

A Study on Automation of Image Collection Planning

Jaejoong Han[†], Kyungjin Jung, Jaeseung Choi, Sunghee Kwak, and Moongyu Kim

Ground Systems Division, Satrec Initiative Co.

Abstract : One of main concerns of operators of the Earth observation satellite is taking images as many as possible under the constraints of satellite resources during fixed period. In order to achieve this goal, satellite operators are strongly required to generate the optimized image collection plans, and it is a very time consuming process to achieve an optimized image collection plan when it is done by manual. This paper suggests automation of image collection planning based on the dynamic programming algorithm to reduce the time required for image collection planning. The validity of the proposed method is tested using operating satellite system and the result is given in this paper.

Key Words : Image Collection Plan, Automation, Dynamic Programming, Earth Observation Satellite

1. Introduction

Many number of Earth observation satellites have been launched and more number is expected as the interests on the remote sensing is increasing.

The satellite system is a very expensive so that the common concern of satellite operator is to fully utilize all of satellite resources available. In case of Earth observation satellite, the objective is to acquire as many images as possible under constraints such as limited memory, downlink time, power, orbit, thermal constraints, sun illumination and etc. For commercial Earth observation satellite, this requirement is more emphasized.

Conventionally, the optimization of imaging plan is done by manual. This is a very time-consuming procedure. In this paper, an automatic ICP generation

procedure is proposed. (William Martin, 2002)

In Chapter 2, the conventional ICP generation procedure is explained and the problem is identified. Chapter 3 explains four components of proposed approach, namely, Selector, Decomposer, Scorer and Scheduler in detail. The test result will be presented in Chapter 4, and the conclusion and future work will be given in Chapter 5 and 6 respectively.

2. Conventional Icp Generation

Fig. 1 shows a flow of conventional manual image collection plan generation. The term "Request" is used to refer to customer's request, "ICP" stands for an image collection plan, and "ICP Set" means a collection of ICP.

Received December 10, 2010; Revised February 4, 2011; Accepted February 10, 2011.

[†] Corresponding Author: Jaejoong Han (jjhan@satreci.com)

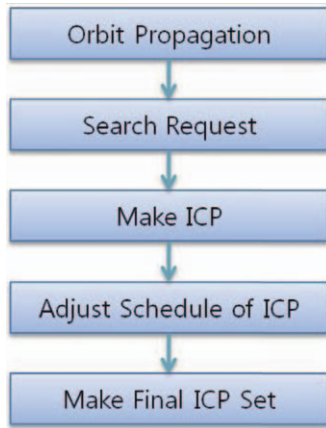


Fig. 1. The Sequence of ICP generation.

The detail procedure is as follows:

Firstly, the operator performs orbit propagation using orbital elements of given satellite. As a result, the position and velocity of satellite are predicted as a function of time for given duration, and they are displayed on the Map.

Secondly, operator searches Requests, then the area of each Request found are displayed on the Map. Operator generates ICP by selecting the orbit and time, at which the satellite can image the area given by specific Request. At this point, operator shall consider the constraints to obey.

Constraints are defined various ways by satellite system by satellite system. A constraint can be defined on a single plan, on the relationship between two plans, or collection of plans. For example, tilt angle constrain, min/max duration for an imaging or Request’s requirements such as target area, cloud score, priority, term of validity, etc. are defined for single plan.

The agility capability applies on the relation between two plans. To take a simple example, the Fig. 2 and Table 1~Table 3 show the imaging constraints violation case due to agility. The blue rectangles illustrate ICPs and the red polygon shape indicates the area of Requests. There are two ICPs



Fig. 2. Imaging Constraints Violation Case.

Table 1. Request information

Request ID	Country	Priority
Req_A	Japan	Low
Req_B	Korea	High

Table 2. ICP information

ICP ID	Imaging Time	Roll Tilt Angle
Img_A	12:11:00 ~12:11:40	20 deg
Img_B	12:14:43 ~12:15:34	- 25deg

Table 3. Agility Time Constraints

Roll Angle Difference	Agility Time
10 deg	1 min
20 deg	2 min
30 deg	3 min
40 deg	4 min
50 deg	5 min

named as “Img_A” and “Img_B” which cover the given area of Requests labelled as “Req_A” and “Req_B”. The interval between ICPs is around 3 minutes while the required agility time in this case is 5 min. The operator needs to delete, re-generate or

adjust one of ICPs to meet agility constraint.

The example of constraints on collection of ICPs is memory and power constraints. If any of those constraints are not followed the collection of ICPs is not valid for satellite programming, and the satellite cannot take any of images requested. Therefore, the operator shall adjust and validate all ICPs again and again until to meet all constraints requirements before finalizing ICPs for given duration. It is very time consuming ineffective procedure.

The purpose of this paper is to propose an automatic ICP generation method. The proposed method eliminates or reduces the iterative procedure by operator. It will significantly enhance the efficiency of work and reduce the working time. Furthermore, the constraints can be defined flexibly using configuration file for the optimization.

3. A Sequence of Automatic Icp Generation

The automatic ICP generation consists of 4 steps derived from the operator's manual ICP generation procedure.

Fig. 3 shows four steps of automatic ICP generation suggested in this paper and output data of each step. The output of previous step is used as an input to current step. KML format is used for output data for easier verification of outputs using Google Earth.

1) Selector

The objective of selector is to calculate orbits, find Requests and pre-generated ICP for given period. These data are re-organized based on orbit number as shown in Fig.4.

Each orbit has their own collection named as "Request List" and "ICP List". The metadata construction based on correlation of each data is

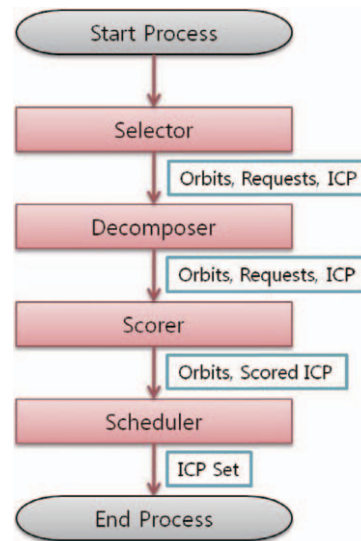


Fig. 3. The Sequence of automatic ICP generation.

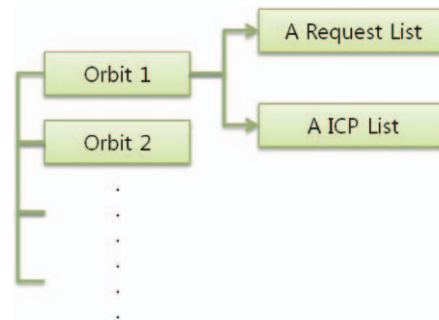


Fig. 4. Structure of metadata.

shown as follow.

Firstly, selector performs orbit propagation by the given period and calculates a possible imaging area of each orbit. The possible imaging area means the area which can be imaged from the given orbit within satellite's tilt capability.

Secondly, it searches Request form database included in the given period. Next, it compares the searched Request's area and possible imaging area of each orbit. If the areas are overlapped, it allocates the Request to "Request List" of the orb it. There is possibility that one Request can be taken from more than one orbit. In this case, the Request is allocated to all of those orbits.

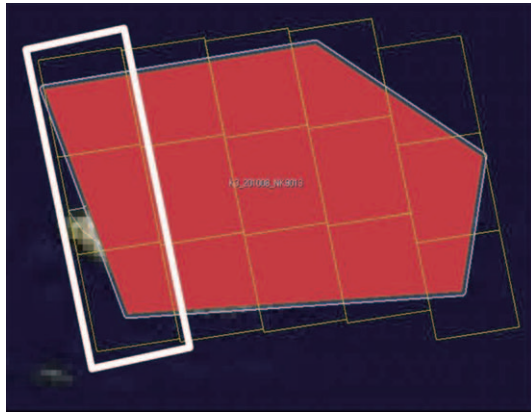


Fig. 5. The Result of Simulated Scene.

Lastly, it searches ICP from database by the given period and allocates the searched ICP to “ICP List” of orbit based on the orbit number of ICP.

2) Decomposer

The purpose of decomposer is to generate candidate ICPs. The Decomposer divides the area given by Request into smaller segments which can be imaged using the actual swath of satellite. This smaller segment is called “simulated scenes”. Then, candidate ICPs are generated by defining relation between orbit and those simulated scenes. The simulated scenes are also used to define required strips and scenes to complete the Request.

A polygon shape in red colour in Fig. 5 shows the Request’s area and the simulated scenes are illustrated as rectangles with yellow line. A rectangle with the white thick line shows strip, which is a set of consecutive scenes in scan direction. These strips are used to generate candidate ICPs in Decomposer. An imaging duration and roll / pitch tilt angle of ICP is calculated based on the orbit and the target location. (C. Mancel, P. Lopez, 2003)

The ICP generated by Decomposer is added to the “ICP List” of the orbit. If the information of ICP does not meet the Request’s demands such as minimum and maximum tilt angle, the ICP is not added.

3) Scorer

The scorer allocates a score value to each ICP. The score value is calculated by pre-defined cost function. The argument of cost function can be different from satellite to satellite. The score function is defined by the argument and its weight in the configuration file.

Request’s priority, remaining imaging expiration date, remaining imaging area, expected cloud score of ICP, possibility of duplicate imaging, etc. are examples for scoring arguments.

4) Scheduler

The scheduler generates the optimized ICP set based on ICP information. The scheduler utilizes imaging period, score, etc and imaging constraints for the optimization. More than one ICP can be generated for the Request from the Decomposer as shown Fig. 6 depending on the area size of area of Request.



Fig. 6. Candidate ICPs.

Table 4. Candidate ICP set information

Orbit Number	Candidate ICP Set(ICP ID Pair)
9200	A - H
9200	B - J
9200	B - K
9200	C - H
9200	C - J
9200	D - K
...	...
...	...

Long rectangle areas named as “A”~“K” indicates the ICPs and the polygon area in red colour which behind ICPs illustrates the area of Request.

As can be seen from Table 4 Candidate ICP set information, there can be lots of candidate ICP set. When there are too many candidate ICPs, it takes very long time to get the optimized ICP set among them by the sequential approach. In order to enhance the performance, the dynamic programming approach is used for the scheduler.

The dynamic programming is a method to simplify a complicated problem by breaking it down into simpler sub-problems. It resolves the sub problems and combines the result recursively to get the solution for the original problem. If the sub-problem is overlapped, it re-uses the result of pre-calculated sub-problem instead of calculating it again. In this way, dynamic programming reduces overall processing time. In order to get the optimized ICP set, lots of sub-set is generated based on the imaging time, score etc. and each sub-set can be an input for the other sub-set so that the Dynamic Programming method is an effective solution in this case. (Josef Roach Bogosian, 2008)

Table 5 shows conditions and goal in order to verify the validity of the Dynamic Programming method for the optimized ICP generation. The complete enumeration survey of possible ICP set is conducted, and the result of the Dynamic Programming is compared with it. A specific program is implemented and used for the Dynamic

Table 5. Conditions for Dynamic Programming & Complete Enumeration

Number of ICP	20
Minimum Gap time(Sec) between ICPs	1 ~ 5 (Random Assignment)
ICP Score (Used to calculate a score of ICP set)	1 ~ 10 (Random Assignment)
Goal	Find the highest score of ICP

Table 6. Result of the Complete Enumeration of Valid ICP Sets

Average Execution Time (sec)	25.62753906
------------------------------	-------------

Table 7. Result of the Dynamic Programming

Average Execution Time (sec)	0.00097652 (26,000 times faster than the complete enumeration result)
The identical ICP Set in the complete enumeration result is generated or not	All generated (10 / 10)
The average rank of ICP Set’s score based on the complete enumeration result	1.921157578 (Top 2%)

Programming and complete enumeration survey.

The results are compared one by one directly based on 2 ~ 5 number of candidate ICP from each result in order to confirm the accuracy of the method.

Table 6 and Table 7 show results of the complete enumeration and the Dynamic Programming.

As shown in the tables, the result of the Dynamic Programming method is much faster than the result of the complete enumeration. Furthermore, the generated ICP Set by the Dynamic Programming is exactly matched with one of ICP Sets generated by the complete enumeration, and its score is ranked in the top 2%. It is not possible or not effective to generate ICP Set such like the complete enumeration style due to the limitation of memory and the operation time in the actual operation environment so that the Dynamic Programming could be worthy solution.

Fig. 7 shows the scored candidate ICPs form the scorer and Fig. 8 shows the optimized ICP set selected by the scheduler based on its score. The red circle illustrates the area of Request and the yellow rectangle represents an ICP. The orbit in this figure is an ascending orbit and is displayed using blue line.

The Fig. 7 and Fig. 8 show that 24 ICPs (two ICPs for each Request) are used as input to Scheduler and the optimized ICP set with two ICPs is generated.

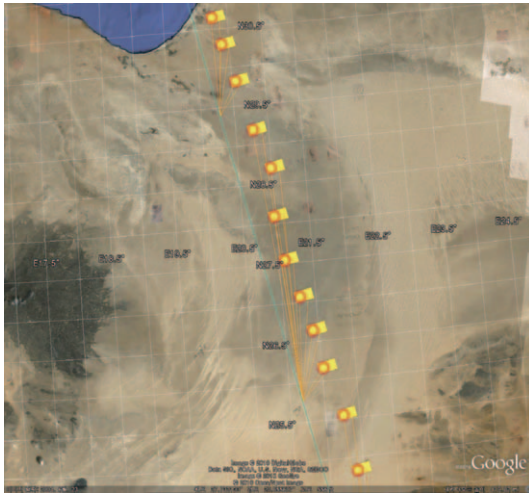


Fig. 7. The Result of Scorer (Input to Scheduler).

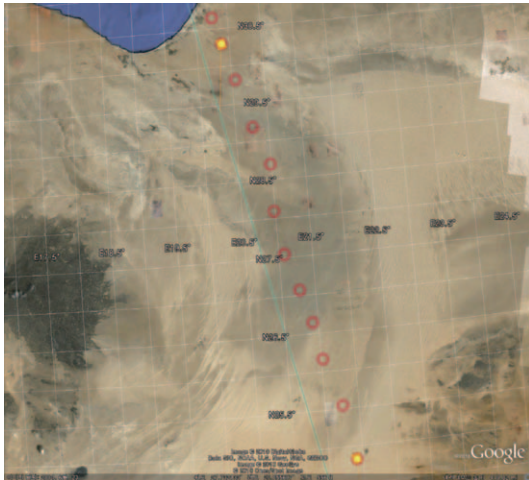


Fig. 8. The Result of Scheduler.

4. Validation of The Automatic Icp Generation

Four of functional test and one of performance test have been done in order to show the validity. A pseudo satellite system with high agility is defined for the test. Parameters for the pseudo satellite are defined referring those of real high performance satellites.

The test computer's specification is as follows:

- OS : Windows XP

- CPU : Core2Duo 2.6G
- RAM : 2GB

The score items and the constraints for the performance test in the Scorer step is as follows.

- Priority of Request
- Agility Time
 - 23 seconds + (0.6 * Difference between two neighbouring ICP's tilt angle)

Very simplified score function and constraint are used for test purpose, but more complicated score function and constraints do not affect the performance of proposed approach.

1) Functional Test # 1 [Agility Constraint]

The purpose of this test is to check whether the proposed approach utilizes the agility of satellite. For this test, 12 Requests with area of interest in circle are used and the same priority is assigned to all Requests.

Table 8 shows input Requests information. All priorities are assigned as 1, and the area information is given by 'Latitude; Longitude; Radius (Km)' format. The "Imaging" column indicates the Requests that are finally selected by automatic ICP generation.

Fig. 9 illustrates the output KML file from each step on the Google Earth map. Each Request can be covered by two strips, so that 24 candidate ICPs are

Table 8. Input Requests of Function Test - 1

ID	Priority	Coordinates	Imaging
1	1	30.64;20.25;7.5;	
2	1	30.29;20.38;7.5;	O
3	1	29.81;20.55;7.5;	
4	1	29.18;20.76;7.5;	
5	1	28.69;20.98;7.5;	O
6	1	28.09;20.98;7.5;	
7	1	27.54;21.08;7.5;	
8	1	27.07;21.26;7.5;	O
9	1	26.64;21.41;7.5;	
10	1	26.16;21.52;7.5;	
11	1	25.54;21.76;7.5;	O
12	1	24.82;21.90;7.5;	

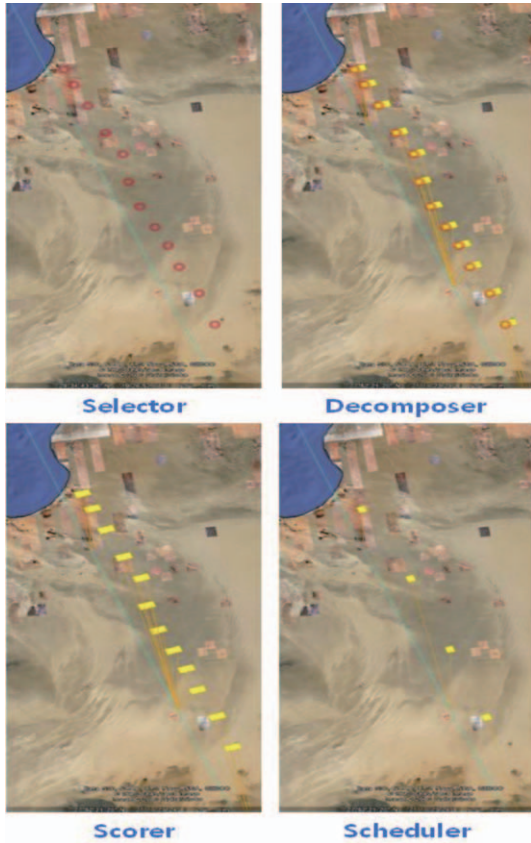


Fig. 9. The Result of Functional Test-1.

generated from Decomposer. Four ICPs out of 24 candidate ICPs are selected as the final ICP set. Top Request ID 1 is shown at the top.

When all candidate ICPs have same priority, it is validated that the proposed method select final result considering the agility capability.

2) Functional Test # 2 [Optimization based on Priority]

The objective of this test is to check the automatic ICP generation considers the priority of each candidate ICPs. The same Requests with test-1 used, but different priorities are applied.

Table 9 shows the Request table used in functional test-2. Priority 1 is the highest and 9 is the lowest. In “imaging time” column shows imaging star time and time gap with previous candidate ICP in parenthesis.

Table 9. Input Requests of Function Test - 2

ID	Priority	Imaging Time (Time Differ)	Imaging (Time Differ)
1	1	47:20 (5)	O (27)
2	9	47:14 (7)	
3	1	47:06 (8)	
4	9	46:55 (7)	
5	1	46:47 (9)	O (30)
6	9	46:37 (8)	
7	1	46:28 (7)	
8	9	46:20 (6)	
9	1	46:13 (7)	O (27)
10	9	46:05 (9)	
11	1	45:55 (11)	
12	9	45:43 (0)	O (0)

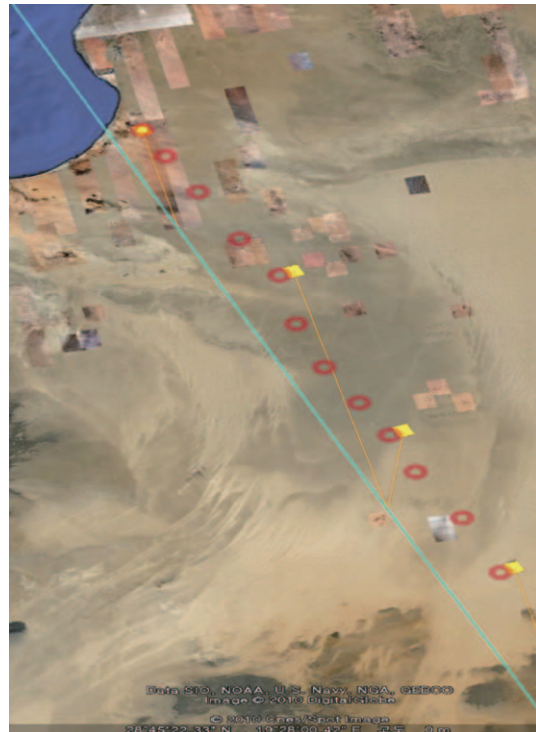


Fig. 10. The Result of Functional Test-2.

For example, ID 12 finished its imaging at 45:44 and then ID 11 follows after 11 seconds. The meaning of “Imaging” is the same as Test-1’s and the number in parenthesis shows the time gap between the selected ICPs.

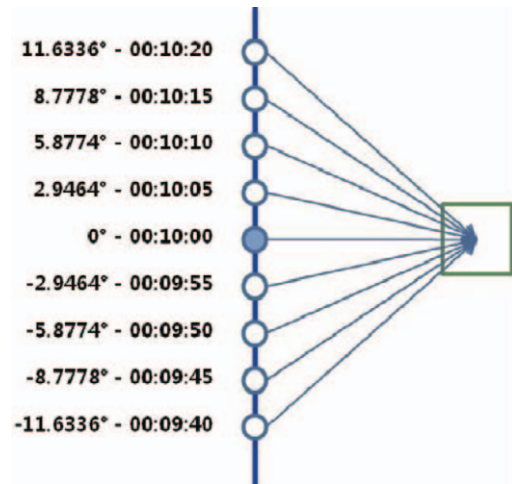


Fig. 11. Example of Pitch Tilt Angle.

Fig. 10 represents the result of functional test-2 shown on Google Earth. 4 ICPs out of 24 candidate ICPs are selected like the functional test 1, but the IDs are different.

The result can be interpreted as follows: The proposed algorithm searches global optimization. In the given candidate ICP set, there is no way to take four Requests with priority 1 when considered agility. Therefore, the algorithm selected a set with three Requests with priority 1 and one Request with priority 9. The proposed method achieved lowest sum of priorities in the set, namely 12.

3) Functional Test # 3 [Pitch Tit Capability]

In the test, same locations used in functional test 1 are used. However, the priorities have been changed and pitch tilt capability is enabled. The pitch tilt capability is common capability of high performance satellites. It makes it possible for the satellite to take image of same area at different time using pitch tilt. When this capability is enabled there can be more candidate ICPs.

In this paper, the pitch tilt capability is considered in time domain. Discrete fixed time gap of 5 seconds was used to represent different pitch angles in this

Table 10. Input Requests of Function Test - 3

ID	Priority	Imaging Time (Time Differ) [Pitch Time]	Imaging (Time Differ)
1	1	47:20 (5) [47:40]	O (25)
2	2	47:14 (7) [47:14]	O (25)
3	3	47:06 (8)	
4	4	46:55 (7)	
5	5	46:47 (9)	
6	6	46:37 (8)	
7	7	46:28 (7) [46:48] [46:18]	O (27) O (30)
8	8	46:20 (6)	
9	9	46:13 (7)	
10	10	46:05 (9)	
11	1	45:55 (11) [45:50]	O (26)
12	9	45:43 (0) [45:23]	O (0)

test. We allowed eight more candidate ICPs with 5 seconds gap as shown Fig. 11, which represents pitch tilt capability of about 11.6 degree. AS a result we had 216 candidate ICPs from 12 Requests.

Table 10 shows Requests information used in functional test 3. The “Pitch Time” is an imaging start time of the selected ICP among the candidate ICPs. Fig. 12 illustrates the test result on the Google Earth. We can see 6 ICPs are selected in the test while 4 ICP are selected in functional test-1 and 2. The increased number of ICPs selected is due to pitch tilt capability.

4) Functional Test-4 [Various Shapes for Request]

Only circle shape Requests are used for the previous test. In this test, various shapes for Requests such as circle, rectangle and polygon are used.

Table 11 shows Requests information table used in functional test-4. Four circle shapes, two rectangle shapes and four polygon shapes are used. The total count of candidate ICPs generated in the Decomposer step is 44. The test result shows that the proposed



Fig. 12. The Result of Functional Test-3.



Fig. 13. The Result of Functional Test-4.

Table 11. Input Requests of Function Test - 4

ID	Priority	Coordinates	Imaging
1	1	Circle	O
2	2	Rectangle	O
3	3	Circle	
4	4	Rectangle	
5	5	Circle	O
6	6	Circle	O
7	7	Polygon	O
8	8	Polygon	
9	9	Polygon	O
10	10	Polygon	

algorithm works very well with Requests of various shapes based on pre-defined condition and constraints. Fig. 13 illustrates the test result on the Google Earth. The number displaying on the each white shape means Request ID. 6 ICPs are selected among 44 candidates ICP.

Table 12. Test condition and data of the performance test

Period of automatic generation of ICP	24 hour
Pitch tilt capability	Not applied
Requests	2003 circle shape Requests which has interval of latitude 2 degree and longitude 16 degree between each neighbouring Requests.
Score Calculation Item	1. Request Priority 2. ICP Forecast weather data 3. ICP Snow 4. Request Tilt Angle 5. Request Locality

5) Performance Test

The purpose of this test is to estimate the time required for the generation of optimized ICP set for 1 day using proposed algorithm.

Test condition and data used are as follows:

The performance test result is as follows:

It takes approximately 13 minutes to generate the optimized ICP set. The result shows that this

Table 13. The performance test result

Total time used	~ 13 minutes
Time used by Selector	12 minutes 35 seconds
Time used by Decomposer	18 seconds
Time used by Scorer	56 seconds
Time used by Scheduler	6.5 seconds

approach can be applied to real operation of Earth observation satellite. Furthermore introducing parallel implementation of each step will decrease more time for the processing.

5. Conclusion

In this paper, the automatic ICP generation algorithm was introduced, which reduces the time required by manual operation for optimization considerably. The result of functional tests and performance test were given to show that the proposed approach is applicable to real operation.

The developed approach has general architecture so that it can be used for any satellite system for the optimization of planning. Proposed approach has been integrated into KOMPSAT-3 planning tool for example.

6. Future Works

The proposed algorithm optimizes the cost function given by Scorer. It also utilizes some constraints to rule out some candidates before optimization. Therefore, to achieve more reliable result, it is very important to design the cost function and usage of constraints. In future work, various cost function and constraints implementation will be tested for better and more reliable result.

References

- Mancel, C. and P. Lopez, 2003. Complex optimization problems in space systems, *International Conference on Automated Planning and Scheduling 2003*.
- Bogosian, J. R., 2008. Image Collection Optimization in the Design and Operation of Lightweight, Low Areal-Density Space Telescopes, *Massachusetts Institute of Technology*.
- M. William, 2002. Satellite image collection optimization, *Optical Engineering*, 41(9): 2083-2087.