DynaQUEST: A New Approach to the Dynamic Multi-satellite Scheduling Problem

J. Berger[1] [a], N. Lo[2], M. Noel[3] and L. Noutegne[2]
[1]*Defence Research Development Canada - Valcartier, 2459 Pie-XI Blvd. North, Québec, PQ, Canada*
[2]*SATWII Solutions, Québec, PQ, Canada*
[3]*Université Laval 1025, Avenue des Sciences-Humaines Québec, PQ, Canada*
jean.berger@drdc-rddc.gc.ca, nassirou.lo@satwii.com, martin.noel@atn.ulaval.ca, noutegnelionel@gmail.com

Keywords: Multi-satellite Scheduling, Dynamic Planning, Mixed-integer Quadratic Programming.

Abstract: Reported dynamic multi-satellite scheduling approaches for Earth observations show many limitations when operating in time-varying uncertain environment. They largely run over predetermined time periods and often offline, assume negligible execution time, improperly account for the passage of time during planning, remain myopic or fail to show "anytime" behavior. A novel approach to solving the dynamic multi-satellite scheduling problem is proposed. The open-loop with feedback DynaQUEST approach includes an event-driven controller monitoring dynamic situation evolution while supervising a co-evolving episodic scheduling problem solver. Reactive to real-time and delayed information feedback, the controller timely enables the problem-solver to stay responsive, interruptible and adaptive, taking advantage of emerging opportunities to timely improve solution quality. The problem-solver continually solves a new static problem shaped by dynamic changes and constrained by current resource commitments to adaptively expand the emergent solution. Problem model formulation is based on network flow optimization using mathematical programming. Departing from mainstream approaches widely promoting an exact objective function coupled with a heuristic problem-solving method, the proposed approach alternatively combines an approximate objective function and an exact algorithm. The approach embraces an extended time horizon relaxing myopic planning. Computational results prove the approach to be cost-effective and to outperform alternate baseline heuristics.

1 INTRODUCTION

Dynamic space-based collection tasking to support time-critical mission is pervasive in many application domains such as military Intelligence, Surveillance and Reconnaissance, environment monitoring and disaster/emergency management. Adaptive and efficient multi-satellite collection scheduling is crucial to maintaining persistent situational awareness in time-varying uncertain environment. A satellite image acquisition schedule may be very sensitive to uncertainty, stochastic task demand, emergency task surge, resource capacity fluctuations, dynamic constraints, plan execution failures, decision cycle latency, computational and/or communication resources available, changing weather conditions or unexpected exogenous events. As a result, a dynamic collection tasking solution approach to be embedded in a decision support system must be fast, responsive, opportunistic and demonstrates graceful adaptation.

[a] https://orcid.org/0000-0001-5885-9254

Recent research contributions on the dynamic multi-satellite collection scheduling problem are numerous. (Pemberton and Greenwald, 2002) first introduced the dynamic problem, considering contingency conditions. In (Verfaillie and Schiex, 1994), the authors exploit a dynamic constraint satisfaction approach to locally search the solution space modifying decision variables. (Kramer and Smith, 2003) propose a repair-based technique to handle the oversubscribed scheduling problem. A rule based heuristic algorithm is used by (Wang et al., 2007) to balance solution performance and solution adjustment. (Dishan et al., 2013) promote an integer programming-based decision model over a receding horizon combined with a flexible approach to schedule dynamic tasks, but overlook anytime problem-solving considerations while planning. In another respect, (Wang et al., 2013), (Wang et al., 2015) propose a multi-objective dynamic scheduling model for emergency tasks coupled to a dynamic task merging algorithm in order to revisit tasks and meet real-time constraints.

In (Zhai et al., 2015), (Niu et al., 2015), the authors combine a hybrid genetic algorithm to build an initial solution and, heuristic variants to adjust and insert dynamically new tasks resorting to task execution duration to handle both performance and robustness. Assuming on-board reactive planning, and monitoring capability to detect imaging precondition violations, (Maillard et al., 2015) use local plan adaptation to accommodate more suitable tasks to meet data imaging contingencies. This is achieved minimizing changes to the plan shared with the ground segment. In counterpart, (Wang et al., 2016) relies on sequential decision-making to solve dynamic scheduling. But they introduce many additional parameters; ad hoc imaging rewards which are misaligned with the global objective pursued; and overlook real-time decision-making contingencies. Concurrently, (Valicka et al., 2016) promote a three-level stochastic programming approach with recourse, but use a state aggregation strategy to significantly reduce or bias problem complexity. Recently, (Chu et al., 2017) propose an anytime branch and bound algorithm for satellite on-board scheduling, but the algorithm is designed for a bi-satellite cluster and remains limited to a small time horizon problem. Alternatively, (Song et al., 2018) introduce fast and refined emergency tasks greedy insertion algorithms to dynamically revisit a schedule, exploiting new task demand priority, task dominance and density to adjust and improve solution quality. In (Wang et al., 2018b), an approach integrating observation and downlink scheduling is proposed. It uses linear programming optimization and a problem decomposition problem-solving technique (column generation), but real-time feasibility and run-time performances for medium or large size problems remain unclear. Finally, (Cui and Zhang, 2019) solve multi-satellite dynamic mission scheduling using priority. The objective is to maximize the observation mission priority and revenues, and minimize the waiting time. A hybrid genetic tabu search algorithm is used to compute the initial satellite scheduling plan, and a dynamic scheduling heuristic is employed to repair the solution and insert new tasks. Most of the recent approaches described above mainly rely on a variety of related exact, heuristic and meta-heuristic problem-solving methods and variants ranging from operations research, to artificial and computational intelligence techniques.

Despite a large body of research, reported real-time dynamic scheduling approaches are often myopic, uninterruptible or are performed offline, occur on arbitrarily predetermined time periods, assume negligible run time or improperly account for the passage of time during planning, coming short to demonstrate anytime behavior. Proposed work contributions generally assume that world state remains unchanged while constructing the solution schedule/plan. This particularly prevails for two-stage myopic approaches relying on initial plan/schedule generation and dynamic repair. As a result, possible computational opportunities and events that may impact solution quality/feasibility or key real-time requirements such as responsiveness, timeliness and graceful adaptation are overlooked.

In this work, a new dynamic multi-satellite scheduling framework is proposed. The open-loop with feedback DynaQUEST approach involves an event-driven controller monitoring dynamic situation development while controlling an episodic multi-satellite scheduling problem solver evolving concurrently. It can be modeled through an event-driven outer thread, the controller, interacting with an inner thread, the episodic multi-satellite scheduling problem solver, running in parallel. Computational results reporting comparative performance clearly show the value of the advocated approach. DynaQUEST proves to be cost-effective and outperforms alternate baseline heuristics. It demonstrates measurable collection and run-time gains.

The remainder of the paper is structured as follows. Section 2 first introduces the centralized dynamic multi-satellite scheduling problem. A description of the proposed open-loop with feedback approach is then given in Section 3. The basic solution design and the detailed dynamic framework integrating a static model are presented. Computational results reporting comparative performance with alternate baseline methods are presented and discussed in Section 4. A summary of the findings is finally given in Section 5.

2 MULTI-SATELLITE SCHEDULING PROBLEM

Given a set of information requests (areas of interest to be observed) properly translated into weighted tasks, the basic multi-satellite scheduling problem supporting information collection consists of allocating collection assets (satellites) to observation tasks (imaging opportunities) to maximize overall expected collection value, subject to a variety of constraints over a predetermined time horizon. Constraints may relate to mission, task and related dependencies, operational, collector, supporting resource, communication, on-board resource capacity, temporal, itinerary (e.g. duty cycle) and cost considerations respectively. Image acquisition is characterized by a probability of successful observation to reflect outcome uncertainty. Image downlinks to and command uplinks from the ground segment are finally ensured through communication assets (e.g. ground stations). The current dynamic problem setting assumes centralized image acquisition and downlink scheduling, achieved by a ground (mission control center, ground station antennas) segment, and distributed plan execution realized by the space (spacecraft components) segment. Satellite on-board processing capability and scheduling adaptations are assumed to be very limited. Dynamic scheduling problem characteristics include: Near real-time task demand, an uncertain environment with imperfect sensors, stochastic resource capacity level and exogenous events (e.g. observation outcomes, new task), bounded rationality (computational resources) and reactivity of the ground scheduler. These stringent problem features impose the ground segment to be fast, adaptive, responsive, opportunistic (timely actions) to support "anytime" scheduling and re-tasking.

3 DynaQUEST APPROACH

3.1 General Description

A centralized open-loop with feedback scheduling decision model over a rolling time horizon is proposed in order to reduce computational complexity and maintain the ground segment responsive. Open-loop planning assumes a single episode (static model) disregarding explicit any intermediate information feedback. Feedback information such as observation outcomes resulting from plan execution is alternatively handled through repeated episodic re-optimization driven by communication opportunities as shown in Figure 1.

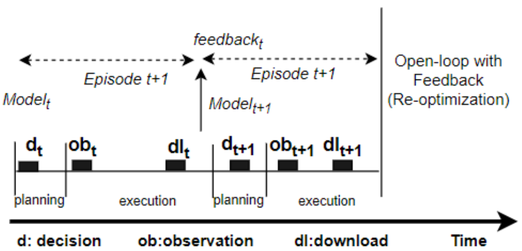

Figure 1: Open-loop with feedback multi-satellite scheduling optimization.

Episodic re-optimization consists to dynamically compute a solution by solving a new static scheduling problem model given information feedback such as new tasks, or recent observation outcomes (image downlink) from the latest episode. Event-driven re-optimization is achieved over a receding time horizon subject to committed action plans or ongoing resource commitments, revisited target/task value, updated constraints and current world state.

3.2 Solution Design

The open-loop with feedback approach called DynaQUEST, involves an event-driven controller monitoring situation evolution and interacting with a time-varying uncertain environment, supervising an episodic multi-satellite scheduling problem solver running continually as shown in Figure 2. It manages incoming events taking into account the passage of time capturing world state evolution while planning.

3.2.1 Controller

The outer thread controller timely reacts to, and processes incoming information feedback (e.g., observation outcomes, new tasks) guiding inner thread problem-solving and making it "anytime"/interruptible. It properly accounts for the passage of time during solution planning, as situation unfolds, deciding on timely algorithm activation and/or interruption over and during communication opportunities (contact periods) with single or multiple satellites. As a wrapper, it is designed to handle world state transition while constructing the solution. The outer thread controller enables the problem-solver to stay responsive, timely and adaptive, benefiting from emerging opportunities (e.g. incoming new tasks, remaining communication time) to further enhance solution quality and timely execution.

3.2.2 Problem-solver

The problem-solver continually solves new static problems dictated by dynamic changes and constrained by current resource commitments to adaptively expand or improve the emergent solution at hand. Problem model formulation is based on network flow optimization using mathematical programming. The proposed problem modeling/-solving approach is called QUEST in reference to its QUadratically constrained program Solver Technology solution implementation, exploiting best state-of-the-art commercial optimization problem-solving machinery (Berger et al., 2020). The QUEST model embraces a larger time horizon relaxing myopic planning, while providing fast problem-solving. QUEST is a new static problem solver using a mathematical quadratic programming approach exploiting problem structure reflected in collection graph and prior domain knowledge (bounding visits to a task) to compute an efficient solution. Based upon the non-linear objective function proposed by (Wang et al., 2018a), QUEST relies on an approximate objective function combined with the utilization of commercially available exact problem-solving techniques. The QUEST problem model concurrently captures coverage approximation and collection uncertainty captured through imaging probability of success.

3.3 DynaQUEST Cycle

New upcoming events such as communication downlinks (environment state, observation outcomes, onboard satellite resource state: energy level, memory usage, orbital cumulative observation time), new tasks and command uplinks mediated by the controller. During respective intermittent contact periods and based on best current computed solution so far, the controller first handles new information. It then updates collection value alluding to the latest serviced targets, interrupts the QUEST solver and revisits the problem model to re-initiate open-loop optimization by directing the problem solver toward a new decision cycle. Kept informed of the latest best computed solution update, the controller then timely uplinks relevant revisited satellite collection plans before triggering a new re-optimization cycle again.

The following description of a simple DynaQUEST cycle and its associated timeline is illustrated in Figure 3 over a single contact period. Here,

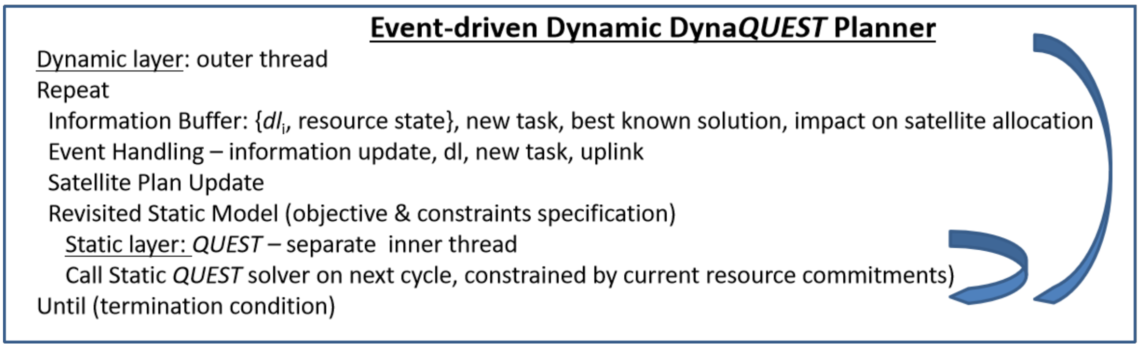

Figure 2: DynaQUEST framework.

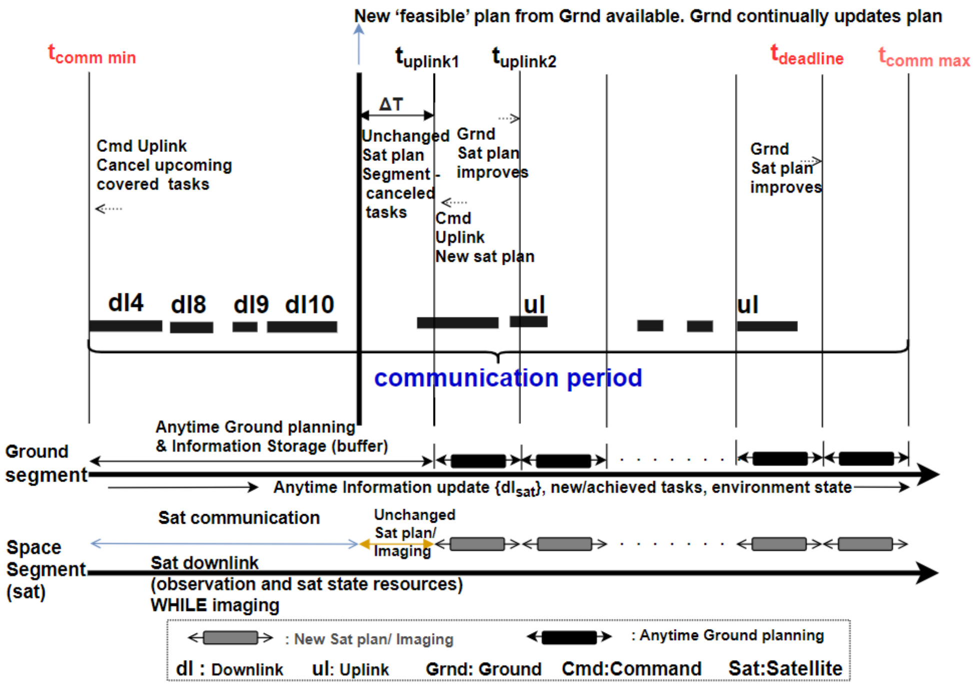

Figure 3: DynaQUEST cycle over a contact/communication period.

we assume that a satellite can image and downlink concurrently.

On a new satellite contact beginning at time t_{comm_min}, the ground segment, based on latest best solution, sends a command uplink, informing the satellite of its new updated local plan from now on, while stored images and resource state update are downlinked. Based on latest downlinked information and assuming sufficient communication time, the ground segment quickly refines during a brief period δT, amounting to a few seconds, its best current solution to possibly re-synchronize with the satellite if the local plan has changed. Plan revision is repeated as required based on new incoming information emerging from any sources over the entire communication time window [t_{comm_min}, t_{comm_max}] defining the contact period. Such information may include latest observation outcomes from other satellites, new tasks, revised constraints, time-varying environmental/weather conditions or any unexpected exogenous events impacting other satellite plan commitments. Reiterated communication may not even take place during that time interval if short-term satellite plan remains unchanged until the next communication contact. Mediated by the controller, an updated local plan is finally uplinked to the satellite if required, by a predetermined deadline $t_{deadline}$ before the end of the contact period t_{comm_max}. Satellite-ground synchronization is ensured by the controller. Should communication time be insufficient to complete the downlinks, the satellite would proceed according to its latest new plan subject to local resource constraint whereas the ground segment would pursue its planning activity with partial information and its current distorted beliefs. Satellite downlink completion and plan adjustment would then be delayed to the next communication cycle. Communication is assumed to occur if sufficient time is minimally available to uplink a new precomputed local plan from the ground. Respective activity timeline for ground and satellite segments is displayed at the bottom of Figure 3.

3.3.1 Ground Segment Re-planning

A new re-optimization cycle is triggered by the event-driven controller as described above in Section 3.3 reactivating the problem-solver. The re-planning problem is conditional to the computed solution partially implemented in the previous episode and defined over the remaining mission time horizon. The new problem model is provided with new/pending tasks, task value adjustments based on past collections, committed imaging actions and current satellite intents, refined environmental conditions, and, revised resource capacity constraints respectively. The latter include bounding on-board satellite memory storage until the next communication period. Problem-solving using the open-loop QUEST solver is then initiated.

3.3.2 Space Segment

At the satellite platform level, image acquisition is assumed to take place over a given time interval while image/resource state status (energy budget, available memory storage, duty cycle, failure) downlink are predictably scheduled at the beginning of a new contact period to benefit the problem solver (planner) from fresh information update. Assuming limited on-board processing and planning capability, a satellite is nonetheless presumed able to validate task serviceability preconditions before imaging. In that case, unfeasible tasks due to temporal delays, poor weather, on-board resource shortages or exogenous events are simply ignored and planned image acquisitions simply dropped from the local plan. Relaxing such assessment ability would simply lead to spoiled images under adverse circumstances.

4 COMPUTATIONAL RESULTS

A computational experiment has been conducted to assess the value of the proposed approach. Comparative performance results with baseline problem-solving heuristics are reported and discussed.

4.1 Experimental Setting and Scenarios

Performance comparison is based on expected collection value (CV) successfully obtained over the entire mission time horizon. The measure of performance combines cumulative gathered collection value as well as expected collection value associated with acquisition tasks planned to be ongoing or incomplete by the end of the time horizon. A negative observation outcome conveys no collection contribution for the related task. The average CV random variable can be estimated through Monte Carlo simulation over a set of dynamic scenarios characterized by a degree of dynamism, stochastic task demand and observation outcomes. The experiment has been conducted for a 3-satellite constellation daily mission involving a total of 45 orbits (3 satellites x 15 orbits/satellite), subject to 35 re-planning episodes/ground communication time windows with a maximum of 3 contact periods per satellite orbit, responding to a dynamic task demand. Dynamic demand is captured through two stochastic scenarios/datasets presenting a medium (60%) and a high (93%) dynamism level, respectively. The dynamism level reflects the proportion of initially unknown tasks on average composing total demand. Correspondingly, the scenarios include on average 420 (1000) tasks generated dynamically against 250 (75) tasks assumed to be known initially. Tasks are randomly created using a uniform distribution over [409, 429] and [800, 1200] for respective datasets. Simulation parameters such as space-time episodic task demand, imaging time windows, task value ranging over [0.1, 0.9], probability of successful observation running into [0.25, 1] and, communication time window duration spanning across [0, 4] minutes are all randomly selected using a uniform distribution. Other satellite resource capacity parameters introduced in the QUEST decision model are indicated in (Berger et al., 2020). A sample of 30 simulation runs has been considered to estimate average collection value for each dataset.

4.2 Problem-solving Algorithms

Comparative performance investigation has been performed by substituting the QUEST problem-solver of the two-thread DynaQUEST framework by respective baseline heuristic methods for the two dynamic scenarios. These include MY_PICC (Berger et al., 2018), a simple myopic technique, and GATHER (Berger et al., 2018), a competitive best state-of-the-art heuristic. All scheduling algorithms were implemented in Java on an Intel - Core i7-7700 3.6GHz Quad-Core Processor 64 GB RAM memory.

4.2.1 MY_PICC

The MY_PICC Planning-based Image acquisition heuristic (MY_PICC) (Berger et al., 2018) is a myopic heuristic naively mimicking a human-like strategy. The myopic heuristic consists in moving along orbital collection graphs in parallel, and to constructively schedule the next image acquisition, awarding the collection presenting the highest payoff rate (per time units) within the vicinity of the latest observation of a current path solution. The payoff rate refers to the ratio of expected collection gain over combined delays imposed by satellite travel, feasible transition and imaging durations respectively. Earliest start time occurrence breaks ties.

4.2.2 GATHER

The Genetic Algorithm-based collecTion scHedulER (GATHER) (Berger et al., 2018) evolves a mixed population of feasible/unfeasible solution individuals based on natural selection to maximize expected collection value. The advocated hybrid genetic algorithm explicitly takes advantage of collection graphs reflecting problem structure in representing feasible imaging opportunity transitions, to better manage temporal constraint handling during crossover and mutation operations. A low-cost task scheduling heuristic embedded in genetic operators, further provides directed search and speed-up, generating feasible high-quality solutions.

4.2.3 QUEST

QUEST implementation is mainly relying on Lagrangian relaxation (decision variable integrality relaxation) and a variety of efficient built-in problem-solving search methods to find an integer solution.

4.3 Results

Computational results are reported in Figures 4-5 for respective scenarios. DynaQUEST is compared with two dynamic baseline heuristics named DynaMY_PICC and DynaGATHER, in terms of performance gap with respect to expected collection value, and computational run-time performances. The smaller the expected collection value (CV) gap, the better the solution quality.

Performance gap is computed as $(CV^{DQ} - CV)$ / CV^{DQ} where CV^{DQ} is the estimated expected collection value for DynaQUEST.

Average collection value estimators for respective methods are also summarized in Table 1 over both medium (60%) and high (93%) dynamism level scenarios, assuming a 30 simulation sample and a 95% confidence interval. Average cumulative planning run-time and related standard deviation are also reported.

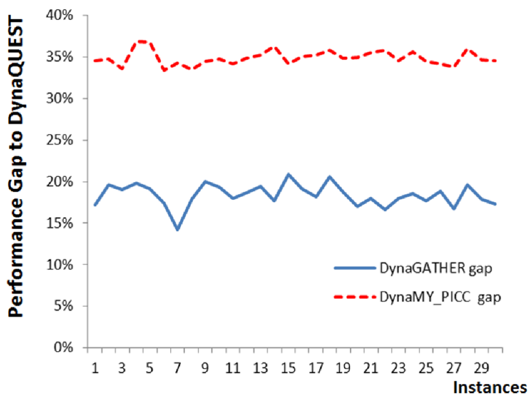

Figure 4: Medium dynamism level scenario relative performance gap to DynaQUEST against problem instances.

Overall computational results show DynaQUEST to outperform other baseline methods, indicating a

Table 1: Average performance.

	Medium dynamism level - 60%		High dynamism level - 93%	
	Average collection value estimator	Average cumulative planning run-time (s) over mission time horizon	Average collection value estimator	Average cumulative planning run-time (s) over mission time horizon
DynaQUEST	286.17± 1.84	59.4±20.67	313.47±8.23	12.40±1.7
DynaGATHER	233.56±1.22	51.54±1.42	248.96±8.14	60.81±6.49
DynaMY_PICC	186.29±1.41	5.77±0.17	186.97±7.35	5.16±0.70

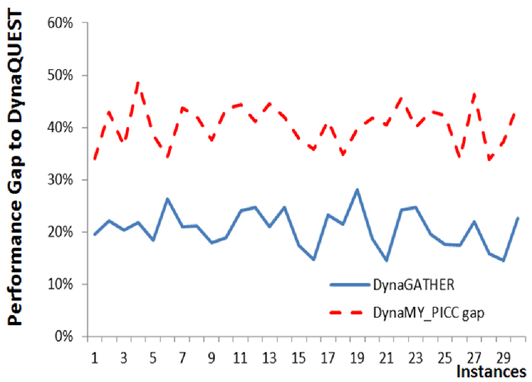

Figure 5: High dynamism level scenario relative performance gap to DynaOUEST against problem instances.

relative performance gap of nearly 20% and 35% in comparison to DynaGATHER and DynaMY_PICC, respectively. Differential estimated average collection value for the medium and high dynamism level scenarios convincingly proves the same outstanding domination as well as shown in Table 1.These results also implicitly suggest relative DynaQUEST superiority for even smaller dynamism level or task demand surge, as the approach further promotes longer term planning. In another respect, one can note from Figures 4-5 a small positive correlation between dynamism and relative CV performance gap. Performance gap is slightly larger for higher dynamism. This is due to a performance degradation expected for myopic methods when uncertainty on task demand increases.

Cumulative planning run-time performance reported in Table 1 expectedly indicates QUEST to be 5 times faster than DynaGATHER and only twice slower than the myopic DynaMY_PICC when the initial number of tasks is small. Conversely, relative planning run-time slightly degrades when initial task demand is large. Open-loop optimization using exact method (QUEST) and conditional to previous action commitments is very fast, given episodic small task demand. It appears that episodic re-planning run-time for DynaQUEST is on the order of few seconds (2s), typically an order of magnitude under the minute timescale, making it very fast by any near real-time Earth observations standards and virtually "anytime" in steady-state condition. Additional run-time gains could be further expected should a faster computer be used.

5 CONCLUSION

A novel open-loop with feedback approach to solving the centralized dynamic multi-satellite scheduling problem has been proposed. The two-thread DynaQUEST framework couples a responsive event-driven controller monitoring situation evolution to the new efficient non-myopic open-loop problem-solver QUEST responsible for episodic re-planning. In response to incoming events, the problem-solver adapts and expands the solution schedule by solving a new constrained static problem instance subject to ongoing resource commitments. Combining an approximate objective function and an exact algorithm, QUEST departs from traditional myopic planning, while providing fast problem-solving. These features confer to the approach suitable real-time properties such as timeliness, responsiveness, speed and graceful adaptation and anytime behavior. A computational experiment has demonstrated DynaQUEST to outperform alternate baseline heuristics.

Future work aims at exploring non-myopic onboard plan adaptations assuming sufficient processing capability. The challenge consists in efficiently combining an evolving open-loop with feedback ground segment solution and, a closed-loop re-planning space segment more responsive to local event contingencies to better handle uncertainty. This would pave the way toward progressively examining multi-satellite distributed scheduling. Another direction is to look at potential implications of full network connectivity that can be provided by communication satellite networks, on dynamic planning solution quality.

REFERENCES

Berger, J., Giasson, E., Florea, M., Harb, M., Teske, A., Petriu, E., Abielmona, R., Falcon, R., and Lo, N. (2018). A graph-based genetic algorithm to solve the virtual constellation multi-satellite collection scheduling problem. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–10. IEEE.

Berger, J., Lo, N., and Barkaoui, M. (2020). Quest–a new quadratic decision model for the multi-satellite scheduling problem. *Computers & Operations Research*, 115:104822.

Chu, X., Chen, Y., and Tan, Y. (2017). An anytime branch and bound algorithm for agile earth observation satellite onboard scheduling. *Advances in Space Research*, 60(9):2077–2090.

Cui, J. and Zhang, X. (2019). Application of a multi-satellite dynamic mission scheduling model based on mission priority in emergency response. *Sensors*, 19(6):1430.

Dishan, Q., Chuan, H., Jin, L., and Manhao, M. (2013). A dynamic scheduling method of earth-observing satellites by employing rolling horizon strategy. *The Scientific World Journal*, 2013.

Kramer, L. and Smith, S. F. (2003). Maximizing flexibility: a retraction heuristic for oversubscribed scheduling problems. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI '03), 1218–1223*.

Maillard, A., Verfaillie, G., Pralet, C., Jaubert, J., Sebbag, I., Fontanari, F., and Lhermitte, J. (2015). Adaptable data download schedules for agile earth-observing satellites. *Journal of Aerospace Information Systems*, 13(8):280–300.

Niu, X., Tang, H., Wu, L., Deng, R., and Zhai, X. (2015). Imaging-duration embedded dynamic scheduling of earth observation satellites for emergent events. *Mathematical Problems in Engineering*, 2015.

Pemberton, J. and Greenwald, L. G. (2002). On the need for dynamic scheduling of imaging satellites. In *Remote Sensing and Spatial Information Sciences, 34, 165–171*. International Archives of Photogrammetry.

Song, Y., Huang, D., Zhou, Z., and Chen, Y. (2018). An emergency task autonomous planning method of agile imaging satellite. *EURASIP Journal on Image and Video Processing*, 2018(1):29.

Valicka, C. G., Rowe, S., Zou, S., Mitchell, S. A., Irelan, W. R., Pollard, E. L., Garcia, D., Hackebeil, G. A., Staid, A., Rintoul, M. D., et al. (2016). Dynamic multi-sensor multi-mission optimal planning tool. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).

Verfaillie, G. and Schiex, T. (1994). Solution reuses in dynamic constraint satisfaction problems. In *Proceedings of the 12th National Conference on Artificial Intelligence, 307–312*. Seattle, Wash, USA.

Wang, C., Tang, J., Cheng, X., Liu, Y., and Wang, C. (2016). Distributed cooperative task planning algorithm for multiple satellites in delayed communication environment. *Journal of Systems Engineering and Electronics*, 27(3):619–633.

Wang, J., Demeulemeester, E., Hu, X., Qiu, D., and Liu, J. (2018a). Exact and heuristic scheduling algorithms for multiple earth observation satellites under uncertainties of clouds. *IEEE Systems Journal*.

Wang, J., Zhu, X., Qiu, D., and Yang, L. T. (2013). Dynamic scheduling for emergency tasks on distributed imaging satellites with task merging. *IEEE Transactions on Parallel and Distributed Systems*, 25(9):2275–2285.

Wang, J., Zhu, X., Yang, L. T., Zhu, J., and Ma, M. (2015). Towards dynamic real-time scheduling for multiple earth observation satellites. *Journal of Computer and System Sciences*, 81(1):110–124.

Wang, J.-m., Li, J.-f., and Tan, Y.-j. (2007). Study on heuristic algorithm for dynamic scheduling problem of earth observing satellites. In *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)*, volume 1, pages 9–14. IEEE.

Wang, Y., Sheng, M., Zhuang, W., Zhang, S., Zhang, N., Liu, R., and Li, J. (2018b). Multi-resource coordinate scheduling for earth observation in space information networks. *IEEE Journal on Selected Areas in Communications*, 36(2):268–279.

Zhai, X., Niu, X., Tang, H., Wu, L., and Shen, Y. (2015). Robust satellite scheduling approach for dynamic emergency tasks. *Mathematical Problems in Engineering*, 2015.