignment problem, i.e. the 3-dimensional axial assignment problem with decomposable costs (3AP-DC), which was introduced in, and proven to be NP-hard by Burkard et al. (1996). This implies that it is unlikely (unless P = NP) that a polynomial time algorithm for this problem exists.

Problem: [3-dimensional assignment with decomposable coefficients (3AP-DC)]

Input: $3q$ nonnegative numbers a_i, b_i and c_i, for $i = 1, 2, \ldots, q$ and an integer K.

Question: do there exist two permutations $\pi : \{1, 2, \ldots, q\} \mapsto \{1, 2, \ldots, q\}$ and $\sigma : \{1, 2, \ldots, q\} \mapsto \{1, 2, \ldots, q\}$ such that $\sum_{i=1}^{q} a_i b_{\pi(i)} c_{\sigma(i)} \leq K$?

We now show that the E-P2P is at least as hard as the 3AP-DC. To this end, we build an instance of the E-P2P problem that consists of q fuel-sufficient satellites (indexed by $i = 1, 2, \ldots, q$), as well as q fuel-deficient satellites (indexed by $i = q + 1, q + 2, \ldots, 2q$). Hence $n := 2q$. We now specify all input parameters (see Section 2), using the symbol M for some large value. For each satellite s_i $(i = 1, \ldots, n)$, we set:

$$f^-_i := M, \qquad I_{\mathrm{spi}} := 1.$$

For each fuel-sufficient satellite s_i $(i = 1, 2, \ldots, q)$, we set:

$$\underline{f}_i := 0, \qquad m_{\mathrm{si}} := a_i.$$

For each fuel-deficient satellite s_i $(i = q + 1, q + 2, \ldots, 2q)$, we set:

$$\underline{f}_i := M + 1, \qquad m_{\mathrm{si}} := M.$$

For each pair of positions (ϕ_i, ϕ_j) where position ϕ_i harbors a fuel-sufficient satellite $(1 \leq i \leq q)$, and position ϕ_j harbors a fuel-deficient satellite $(q + 1 \leq j \leq 2q)$, we set

$$\Delta V_{ij} := 0, \text{ and } \Delta V_{ji} := g_0 \ln(b_j c_i + 1). \quad (5)$$

Finally, we set $L := K$.

This completes the description of the instance of the E-P2P refueling problem. Notice that we have allowed for non-symmetric ΔV_{ij} values.

We now argue that there exists a 1-to-1 correspondence between yes-instances of 3AP-DC and the E-P2P refueling problem. Suppose that the instance of 3AP-DC admits a yes-answer, i.e., there exist permutations π and σ such that $\sum_{i=1}^{q} a_i b_{\pi(i)} c_{\sigma(i)} \leq K$. Then we simply copy that solution to the E-P2P refueling problem: fuel-sufficient satellite s_i $(1 \leq i \leq q)$ moves to fuel-deficient satellite $s_{q+\pi(i)}$, and afterwards returns to position $\phi_{\sigma(i)}$. The costs of these movements follow from substituting the values given above in (4), leading to

$$\begin{aligned} c(i,j,k) &= m_{\mathrm{si}}(e^{\Delta V_{jk}/g_0 I_{\mathrm{spi}}} - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}) \\ &\quad + f^-_i(1 - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}) + \underline{f}_i(e^{\Delta V_{jk}/g_0 I_{\mathrm{spi}}} - 1) \\ &= a_i(e^{\ln(b_{\pi(i)+q} c_{\sigma(i)} + 1)} - 1) = a_i b_{\pi(i)} c_{\sigma(i)}. \end{aligned}$$

Hence, if there exist permutations π and σ such that $\sum_{i=1}^{q} a_i b_{\pi(i)} c_{\sigma(i)} \leq K$, then there exists a solution with cost bounded by L for the E-P2P refueling problem.

Suppose now that the instance of the E-P2P problem has a solution with a cost no more than L. Notice that the fuel-deficient satellites cannot move due to their large mass $m_{\mathrm{si}} = M$; moving any fuel deficient satellite would lead to costs exceeding L. Thus, we can define the permutation π by inspecting which fuel sufficient satellite s_i visits which fuel deficient satellite $s_{\pi(i)}$ where $\pi(i) := j - q$, and we can find the permutation σ by verifying to which position $\phi_{\sigma(i)} := \phi_k$ each satellite s_i returns. The resulting solution to 3AP-DC will have cost no more than K. All this leads to:

Theorem 1 *The Egalitarian Peer-to-Peer refueling problem is NP-hard.*

Clearly, the difficulty of the E-P2P problem changes when we restrict the solutions such that each satellite s_i must return to its

DOI 10.3138/infor.50.2.088

original position ϕ_i, $(1 \leq i \leq n)$. The costs then simplify to:

$$c(i,j,i) = p^i_{ij} + p^i_{ji} = m_{\mathrm{si}}(e^{\Delta V_{ji}/g_0 I_{\mathrm{spi}}} - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}) + f^-_i(1 - e^{-\Delta V_{ij}/g_0 I_{\mathrm{spi}}}) + \underline{f}_i(e^{\Delta V_{ji}/g_0 I_{\mathrm{spi}}} - 1),$$

and no longer depend on k. Hence, by modeling the resulting problem as an assignment problem (see, for example, Shen and Tsiotras (2005); Dutta and Tsiotras (2006)) with costs $d(i,j) = \min(c(i,j,i), c(j,i,j))$ between the fuel sufficient satellites s_i and the fuel deficient satellites s_j, it is clear that efficient algorithms exist (see Burkard et al. (2009)) for this special case of the problem.

4. COMPUTATIONAL RESULTS USING AN INTEGER PROGRAMMING FORMULATION

Let us now write down an integer programming model for the E-P2P refueling problem. To simplify notation, we use FD (FS) to denote the set of indices of fuel-deficient (fuel-sufficient) satellites. We let $S \equiv FD \cup FS$ stand for the set of all satellites. We use a decision variable

$$x_{ijk} := \begin{cases} 1 & \text{if satellite } s_i \text{ moves to satellite } s_j \\ & \text{and ends up in position } \phi_k; \\ 0 & \text{otherwise.} \end{cases}$$

Recall that we only define variables for those triples that contain a feasible pair, and whose corresponding maneuver is called *feasible* in Dutta and Tsiotras (2008); for instance, variables of the form x_{ijj} do not exist.

We therefore have the following optimization problem for solving the E-P2P problem:

$$\min \sum_{i,j \in S} \sum_{k=1}^{n} c(i,j,k) x_{ijk} \tag{6}$$

$$\text{s.t.} \sum_{j \in FS} \sum_k x_{jik} + \sum_{j \in FS} \sum_k x_{ijk} = 1 \quad \text{for all } i \in FD \tag{7}$$

$$\sum_{j \in FD} \sum_k x_{ijk} + \sum_{j \in FD} \sum_k x_{jik} \leq 1 \quad \text{for all } i \in FS \tag{8}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} - \sum_{j \in S} \sum_{\ell \in S} x_{kj\ell} \leq 0 \quad \text{for all } k \tag{9}$$

$$x_{ijk} \in \{0,1\} \quad \text{for all } i,j,k. \tag{10}$$

Notice that:

- The cost-coefficients in the objective function (6) are defined by (4),
- Constraints (7) express that each fuel-deficient satellite is in one triple with a fuel-sufficient satellite,
- Constraints (8) express that each fuel-sufficient satellite is in at most one triple,
- Constraints (9) express that each position can harbor at most one satellite; indeed, if some satellite ends up in position ϕ_k, then the satellite starting at that position must have moved,
- In case the number of fuel-deficient satellites equals the number of fuel-sufficient satellites, the resulting constraints become exactly those from an axial three-index assignment problem (with the cost-coefficients displaying a specific structure; see Spieksma (2000) for an overview of other possible structures).

The model (6)–(10) was implemented in ILOG OPL Development Studio 5.2, and run on a personal computer with a 2.20 GHz Intel Core 2 Duo processor and 2.00 GB of RAM. We experimented with two sets of instances. The first set contains 10 realistic instances from Dutta and Tsiotras (2008) whose characteristics can be found in Table 1. The second set contains 120 instances that we randomly generated (see Table 4).

We first discuss the outcomes of the first set of instances as described in Tables 2 and 3. Table 2 shows the results of this numerical investigation. Table 3 contains the corresponding

TABLE 1.
Sample constellations

Label	Description
C_1	Altitude $= 35,786$ Km, $n = 10$, $T = 12$, $\bar{f}_i = 30$, $\underline{f}_i = 12$, $m_{si} = 70$ f^-_i: 30, 30, 6, 6, 6, 6, 6, 30, 30, 30
C_2	Altitude $= 1,200$ Km, $n = 16$, $T = 30$, $\bar{f}_i = 30$, $\underline{f}_i = 15$, $m_{si} = 70$ f^-_i: 30, 30, 30, 30, 30, 30, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30
C_3	Altitude $= 2,000$ Km, $n = 12$, $T = 30$, $\bar{f}_i = 30$, $\underline{f}_i = 15$, $m_{si} = 70$ f^-_i: 30, 30, 30, 10, 10, 10, 10, 10, 10, 30, 30, 30
C_4	Altitude $= 6,000$ Km, $n = 18$, $T = 25$, $\bar{f}_i = 25$, $\underline{f}_i = 12$, $m_{si} = 75$ f^-_i: 25, 25, 25, 25, 25, 25, 25, 25, 25, 6, 6, 6, 6, 6, 6, 6, 6, 6
C_5	Altitude $= 12,000$ Km, $n = 12$, $T = 20$, $\bar{f}_i = 25$, $\underline{f}_i = 12$, $m_{si} = 75$ f^-_i: 25, 25, 25, 25, 25, 25, 8, 8, 8, 8, 8, 8
C_6	Altitude $= 1,400$ Km, $n = 14$, $T = 35$, $\bar{f}_i = 25$, $\underline{f}_i = 12$, $m_{si} = 75$ f^-_i: 25, 25, 25, 25, 25, 25, 25, 8, 8, 8, 8, 8, 8, 8
C_7	Altitude $= 30,000$ Km, $n = 16$, $T = 15$, $\bar{f}_i = 30$, $\underline{f}_i = 15$, $m_{si} = 70$ f^-_i: 10, 10, 10, 10, 10, 10, 10, 10, 28, 28, 28, 28, 28, 28, 28, 28
C_8	Altitude $= 1,200$ Km, $n = 16$, $T = 30$, $\bar{f}_i = 30$, $\underline{f}_i = 15$, $m_{si} = 70$ f^-_i: 30, 10, 30, 10, 30, 10, 30, 10, 30, 10, 30, 10, 30, 10, 30, 10
C_9	Altitude $= 2,000$ Km, $n = 36$, $T = 30$, $\bar{f}_i = 30$, $\underline{f}_i = 15$, $m_{si} = 70$ f^-_i: 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30, 8, 30
C_{10}	Altitude $= 2,000$ Km, $n = 40$, $T = 30$, $\bar{f}_i = 25$, $\underline{f}_i = 12$, $m_{si} = 75$ f^-_i: 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6

ISSN 0315-5986 | EISSN 1916-0615
DOI 10.3138/infor.50.2.088

TABLE 2.
Costs and running times

Constellation	n	Opt	Running time (s)	Local Search	Restricted
C1	10	18.515 (i)	0.078	19.11 (3.2%)	26.072 (41%)
C2	16	24.356 (i)	0.172	24.82 (1.9%)	37.478 (54%)
C3	12	18.452 (i)	0.156	18.87 (2.3%)	26.428 (43%)
C4	18	25.787 (i)	0.266	26.26 (1.8%)	40.728 (58%)
C5	12	18.792 (i)	0.125	18.86 (0.36%)	28.385 (51%)
C6	14	19.025 (i)	0.156	19.26 (1.2%)	28.774 (51%)
C7	16	22.539 (i)	0.234	22.75 (0.94%)	34.976 (55%)
C8	16	9.0826 (i)	0.203	10.18 (12%)	9.3751 (3,2%)
C9	36	8.3445 (i)	1.344	na	8.3837 (0,47%)
C10	40	52.0535	2.891	na	90.696 (74%)

TABLE 3.
Satellite assignments

Constellation	Assignment
C1	$s_1 \to s_4 \to s_5, s_3 \to s_2 \to s_3, s_5 \to s_9 \to s_8, s_6 \to s_{10} \to s_1, s_8 \to s_7 \to s_6$
C2	$s_1 \to s_{12} \to s_{11}, s_4 \to s_9 \to s_{10}, s_6 \to s_7 \to s_8, s_8 \to s_5 \to s_6, s_{10} \to s_3 \to s_4,$ $s_{11} \to s_{16} \to s_{15}, s_{13} \to s_2 \to s_1, s_{15} \to s_{14} \to s_{13}$
C3	$s_2 \to s_5 \to s_6, s_4 \to s_3 \to s_4, s_6 \to s_1 \to s_2, s_7 \to s_{10} \to s_9, s_9 \to s_{12} \to s_{11},\ s_{11} \to s_8 \to s_7$
C4	$s_2 \to s_{17} \to s_{16}, s_4 \to s_{15} \to s_{14}, s_7 \to s_{10} \to s_{11}, s_9 \to s_{12} \to s_{13}, s_{11} \to s_8 \to s_9,$ $s_{13} \to s_6 \to s_7, s_{14} \to s_5 \to s_4, s_{16} \to s_1 \to s_{18}, s_{18} \to s_3 \to s_2$
C5	$s_1 \to s_{12} \to s_{11}, s_4 \to s_7 \to s_8, s_6 \to s_9 \to s_{10}, s_8 \to s_5 \to s_6, s_{10} \to s_2 \to s_1, s_{11} \to s_3 \to s_4$
C6	$s_2 \to s_{13} \to s_{12}, s_4 \to s_{11} \to s_{10}, s_7 \to s_8 \to s_9, s_9 \to s_6 \to s_7, s_{10} \to s_5 \to s_4,\ s_{12} \to s_1 \to s_{14}, s_{14} \to s_3 \to s_2$
C7	$s_1 \to s_{16} \to s_1, s_3 \to s_{14} \to s_{15}, s_4 \to s_{13} \to s_{12}, s_6 \to s_9 \to s_8, s_8 \to s_{11} \to s_{10},$ $s_{10} \to s_7 \to s_6, s_{12} \to s_5 \to s_4, s_{15} \to s_2 \to s_3$
C8	$s_1 \to s_2 \to s_3, s_3 \to s_4 \to s_5, s_5 \to s_6 \to s_7, s_7 \to s_8 \to s_9, s_9 \to s_{10} \to s_{11},$ $s_{11} \to s_{12} \to s_{13}, s_{13} \to s_{14} \to s_{15}, s_{15} \to s_{16} \to s_1$
C9	$s_1 \to s_2 \to s_3, s_3 \to s_4 \to s_5, s_5 \to s_6 \to s_7, s_7 \to s_8 \to s_9, s_9 \to s_{10} \to s_{11},$ $s_{11} \to s_{12} \to s_{13}, s_{13} \to s_{14} \to s_{15}, s_{15} \to s_{16} \to s_{17}, s_{17} \to s_{18} \to s_{19},$ $s_{19} \to s_{20} \to s_{21}, s_{21} \to s_{22} \to s_{23}, s_{23} \to s_{24} \to s_{25}, s_{25} \to s_{26} \to s_{27},$ $s_{27} \to s_{28} \to s_{29}, s_{29} \to s_{30} \to s_{31}, s_{31} \to s_{32} \to s_{33}, s_{33} \to s_{34} \to s_{35},\ s_{35} \to s_{36} \to s_1$
C10	$s_1 \to s_{40} \to s_{39}, s_3 \to s_{38} \to s_{37}, s_5 \to s_{36} \to s_{35}, s_7 \to s_{34} \to s_{33}, s_9 \to s_{32} \to s_{31},$ $s_{12} \to s_{21} \to s_{22}, s_{14} \to s_{23} \to s_{24}, s_{16} \to s_{25} \to s_{26}, s_{18} \to s_{27} \to s_{28},$ $s_{20} \to s_{29} \to s_{30}, s_{22} \to s_{19} \to s_{20}, s_{24} \to s_{17} \to s_{18}, s_{26} \to s_{15} \to s_{16},$ $s_{28} \to s_{13} \to s_{14}, s_{30} \to s_{11} \to s_{12}, s_{31} \to s_2 \to s_1, s_{33} \to s_4 \to s_3,\ s_{35} \to s_6 \to s_5, s_{37} \to s_8 \to s_7, s_{39} \to s_{10} \to s_9$

satellite assignments. The first column in Table 2 contains the name of the instance, and the second column gives the number of satellites (n). The third column gives the value of an optimal solution, and indicates (by "(i)") whether the linear programming (LP) relaxation of model (6)–(10) for the corresponding instance is integral. The fourth column gives the running time (in seconds) needed to solve the integer program (6)–(10). Columns 5 and 6 each give the value of a feasible solution: Column 5 reports the outcome of a network flow formulation followed by a local search procedure as described in Dutta and Tsiotras (2008), and Column 6 (called "Restricted") gives the best possible solution when moving satellites need to return to their original position.

Table 2 shows that an optimal solution is found within 1 second for the instances with size 18 or smaller, and in a few seconds for instances with size 40 or less. The LP relaxation yields an optimal solution for all but one instance tested. The optimal solutions reported here show that the solutions found by local search are quite good: with one exception (instance C8), the optimal solution is only 1-3 % better than the solution found by local search. When we compare the optimal solutions to the solutions with the added restriction that satellites need to return to their original position (which can be found in polynomial time), it turns out that the improvement is often significant.

The second set of instances was generated randomly taking into account specific characteristics of the real-life problem.

INFOR, Vol. 50, No. 2, May 2012, pp. 88–94
ISSN 0315-5986 | EISSN 1916-0615
DOI 10.3138/infor.50.2.088

TABLE 4.
Generation of P2P Instances

	Parameter	Value	Comment
1	Number of satellites	$16 \leq n \leq 56$	6 uniformly distributed values
2	Specific impulse	$220 \leq I_{sp} \leq 320$	5 uniformly distributed values
3	Fuel capacity	$20\% \leq \bar{f} \leq 30\%$	4 uniformly distributed values
4	Fuel content of fuel-sufficient satellites	$\bar{f}$	Fixed for a given selection of $\bar{f}$
5	Fuel content of fuel-deficient satellites	$0.05\bar{f} \leq f_{i,d} \leq 0.10\bar{f}$	Randomly chosen
6	Orbital locations of satellites	$0 \leq \Phi_i < 2\pi$	Uniformly distributed

Table 4 summarizes the parameters that were taken into account in order to generate the instances. Instances with realistic size range between 16 and 56 satellites. We only consider satellites that employ chemical propulsion systems, and more specifically, those which are currently being used for satellite station-keeping and attitude control purposes (Ley etal., 2009; Campbell and McCandles, 1996). For such propulsion systems, the specific impulse (I_{sp}) varies between 220 seconds and 320 seconds. For the fuel content of satellites, we consider satellites with a lifespan of at least 5 years, and use data obtained from (Maral and Bousquet, 2009; of Concerned Scientists Satellite Database) related to wet mass and dry mass of such satellites at Beginning-of-Life on orbit. The fuel content of the satellites range between 20% and 30%. In all instances there is an equal number of fuel-sufficient and fuel deficient satellites. In total, consideration of 6 choices of n, 5 choices of I_{sp} values, and 4 choices of maximum fuel content of satellites yields $6 \times 5 \times 4 = 120$ instances of the problem. For each of these instances, the optimal value is calculated using model (6)–(10). This solution value is compared to the solution obtained using the local search approach described in Dutta and Tsiotras (2008) and the optimal value of the restricted peer-to-peer strategy (i.e. each satellite needs to return to its original position). An overview of the results is given in Table 5. This table follows the format from Table 2, only now each entry is an average over 20 instances. Given a total number of satellites in the constellation, this average represents the mean refueling costs over different choices of engine and fuel content of satellites. In the second column is indicated between brackets how many of the instances were solved to optimality by solving the LP-relaxation. Most of the instances could not be solved to optimality by solving the LP-relaxation. However, on average, the optimality GAP was only 0.05% when not solved to optimality. The largest instances can be solved to optimality in less than 12 seconds and solutions are significantly better than the solutions obtained with local search. For instances with 16 satellites, the improvement is only around 4%, but the improvement is substantial when the number of satellites increases. In fact, the improvement is $>30\%$ when the number of satellites is around 50. The optimal solutions for the restricted peer-to-peer strategy can be found faster (since the problem is solvable in polynomial time, see Section 3), however, the costs of the E-P2P strategy are between 50% and 80% lower compared to the restricted version. Again, we can conclude that the improvement of E-P2P compared to restricted P2P is very significant. The running times for the E-P2P increase faster than for the restricted P2P (which was to be expected), they are, however, still reasonable for realistic instance sizes of the problem. Especially since running time is not a major concern in P2P applications.

5. CONCLUSIONS

It is shown that solving the so-called Egalitarian Peer-to-Peer (E-P2P) satellite refueling problem to optimality is a computationally hard problem. However, since most instances from practice are likely to contain 50 satellites or less, finding the solution with minimum fuel costs can be done using state-of-the-art integer programming algorithms. As a result, the implementation of E-P2P strategies is feasible for practical applications; all computations can be done on-board with relatively limited computational resources, thus opening the way to

TABLE 5.
Average costs and running times (dataset 2)

n	Opt	Running time (s)	Local Search	Restricted
16	13.575 (10\ 20)	0.322	14.107 (4%)	20.841 (54%)
24	19.037 (16\ 20)	0.582	21.854 (15%)	32.292 (64%)
32	24.622 (0\ 20)	1.643	29.226 (19%)	41.660 (69%)
40	30.186 (0\ 20)	3.573	37.370 (24%)	52.067 (72%)
48	35.735 (0\ 20)	6.432	46.812 (31%)	62.457 (75%)
56	41.282 (0\ 20)	10.941	55.398 (34%)	72.886 (77%)

INFOR, Vol. 50, No. 2, May 2012, pp. 88–94
ISSN 0315-5986 | EISSN 1916-0615
DOI 10.3138/infor.50.2.088

completely decentralized implementations of the baseline E-P2P algorithms.

ACKNOWLEDGEMENTS

This research was supported by OT Grant OT/07/015.

REFERENCES

Burkard, R. E., Dell'Amico, M., and Martello, S., (2009), *Assignment Problems*, Siam.

Burkard, R. E., Rudolf, R., and Woeginger, G. J., (1996), "Three-dimensional axial assignment problems with decomposable costs", *Discrete Applied Mathematics*, 65, 123–139.

Campbell, B. A. and McCandles, S. W., (1996), *Introduction to Space Sciences and Spacecraft Applications*. Gulf Publishing Company.

Dutta, A. and Tsiotras, P., (2006), "Asynchronous optimal mixed p2p satellite refueling strategies", *Journal of the Astronautical Sciences*, 54: 543–565.

Dutta, A. and Tsiotras, P., (2008), "An egalitarian peer-to-peer satellite refueling strategy", *Journal of Spacecraft and Rockets*, 45: 608–618.

Garey, M. R. and Johnson, D. S., (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co., New York.

Ley, W., Wittmann, K., and Hallmann, W., (2009), *Handbook of Space Technology*, John Wiley and Sons, Ltd.

Maral, G. and Bousquet, M., (2009), *Satellite Communication Systems: Systems, Techniques and Technology*, John Wiley and Sons, Ltd.

Shen, H. and Tsiotras, P., (2005), "Peer-to-peer refueling for circular satellite constellations", *AIAA Journal of Guidance, Control, and Dynamics*, 28: 1220–1230.

Spieksma, F. C. R., (2000), "Multi index assignment problems: Complexity, approximation, applications", In Pardalos, P. and Pitsoulis, L. (eds), *Nonlinear Assignment Problems*, Kluwer, Dordrecht.

Union of Concerned Scientists Satellite Database, http://www.ucsusa.org.