A Logistics Analysis and Control of Distributed Life Support Systems for High-Mobility Exploration

Hiroyuki Miyajima[1]
Tokyo Jogakkan College, Machida, Tokyo, 194-0004

The purpose of this research is to provide a method for analyzing the operations of logistics carriers in making dynamic resource allocations in high-mobility explorations on the lunar surface. A simulator for ground test facilities of an Advanced Life Support System was developed, and the Lagrangian Decomposition and Coordination (LDC) method was integrated into operations planning for the simulator. A previous paper described this approach and its application to the logistics carrier operations planning using a one-dimensional model in a logistics network on the lunar surface. This paper describes the expansion of the LDC formulation from one-dimensional to two-dimensional and application of the method to the control of distributed life support systems for high-mobility exploration. A two-dimensional Dynamic Programming (2DP) method is developed to yield exact solutions of the partial problems, and the capability of the algorithm is confirmed by the numerical simulations.

Nomenclature

A_{ik} = relationship matrix between rover *i* and logistics carrier *k* (not connected if = 0; connected if = 1)
B_{jk} = relationship matrix between movement *j* and logistics carrier *k* (not connected if = 0; connected if = 1)
Ch_{CO2} = carbon dioxide discharge rate of crew
Ch_{O2} = oxygen consumption rate of crew
c_k = unit movement cost of logistics carrier *k*
$dc_{(x,y)kt}$ = position of logistics carrier *k* at time *t*
$dr_{(x,y)it}$ = position of rover *i* at time *t*
$D_{(x,y)L}$ = lower bound position of logistics carrier
$D_{(x,y)U}$ = upper bound position of logistics carrier
El_{M1} = production rate of electrolysis reaction of Habitation Module 1
h_i = unit supply cost of rover *i* at time *t*
h_k = unit supply cost of logistics carrier *k* at time *t*
i = number of rover (i=1, 2, …, I)
j = number of movement (j=1, 2, …, J)
k = number of logistics carrier (j=1, 2, …, K)
l = Lagrangian function
m_{it} = amount of stored mass on rover *i* at time *t*
m_{kt} = amount of stored mass on logistics carrier *k* at time *t*
M_{M1O2} = amount of oxygen in cabin of Habitation Module 1
M_{R1O2} = amount of oxygen in cabin of Rover 1
M_{R2O2} = amount of oxygen in cabin of Rover 2
M_{Ui} = upper amount of stored mass on rover *i* at time *t*
M_{Uk} = upper amount of stored mass on logistics carrier *k* at time *t*
N_{Mh} = number of crew in Habitation Module 1
N_{Rh} = number of crew on Rover
pC_{sa} = carbon production rate of CO_2 reduction
pH_{2El} = hydrogen production rate of electrolysis reaction
pO_{2El} = oxygen production rate of electrolysis reaction
pO_{2Sa} = oxygen production rate of CO_2 reduction

[1] Associate Professor, Faculty of Liberal Arts for Global Studies and Leadership, 1105 Tsuruma, Senior Member.

rH_2O_{EI} = water consumption rate of electrolysis reaction
rO_{2M1} = amount of oxygen supply to cabin in Habitation Module 1
rO_{2R1} = amount of oxygen supply to cabin in Rover 1
rO_{2R2} = amount of oxygen supply to cabin in Rover 2
Sa_{M1} = production rate of sabattier reaction of Habitation Module 1
sw_s = amount of oxygen supply
t = time (t=1, 2, …, T)
Ta_{L1O2} = amount of oxygen in oxygen tank of Logistics Carrier
$Ta_{L1O2max}$ = capacity of oxygen tank in Logistics Carrier
Ta_{M1O2} = amount of oxygen in oxygen tank of Habitation Module 1
Ta_{M1W} = amount of water in water tank of Habitation Module 1
Ta_{R1O2} = amount of oxygen in oxygen tank of Rover 1
$Ta_{R1O2max}$ = capacity of oxygen tank in Rover 1
Ta_{R2O2} = amount of oxygen in oxygen tank of Rover 2
$Ta_{R2O2max}$ = capacity of oxygen tank in Rover 2
$v^k_{(x,y)jt}$ = velocity of logistics carrier k due to movement j in time t
Wfc = amount of water supply from fuel cell
Wh = amount of water consumption of crew
Δm_{it} = amount of change of stored mass on rover i in time t
Δm_{kt} = amount of change of stored mass on logistics carrier k in time t
$\delta_{(x,y)jt}$ = decision index for movement j at time t (go back =-1; stop =0; go forward=+1)
λ = Lagrangian multipliers

I. Introduction

One of the lasting lessons of the Apollo program is that surface mobility is key to improving the efficiency of manned exploration of the lunar surface[1]. In the lunar exploration plan which was recently presented, it is proposed that crews explore the lunar surface at ranges reaching beyond a hundred kilometers from the habitation module with the use of a logistics carrier and pressurized rovers[2]. Included in the lunar outpost will be habitation modules, logistic carriers, in situ resource utilization (ISRU), extravehicular activity (EVA) systems, and the pressurized rovers. With a combination of logistics carriers and manned pressurized rovers, the crew can safely explore a much wider range as compared to without the logistics carriers. During the expedition, life support systems and supplies for life support will not be exhausted at the lunar outpost, but they may be used and may be exhausted at individual points within the range. When the life support system and supplies are distributed in a wide area, it is important to provide a logistics network which resupplies the distributed life support systems. If supplies are inappropriately stored at any point, regardless of whether or not there is a sufficient amount of supplies on the lunar surface, this inappropriate storage will inevitably lead to a lack of supplies.

The purpose of this research is to provide a method for logistics carriers to make such dynamic resource allocations. I developed a simulator for ground test facilities of Advanced Life Support Systems (ALSS), and integrated the Lagrangian Decomposition and Coordination (LDC) method into operations planning for the simulator[3,4,5]. The previous paper[6] described its application to the logistics carrier operations planning using a one-dimensional model in a logistics network on the lunar surface. This paper describes the expansion of the LDC formulation from one-dimensional to two-dimensional and application of the method to the control of distributed life support systems for high-mobility exploration on the lunar surface. In this research, two-dimensional Dynamic Programming (2DP) is developed to solve exact solutions of partial problems, which is each logistics carrier's operation, and the capability of the algorithm is confirmed by numerical simulations with regards to the expanding of the operation time and movement coverage.

II. Formulation of Logistics Network

Life support systems and the supplies for life support are movable and operated as decentralized units at multiple exploration sites to facilitate the exploration of a wider area on the lunar surface with the assistance of logistics carriers and pressurized rovers. The previous paper[6] described the logistics network that distributes supplies to manned pressurized rovers. Logistics carriers and rovers explore multiple points within the range of the habitation module. Although this problem is formulated as a simple Vehicle Routing Problem (VRP) in a logistics exercise, there are differences between lunar and terrestrial terrains, so that it is not simply a shortest route problem or a cost minimization problem. In view of the operation of life support systems in the logistics network, this becomes a Vehicle Scheduling Problem (VSP) for the timely supply of distributed life support systems. Thus the time-space network becomes larger than individual time problems or space problems. Moreover, it is necessary to respond to possible changes in the exploration plan during the expedition, which is dynamic scheduling.

This section illustrates the expansion from the one-dimensional to the two-dimensional model. Both Cartesian coordinates and polar coordinates were considered for the two-dimensional model: Cartesian coordinates are preferred because of their relative simplicity and ease of describing movement distances. Fig. 1 shows the logistics network and movement model using Cartesian coordinates. Here logistics carrier operations planning is formulated using the LDC expanded to two-dimensions. In my research, two-dimensional Dynamic Programming (2DP) is developed to obtain exact solutions of partial problems, that is, each logistics carrier's operation. Since the complexity of the whole problem increases as the polynomial order of the partial problems, it is desirable to decrease the complexity of partial problems to solve large-scale problems. As the complexity of a DP increases exponentially with the scale of the problem even for one variable, 2DP is not generally used. There has only been research in image matching[7].

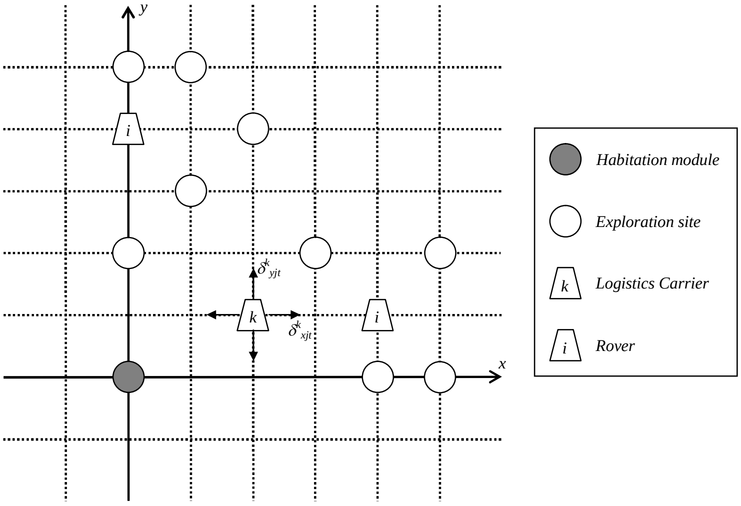


Figure 1 Logistics Network and Movement Model.

III. Vehicle Scheduling Algorithm

A. Formulation of the Lagrangian Decomposition and Coordination Method

Figure 1 shows the movement model of a logistics carrier and rovers in two-dimensional movements using Cartesian coordinates whose origin is a habitation module. Assume that the objective function optimized is the sum of three terms: the first term measures the normalized cost of the logistics carrier's movement, the second term is the normalized supply cost to the rovers, and third term is the normalized supply cost to the logistics carriers. This is expressed in Eq. (1):

$$\begin{aligned} min \quad \sum_{t=1}^{T}\sum_{k=1}^{K}\Bigg[& \sum_{j=1}^{J} c_k \sqrt{(\delta^k_{xjt})^2 + (\delta^k_{yjt})^2} + \sum_{i}^{I} h_i A_{ik} \sqrt{\left((dr_{xit}/D_{xU})^2 + (dr_{yit}/D_{yU})^2\right)}\left((M_{Ui} - m_{it})/M_{Ui}\right) \\ & + h_k \sqrt{\left((dc_{xkt}/D_{xU})^2 + (dc_{ykt}/D_{yU})^2\right)}\left((M_{Uk} - m_{kt})/M_{Uk}\right)\Bigg] \end{aligned} \tag{1}$$

where $\delta^k_{(x,y)jt}$ is the movement index at the x and y of the logistic carrier k's movement j at time t. As the $\delta^k_{(x,y)jt}$ has three values which are -1, 0, and 1 for each x and y, and the logistics carrier has nine movements whose directions are right, upper right, upper, upper left, left, lower left, lower, lower right, and stop, which is a result of the combination of x and y movements. $dr_{(x,y)it}$ is position of rover i at time t, $D_{(x,y)U}$ is upper bound position of logistics carrier, M_{Ui} is upper bound stored mass of rover i, and m_{it} is stored mass of rover i on time t, $dc_{(x,y)kt}$ is position of logistics carrier k at time t, M_{Uk} is upper bound stored mass of logistics carrier k, and m_{kt} is stored mass of logistics carrier k at time t.

In the first term the unit movement cost, c_k of logistics carrier k is multiplied by the movement j's normalized distance at time t, SQRT($(\delta^k_{xjt})^2+(\delta^k_{yjt})^2$). In the second term the unit supply cost, h_i of rover i is multiplied by the relationship matrix A_{ik}, between rover i and logistics carrier k, the distance of logistics carrier k from habitation module normalized upper bound position, SQRT($(dr_{xit}/D_{xU})^2+(dr_{yit}/D_{yU})^2$), and the exhaustion of supply in rover i normalized by the upper bound stored mass at time t, $((M_{Ui}-m_{it})/M_{Ui})$. In the third term the unit supply cost, h_k of logistics carrier k is multiplied by its normalized distance from the habitation module, SQRT($(dc_{xkt}/D_{xU})^2+(dc_{ykt}/D_{yU})^2$), and by the normalized exhaustion of supply in logistics carrier k at time t, $((M_{Uk}-m_{kt})/M_{Uk})$.

As constraint conditions for Eq. (1), the change of the logistics carrier position, the lower bound of the logistics carrier's position, the upper bound of the logistics carrier's position, the change of stored mass on the Rovers and Logistics Carrier, and the competition for the logistics carrier, that is, it is not allowed to go simultaneously in different directions, are defined by Eqs. (2) to (7).

$$subject\ \ to\quad dc_{(x,y)k,t+1} = dc_{(x,y)kt} + \delta^k_{(x,y)jt}\cdot v^k{}_{(x,y)jt}\quad \forall j,k,t \tag{2}$$

$$dc_{(x,y)kt} \ge D_{(x,y)L}\quad \forall k,t \tag{3}$$

$$dc_{(x,y)kt} \le D_{(x,y)U}\quad \forall k,t \tag{4}$$

$$m_{i,t+1} = m_{it} + \Delta m_{it}\quad \forall i,t \tag{5}$$

$$m_{k,t+1} = m_{kt} + \Delta m_{kt}\quad \forall k,t \tag{6}$$

$$\sum_{j}^{J}\delta^k_{(x,y)jt}B_{(x,y)jk} \le 1\quad \forall k,t \tag{7}$$

where Δm_{it} is the amount of change of stored mass on rover i, Δm_{kt} is the amount of change of stored mass on logistics carrier k and $B_{(x,y)jk}$ is a relationship matrix between movement j and logistics carrier k.

B. Lagrangian relaxation

Next, the optimization problem with constraints is replaced by one without constraints using Lagrangian multipliers. This is referred to as a Lagrangian relaxation. That is, the formulation of an optimization problem is changed from a strict one in which constraints must be satisfied to a relaxed one in which constraint violations are minimized[8,9]. This is achieved in Eqs. (1) to (7) by introducing Lagrangian multipliers λ, which may be thought of as use fees or a way of measuring competition for the use of a logistics carrier[10,11], Thus Eq. (7) is relaxed as follows:

$$\begin{aligned} min\quad l = \sum_{t=1}^{T}\sum_{k=1}^{K}\Bigg[\sum_{j=1}^{J}c_k\sqrt{(\delta^k_{xjt})^2+(\delta^k_{yjt})^2} + \sum_{i}^{I}h_iA_{ik}\sqrt{\left((dr_{xit}/D_{xU})^2+(dr_{yit}/D_{yU})^2\right)}\left(\left(M_{Ui}-m_{it}\right)/M_{Ui}\right) \\ +h_k\sqrt{\left((dc_{xkt}/D_{xU})^2+(dc_{ykt}/D_{yU})^2\right)}\left(\left(M_{Uk}-m_{kt}\right)/M_{Uk}\right)\Bigg] \\ +\sum_{t=1}^{T}\lambda_t\sum_{k=1}^{K}\sum_{j}^{J}\delta^k_{(x,y)jt}B_{(x,y)jk} \end{aligned} \tag{8}$$

subject to Eqs. (2) - (6).

C. Decomposition to partial problems

A decision variable vector δ related to Eqs. (8) and (2) to (6), is decomposed for the individual logistics carrier. Hence, minimizing the problem expressed in Eqs. (8) and (2) to (6) is equivalent to independently minimizing partial problems which are expressed using Eqs. (9), and (2) to (6) relative to logistics carrier k.

$$
\begin{aligned}
min \quad l_k = \sum_{t=1}^{T}\Bigg[& \sum_{j=1}^{J} c_k \sqrt{(\delta_{xjt}^k)^2 + (\delta_{yjt}^k)^2} + \sum_{i}^{I} h_i A_{ik} \sqrt{\left((dr_{xit}/D_{xU})^2 + (dr_{yit}/D_{yU})^2\right)} \left((M_{Ui} - m_{it})/M_{Ui}\right) \\
& + h_k \sqrt{\left((dc_{xkt}/D_{xU})^2 + (dc_{ykt}/D_{yU})^2\right)} \left((M_{Uk} - m_{kt})/M_{Uk}\right) \Bigg] \\
& + \sum_{t=1}^{T} \lambda_t \sum_{j}^{J} \delta_{(x,y)jt}^k B_{(x,y)jk}
\end{aligned} \qquad (9)
$$

Although Eqs. (9), and (2) to (6) represent scheduling problems corresponding to individual logistics carriers, these problems are related to each other in that the result of one scheduling influences another since there are terms related to interference of supply and competition for a logistics carrier.

D. Complexity

In this formulation, since a variable has three possible actions, complexity for one variable is 3^t, and complexity for two variables is 3^{2t}. When a 28-stage problem is solved using DP, the theoretical complexities are 3^{28} and 9^{28} respectively. If the number of states increases, the complexity approaches a theoretical limit. In the actual calculation, since only the states permitted by the constraints are calculated, the complexity is decreased.

IV. Description of Distributed Life Support System Operations

In Lunar Scenario 12.0[2] the lunar outpost consists of movable logistics carriers and rovers, a fixed habitation module, communication systems, power systems, ISRU, logistics and parts, and support systems. Four crew members stay at the base for 7 to 180 days. Mobility is emphasized in this scenario and crew members go on expeditions in a wide area using the movable systems for 14 to 28 days.

This example solves an oxygen allocation problem with the oxygen supply system diagramed in Fig. 2, as an example of a confined logistics network in the lunar outpost. The logistics network consists of a Habitation Module, a Logistics Carrier, and two Rovers. The oxygen supply is expressed by Eqs. (10) through (20). Eqs. (10) through (14) refer to the Habitation Module, Eq. (15), the Logistics Carrier, Eqs. (16) through (19) the Rovers, and Eq. (20), a supply model among the Habitation Modules, Logistics Carrier, and Rovers. The supply system has five lines: from the Habitation Module to Logistics Carrier (sw_1), Rovers 1 (sw_2) and 2 (sw_3), and from Logistics Carrier to Rovers 1 (sw_4) and 2 (sw_5). The supply system can operate when the Logistics Carrier is at the same place as a Rovers, or the Logistics Carrier or Rovers are at the Habitation Module. The operation schedule of the Logistics Carrier is calculated considering both the Rover positions and supply possibilities.

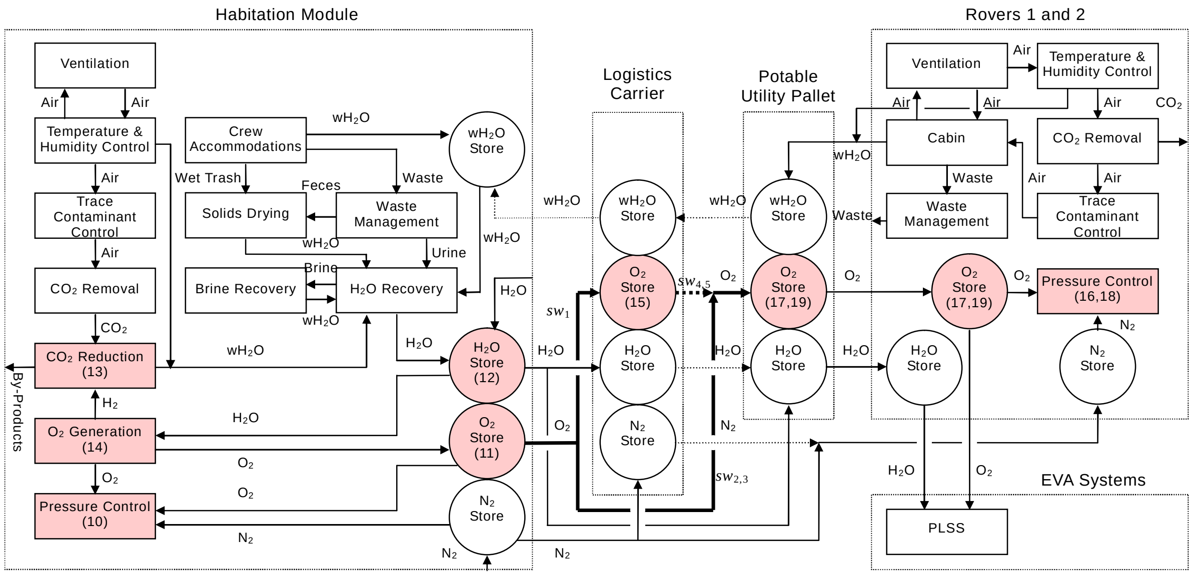


Figure 2 Oxygen Supply Systems in the Lunar Outpost.
Parenthetic numbers in a shaded area correspond to Eqs. (10) - (19). sw_1, sw_2, sw_3, sw_4, and sw_5 in Fig. 2 correspond to Eq. (20).

Table 1 shows the initial conditions used in this calculation. The units of the position and the amount of oxygen are defined: 1 [day] as the distance of 1 day's movement, and 1 [CM-day] as the amount of oxygen consumed per crewmember in 1 day. Table 1 (a) and (b) show fixed numbers of crew in the Habitation Module and Rovers. Four crewmembers can stay at the Habitation Module, and the two groups of two crewmembers can explore with the two-seated Rovers at the same time. Here, as shown in Table 1 (c) and (d), 28 day operation schedules of Rovers 1 and 2 are given in advance, and the operation schedule of the Logistics Carrier is solved by the calculation. This schedule assumes 1000km movement in 300km x 300km square around the Habitation Module. A material is recycled in the Habitation Module and no material is recycled in the Rovers, where waste water is only collected and carbon dioxide is wasted. Table 1 (e) shows the initial starting position of the Logistics Carrier. Table 1 (f) to (i) show the initial stored oxygen for the Habitation Module, Logistics Carrier, and Rovers 1 and 2. The movement cost of Table 1 (j) prevents the Logistics Carrier from repeatedly going forward and back, and the supply cost of Table 1 (k) prevents the Logistics Carrier from remaining far from the Habitation Module when it has insufficient oxygen.

Table 1 Initial Condition of Habitation Module, Logistics Carrier, and Rovers.

	Initial Conditions	Descriptions
(a)	Fixed number of crew in Habitation Module N_{Mh} [person]	4
(b)	Fixed number of crew in Rover N_{Rh} [person]	2
(c)	Positions of Rover 1 [days] dr_{x1} = { 0,1,2,3,4,5,6,6,6,5,5,5,5,6,6, 6,6,5,5,5,5,6,6,5,4,3,2,1,0} dr_{y1} = { 0,0,0,0,0,0,0,1,1,1,1,2,2,2,2, 3,3,3,3,4,4,4,4,5,4,3,2,1,0}	(dr_{x1}, dr_{y1})
(d)	Positions of Rover 2 [days] d_{x2} = { 0,0,0,0,0,0,0,1,1,1,1,2,2,2,2, 3,3,3,3,4,4,4,4,5,4,3,2,1,0} dr_{y2} = { 0,1,2,3,4,5,6,5,5,4,4,5,5,6,6, 5,5,4,4,5,5,6,6,5,4,3,2,1,0 }	(dr_{x2}, dr_{y2})
(e)	Initial starting position of Logistics Carrier [days]	(0,0)
(f)	Stored oxygen in Habitation Module 1[CM-days]	14
(g)	Stored oxygen in Logistics Carrier [CM-days]	56
(h)	Stored oxygen in Rover 1 [CM-days]	28
(i)	Stored oxygen in Rover 2 [CM-days]	28
(j)	Movement cost c_k [-]	0.1
(k)	Supply cost h_i and h_k [-]	1

Oxygen Supply System Models

Habitation Module

$$M_{M1O2}(t+1) = M_{M1O2}(t) + rO_{2M1}(t) - Ch_{O2} \cdot N_{Mh} \tag{10}$$

$$Ta_{M1O2}(t+1) = Ta_{M1O2}(t) + pO_{2Sa}(t) + pO_{2El}(t) - rO_{2M1}(t) - sw_1 - sw_2 - sw_3 \tag{11}$$

$$Ta_{M1W}(t+1) = Ta_{M1W}(t) + Wfc(t) - rH_2O_{El}(t) - Wh(t) \cdot N_{Mh} \tag{12}$$

$$Sa_{M1}(t+1) = Sa_{M1}(t) - \left[pO_{2sa}(t) + pC_{sa}(t) \right] + Ch_{CO2} \cdot N_{Mh} \tag{13}$$

$$El_{M1}(t+1) = El_{M1}(t) + \left[rH_2O_{El}(t) - pO_{2El}(t) - pH_{2El}(t) \right] \tag{14}$$

Logistics Carrier

$$Ta_{L1O2}(t+1) = Ta_{L1O2}(t) + sw_1 - sw_4 - sw_5 \tag{15}$$

Rover 1

$$M_{R1O2}(t+1) = M_{R1O2}(t) + rO_{2R1}(t) - Ch_{O2} \cdot N_{Rh} \tag{16}$$

$$Ta_{R1O2}(t+1) = Ta_{R1O2}(t) - rO_{2R1}(t) + sw_2 + sw_4 \tag{17}$$

Rover 2

$$M_{R2O2}(t+1) = M_{R2O2}(t) + rO_{2R2}(t) - Ch_{O2} \cdot N_{Rh} \tag{18}$$

$$Ta_{R2O2}(t+1) = Ta_{R2O2}(t) - rO_{2R2}(t) + sw_3 + sw_5 \tag{19}$$

Supply Model among Habitation Module, Logistics Carrier, and Rovers

$$sw_s = \begin{cases} if\,(dc_1 = 0) & sw_1 = Ta_{L1O2\max} - Ta_{L1O2}(t) \\ if\,(dr_1 = 0) & sw_2 = Ta_{R1O2\max} - Ta_{R1O2}(t) \\ if\,(dr_2 = 0) & sw_3 = Ta_{R2O2\max} - Ta_{R2O2}(t) \\ if\,(dc_1 = dr_1) & sw_4 = Ta_{R1O2\max} - Ta_{R1O2}(t) \\ if\,(dc_1 = dr_2) & sw_5 = Ta_{R2O2\max} - Ta_{R2O2}(t) \end{cases} \tag{20}$$

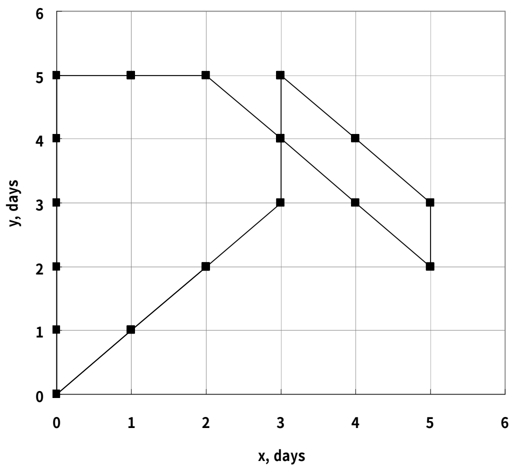
V. Results of the Logistics Carrier Operations

A. Movement Trajectory

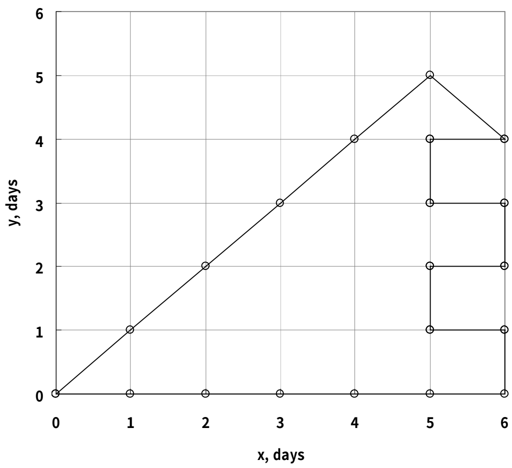
Figure 3 shows positions of Logistics Carrier, and Rovers 1 and 2 using x-y coordinates in a 28 day operation. Fig. 3 (a) shows the trajectory of the Logistics Carrier which is solved by this method. In Fig. 3 (b) and (c) show the trajectories of Rover 1 and 2, which are predetermined by the exploration schedule. As it is not easy to find the relationship between the positions of the Logistics Carrier and Rovers at a specific time in Fig. 3 (a)-(c) using an x-y coordinate, the graph is changed from the x-y coordinate to a time-(10x+y) coordinate in Fig. 3 (d), which shows the Logistics Carrier moving with the same position as Rover 2 from day 1 to day 5, and day 7. Next, it moves with the position of Rover 1 on days 11 and 12 to supply to Rover 1. Then, it moves back to the position of Rover 2 on days 15 to 18 to supply to Rovers 2. Finally, it moves back to the Habitation Module on day 24 to restock. It also supplies both Rovers on day 26.

B. Changes of Stored Oxygen

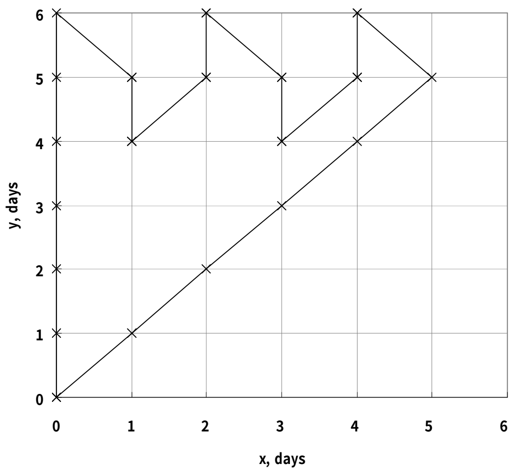
Figure 4 shows the amount of changes in oxygen in the Habitation Module (Mm1), Logistics Carrier (Mc), and Rovers 1 (Mr1) and 2 (Mr2). When the Logistics Carrier is at the same place as the Rovers, it can deplete Mc to replenish Mr1 or Mr2. The Logistics Carrier supplies oxygen to Rover 1 on days 11 and 26, Rover 2 on days 7, 15, and 26. Consequently, Rovers 1 and 2 can be operated without any lack of oxygen during the expedition. Since, stored oxygen in the Habitation Module is maintained by producing oxygen there is no depletion of oxygen.



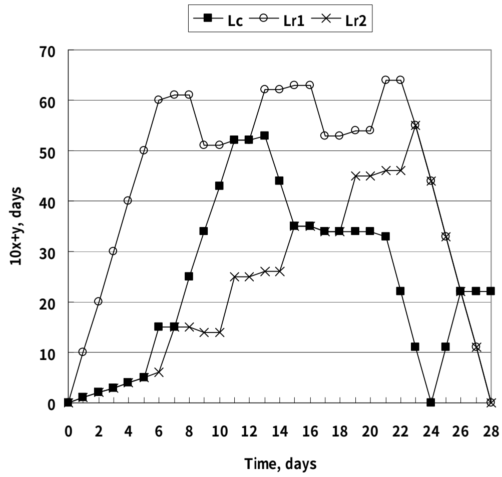
(a) Trajectory of Logistics Carrier (Lc).



(b) Trajectory of Rover 1(Lr1).



(c) Trajectory of Rover 2(Lr2).



(d) Time-(10x+y) coordinate graph of trajectory.

Figure 3 Positions of Logistics Carrier (Lc), Rover 1(Lr1), and 2 (Lr2).

The unit of distance [days] means distance which Logistics Carrier and Rovers can move in a day.

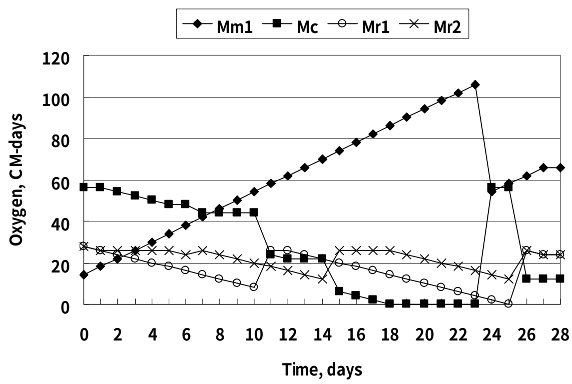
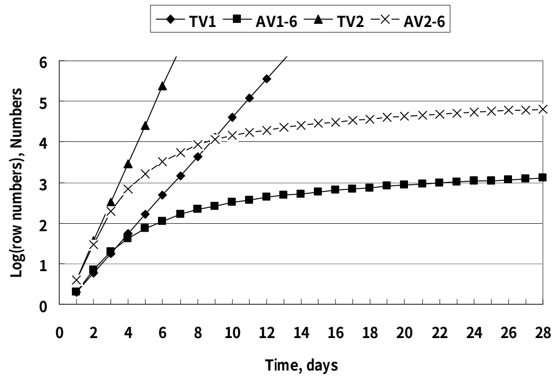


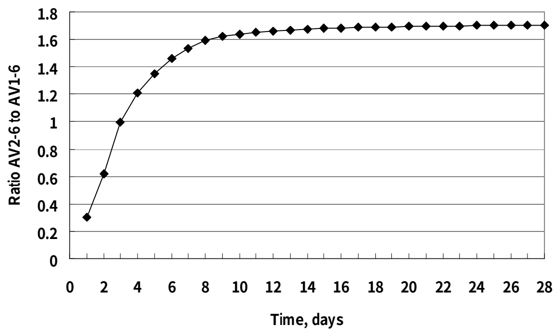
Figure 4 Amount of changes in oxygen of Habitation Module (Mm1), Logistics Carrier (Mc), Rovers 1 (Mr1), and 2 (Mr2).

The unit of amount of oxygen [CM-days] means oxygen amount that a crew consumes in a day.



(a) Comparison between theoretical and actual row numbers.

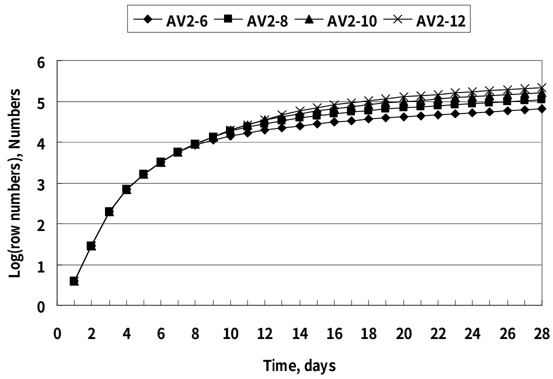
The actual memory usage in the actual calculations is described by the row numbers of the two dimensional array used in 2DP. The legend shows theoretical (TV1) and actual (AV1-6) row numbers of one variable, and theoretical (TV2) and actual (AV2-6) row numbers of two variables.



(b) Comparison of actual row numbers between one and two variables.

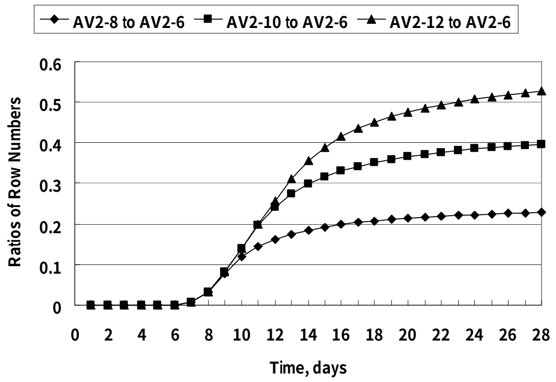
The ratio of A to B expressed as log(A) - log(B). The ratio of actual row number for two variables (AV2-6) to the number for one variable (AV1-6) is used in the comparison.

Figure 5 Comparison of memory usage between one and two variables.



(a) Comparison of row numbers between state numbers 6, 8, 10, and 12.

The actual memory usage in the actual calculations is described by the row numbers of the two dimensional array used in 2DP. In the legend of AV2-6, AV2-8, AV2-10, and AV-12 refer to states 6, 8, 10, and 12.



(b)Row number ratios of state numbers 8, 10, and 12 to state number 6.

The ratio of A to B expressed as log(A) - log(B). The ratios of actual row numbers of state numbers 8, 10, and 12 (AV2-8, AV2-10, and AV2-12) to that of state 6 (AV2-6) are used in the comparison.

Figure 6 Comparison of memory usage when the state number is increased.

Table 2 Comparison of calculation times when the x and y state numbers are changed from 6 to 8, 10, and 12 in one and two variables.

Computation Conditions: Dell Vostro 200 Intel® Core(TM)2 Duo CPU E8300 @2.83GHz, RAM 4.00MB (3.24MB Available), OS Windows Home Premium SP1. The program is developed by Visual Studio C++ 2008.

State numbers of x and y	6	8	10	12
Times of one variable [msec]	55	63	72	79
Ratios of the state to 6 [-]	1.00	1.15	1.31	1.44
Times of two variables [msec]	7947	24155	54647	99892
Ratios of the state to 6 [-]	0.97	2.95	6.68	12.21

C. Complexity Comparison

Figure 5 shows a comparison of the memory usage between one and two variables in this method. The memory usage in the actual calculation is represented by a row number of the two dimensional array used in 2DP. As the row number is in the millions, the number is expressed by common logarithms in the figure. Fig. 5 (a) compares the theoretical and actual row numbers. Here, theoretical row number means row number without constraints. Fig. 5 (b) shows a comparison of the actual row numbers using one and two variables. The legend in Fig. 5 (a) shows theoretical (TV1) and actual (AV1-6) row numbers of one variable, and theoretical (TV2) and actual (AV2-6) row numbers of two variables. Fig. 5 (a) shows the results when the state number is six. If the state number increases, the row number will approach the theoretical number. Comparing theoretical and actual numbers in Fig. 5 (a), as the complexity can be suppressed by using constraints in 2DP, an exponential increase of complexity does not happen in the actual calculation. In the case of expanding from one to two variables, as the row number of AV2-6 is 1.7 times of AV1-6 at t = 28 in Fig. 5 (b), two variables can be calculated using $10^{1.7}$ times the row number of one variable. The whole usage of memory is the sum of each time row number.

Figure 6 shows the memory usage when the state number at the x-y coordinates is increased, which means increasing the width and resolution of the x-y plane. Fig. 6 (a) compares the row numbers between state numbers 6, 8, 10, and 12. Fig. 6 (b) shows row number ratios of state numbers 8, 10, and 12 to state number 6. Fig. 6 (a) shows that the increase of states does not increase memory usage exponentially. Fig. 6 (b) shows when the x and y states increase number from 6, to 8, 10, and 12, the increases in memory usage are $10^{0.23}$, $10^{0.4}$, and $10^{0.53}$. That is, the increase of state numbers by 2, 4, or 6 only slightly increases the memory usage.

Next, Table 2 shows the calculation time when the x and y state numbers are changed from 6 to 8, 10, and 12 in one and two variables. The time only means that the time to calculate the problem by using 2DP which does not include the time to deal with graphics. Although the result shows that the time for calculating two variables increases over that of one variable, the time of two variables and 12 states takes less than 2 minutes.

VI. Conclusion

This paper described the operations planning method of the Logistics Carrier formulated using LDC in a logistics network for high-mobility exploration on the lunar surface. In this research, 2DP was developed to yield exact solutions of the partial problems, which is each logistics carrier's operation. The capability of the algorithm is confirmed by numerical simulations in which the operation time and movement coverage were expanded. It is further confirmed that the complexity of this method does not increase exponentially when the operation time becomes longer and the movement coverage becomes wider. This method has features that the complexity does not increase as the number of Rovers increases, and it increases only as the polynomial order of the number of Logistics Carriers. Based on these results, this method's extension to larger logistics networks is strongly indicated.

References

[1]Bagdigian R. M., Challenges with Deploying and Integrating Environmental Control and Life Support Functions in a Lunar Architecture with High Degrees of Mobility, 2009-01-2481, SAE Technical Paper Series, 2009.

[2]Kennedy K. J., Toups L. D., and Rudisill M., Constellation Architecture Team-Lunar Scenario 12.0 Habitation Overview, JSC-CN-19362, 2010, NASA Technical Report Server, cited 10 August 2010.

[3]Miyajima H., Abe K., Hirosaki T., and Ishikawa Y., Design of Intelligent Control Software for Mini-Earth, 2006-01-2123, SAE Technical Paper Series, 2006.

[4]Miyajima H., Abe K., Hirosaki T., and Ishikawa Y., Development of Advanced Life Support Systems Control Software Considering Computational Effort and Mathematical Validity, 2007-01-3025, SAE Technical Paper Series, 2007.

[5]Miyajima H., Abe K., Hirosaki T., and Ishikawa Y., Development of Advanced Life Support Systems Control Software Integrating Operators' Empirical Knowledge, 2008-01-1973, SAE Technical Paper Series, 2008.

[6]Miyajima H., An Operations Management Method for a Logistics Network of Distributed Life Support Systems on the Lunar Surface, AIAA 2010-6226, 2010.

[7]Miyazaki H., Uchida S., and Sskoe H., Fast Elastic Image Matching Algorithm Based on Coarse-to-Fine DP, Meeting on Image Recognition & Understanding, Vol. 2, pp. 423-428, 2004.

[8]Bertsekas, D. P., Dynamic Programming and Optimal Control I, II, Athena Scientific, 1995.

[9]Lasdon, L. S., Optimization Theory for Large Systems, Dover Publications, 2002.

[10]Zhang Y., Luh P. B., Yoneda K., Kano T. and Kyoya Y., Mixed-model Assembly Line Scheduling Using the Lagrangian Relaxation Technique, IIE Transactions 32(2), pp.125-134, 2000.

[11]Kuroda M. and Enomoto M., Usefulness of Lagrangian relaxation approaches to production scheduling under a dynamic environment, JAPAN-USA Symposium on Flexible Automation, Vol.2, pp.899-905, 1998.