Managing Uncertainties via Preconditioning Problem Complexity

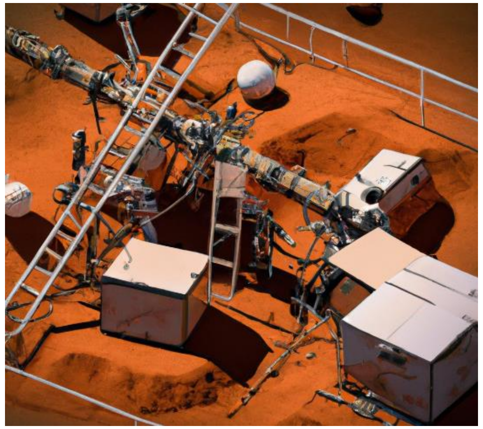
Natalia M. Alexandrov

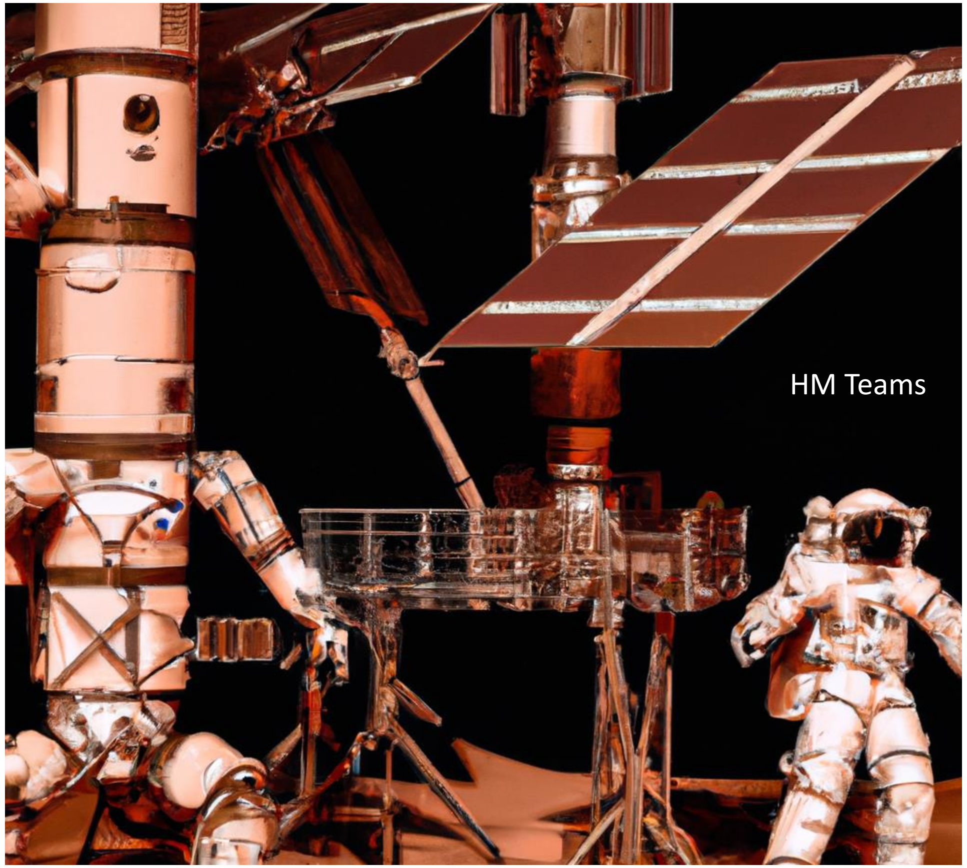
NASA Langley Research Center

IEEE/SMC-IT, 19 July 2023

Image credits throughout the talk: NASA, NMA, NMA+AI

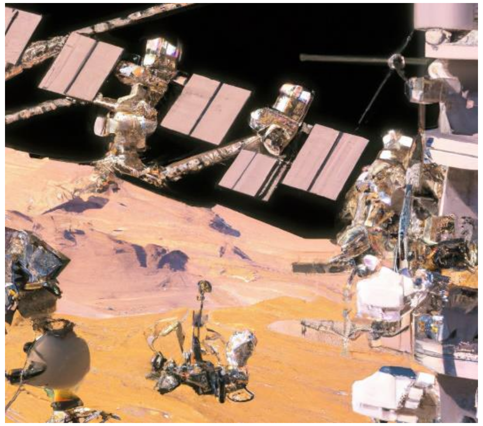
Problem Setting





- Autonomous machine systems and human-machine (HM) teams

- Critical aspect of uncertainty in operation of complex systems: the ability of agents to arrive at satisfactory decisions and the attendant actions in time, as a function of problem complexity



Complexity ⟶ t

- ∀ action preceded by solution of a decision-making problem
- Reducing solvability uncertainty always done via bounding problem complexity; e.g., current airspace
- This work:
  - Dynamic complexity bound for multiagent systems, both machine and HM
  - Additional challenge for machine component: must have explicit problem formulation
  - (Beneficial) side effect: an approach to informing decision allocation
- Desiderata:
  - Represent complexity in computable, actionable form
  - Detect approaches to unacceptable complexity
  - Reconfigure system to forestall transition to unacceptable complexity & back when resolved

Proposed Measure of Complexity

Here rely on formulating decision-making as optimization; other formulations possible

Complexity

- Tractability, with a look ahead, of the problem solution on a time budget, as a function of external and internal parameters

+

- The quality of solutions of the decision-making problem measured in terms of constraint satisfaction, optimality, and robustness

+

- Other context-dependent criteria

Conceptual Reasoning Framework

Given:

- *An agent and a goal*

Initialize:

- *Set time $t=0$*
- *Set initial sampling time interval Δt*
- *Set initial look-ahead time T*
- *Set initial look-ahead sampling time interval $\Delta\tau$*
- *Select initial problem-solving algorithm P*
- *Select environmental complexity parameters C*
- *Select initial environmental complexity model M*
- *Select transition criteria*
- *Select stopping criteria*

Conceptual Framework, cont.

Do until (stopping criteria are satisfied)

Acquire and assess complexity parameters C at time t

Set $\tau = t$

Do while ($\tau \leq t + T$)

Estimate look-ahead complexity parameters at time τ

$\tau = \tau + \Delta\tau$

End do

Input complexity parameters into complexity model M; assess approach to phase transition

If (approach to transition detected) then

Reconfigure operations ("add structure")

Else

Assess system's performance slack

If (slack)

Reconfigure operations ("relax structure")

Else

Continue present operations

End if

End if

Update time

End do

Example: *ab initio* air traffic control

Problem:

- Compute trajectory of an aircraft from point A to point B
- No assumption on structure

Complexity parameters:

- External:
  - Density and heterogeneity of the relevant airspace volume
  - Interaction activation
- Internal, e.g.: physical properties of the aircraft and computational properties of decision-making algorithms

Tractability:

- Solving trajectory optimization problem on a time budget

Metrics for quality of solutions of the decision-making problem:

- Constraint satisfaction
- Optimality
- Robustness: robust solutions live in the regions of space that allow for many alternative solutions (e.g., flexibility preservation, Idris et al.).

MAGE (Monitor, Anticipate, Guide, Evolve) for Air Traffic

- <u>Complexity model construction</u>:

- <u>Reconfiguration</u>: change in the decision problem objectives, constraints, and variables
  - Directed modification
  - Autonomous, distributed modification with emergent outcomes
  - Hybrid directed-autonomous

  - Example: reduce weight of the delay objective in high-risk, dense environment

- <u>Decision-making strategies</u>:
  - Solution strategy (e.g., optimization) affects the outcome of actionable complexity prediction
  - As new capabilities arise, complexity models must be re-calibrated

Preliminary Computational Results

- Operation setup in ATMLG (Air Traffic Monotonic Lagrangian Grid)
  - **Initialize system**
    - N = number of aircraft
    - System aircraft interaction distance (10 mi)
    - Set time t_i = start time_i (can be 0 or chosen randomly from a set) for i=1,…,N
  - **Do ∀ 10 sec until End**
    1. ID all aircraft in conflict (within interaction distance)
    2. For ∀ aircraft in conflict, select: {x_1 = Δ heading±45°, x_2= Δ speed±10 kt, x_3= Δ altitude±1000 ft}
    3. For ∀ pair of aircraft in conflict, formulate a constraint:
       1. Compute t of closest approach for 2 aircraft based on current position, heading and speed (regardless of altitude)
       2. Compute separation distance between the 2 aircraft at t of closest approach
       3. Compute a projected altitude factor

          $f_1 = 1 - \frac{(alt_1 - alt_2)}{Alt_{sep}}$ $\quad f_2 = 1 - \frac{(alt_2 - alt_1)}{Alt_{sep}}$ $\quad g = \left(1 - \frac{distance_{closest}}{distance_{allowed}}\right) * f_1 * f_2$
    4. Solve minimize $\sum\left(\frac{x_i}{scale_i}\right)^2$, subject to $g \leq 0$
  - **End Do**

Sample of Outcomes

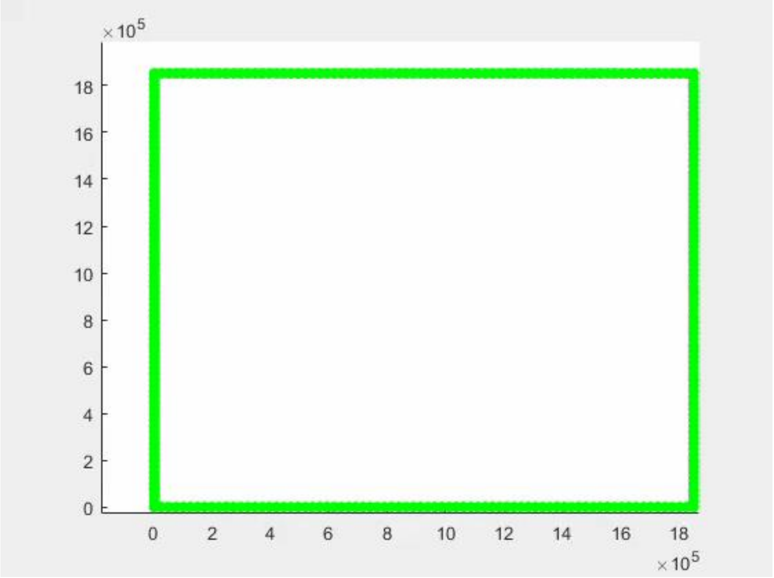
Training set

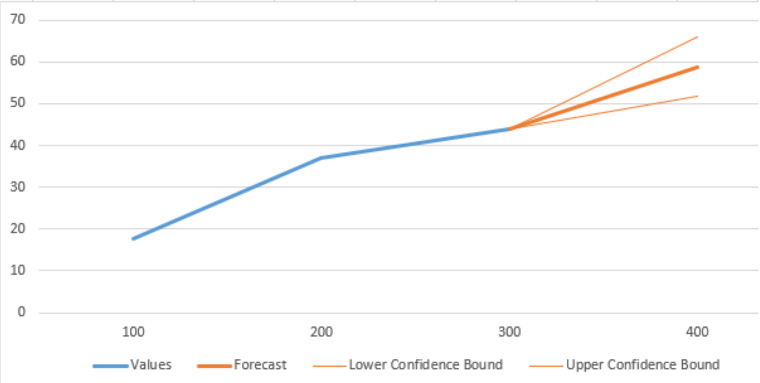
Validation set

Different optimizer

1 All planes flying at a constant altitude of 30,000 feet.
2 All planes flying at a constant airspeed of 400 knots.
3 Check distance is 10 nautical miles and intrusion distance is 5 nautical miles.
4 Circle radius is 1000 nautical miles.
5 All planes start at time=0.

	Optimizer name	number of planes	initial separation	trial number	number of conflicts	number of course changes	closest approach	execution time	optimizer total time	optimizer worst case time
9	KSOPT	100	62.83	1	118	89	5.00	19.187500	0.026000	0.008000
10		100	62.83	2	118	103	5.00	16.609375	0.055000	0.009000
11		100	62.83	3	98	70	4.50	17.171875	0.012000	0.008000
12		100	62.83	4	134	125	5.00	17.546875	0.038000	0.001000
13		100	62.83	5	126	111	5.00	17.656250	0.016000	0.001000
14		200	31.42	1	500	495	2.50	29.781250	0.261000	0.010000
15		200	31.42	2	482	545	1.70	29.031250	0.193000	0.020000
16		200	31.42	3	410	400	3.60	27.500000	0.232000	0.021000
17		200	31.42	4	508	423	4.60	28.281250	0.215000	0.021000
18		200	31.42	5	446	415	4.20	27.406250	0.259000	0.010000
19		300	20.94	1	1066	1034	1.90	42.968750	2.955000	0.402000
20		300	20.94	2	1064	1083	1.50	44.015625	3.389000	0.419000
21		300	20.94	3	1148	1049	2.60	11.859375	2.359000	0.203000
22		300	20.94	4	1052	1004	2.50	12.968750	3.735000	0.923000
23		300	20.94	5	1072	1058	0.90	13.890625	3.677000	0.616000
24		400	15.71	1	1906	1865	1.10	33.375000	22.192000	2.733000
25		400	15.71	2	1972	1892	0.70	47.406250	34.176000	7.738000
26		400	15.71	3	1910	1781	0.90	30.625000	19.172000	3.087000
27		400	15.71	4	2008	2026	1.30	39.984375	27.847000	4.613000
28		400	15.71	5	1984	1944	1.70	38.609375	26.579000	3.765000
29	NLOPT_MMA	100	62.83	1	84	69	5.00	36.546875	31.311000	12.550000
30		100	62.83	2	100	65	4.30	65.031250	59.835000	12.488000
31		100	62.83	3	82	44	3.60	77.453125	70.474000	11.719000
32		100	62.83	4	118	67	3.10	17.906250	11.362000	3.866000
33		100	62.83	5	114	125	4.60	81.093750	75.586000	10.797000





Timeline	Values	Forecast	Lower Confidence Bound	Upper Confidence Bound
100	17.63438			
200	36.99375			
300	44	44	44.00	44.00
400		58.77883	51.65	65.91
		actual 38.		

on

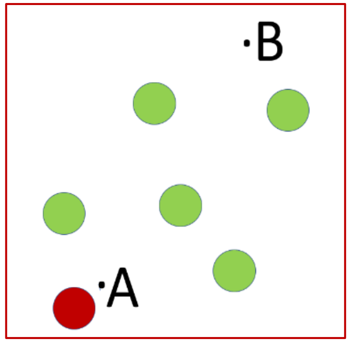
More Numerical Tests of Tractability Prediction

- Shortest Path Finding via Visibility Graph

# obstacles	# scenarios	Avg. # solutions	Avg. time to solution (sec)	# test scenarios	Avg. # test solutions	Avg. % time prediction Δ
4	100	97	0.86	25	24	0.02
5	100	89	1.31	25	19	0.07
10	100	34	324.33	25	7	2.03
20	100	2	4800.00	25	Not found	N/A

- Minimization of Deviation from Optimal Path

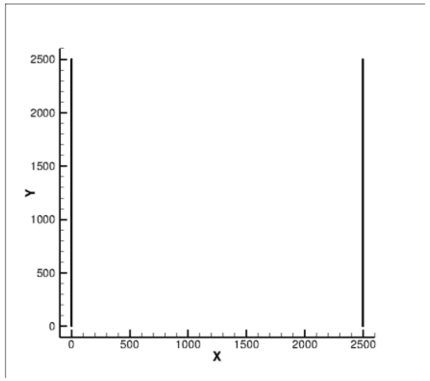
# obstacles	# scenarios	Avg. # solutions	Avg. time to solution (sec)	# test scenarios	Avg. # test solutions	Avg. % time prediction Δ
4	100	94	0.38	25	25	0.01
5	100	100	0.37	25	23	0.01
10	100	83	11.60	25	24	0.03
20	100	86	64.52	25	19	0.05



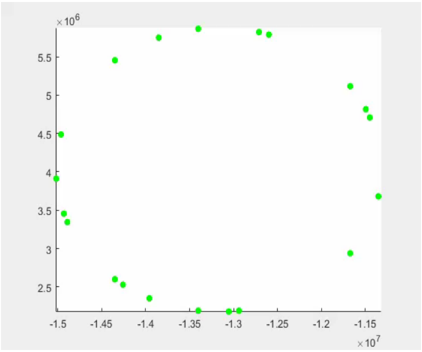
- Detect trends in computational tractability of simple decision problem formulations, using examples of two formulations and two decision-making schemes.
- Results to date indicate that further development of the planned complexity models is justified.

In Progress

- Problem
  - Suppose perfect information is available to all
  - Suppose all participants use the same optimization algorithm
  - Satisfactory solutions not guaranteed
- Current computational setup: Locally centralized
- To do:
  - Find the minimum necessary (and sufficient?) commonality in heterogeneous decision-making to ensure viability of any architecture



Information and decision-making assumptions



Evaluating Approach

Metrics

- Accurate detection of shifts in tractability and quality of solutions from nominal?
- Clear ID of dependence of precursors (slowing down of solutions, shrinking of feasible regions) as a function of variables?
- Low percentage of false positives?
- Note: Comparative metrics are not available because there is no baseline SOA.

Risks

- In shared-rules approaches, low risk
- In algorithmic decision-making, the risk of not detecting precursors in time is higher.
- Mitigation: Lack of detectable precursors will point to the need for different control schemes.

Rewards

- Insights into practical sensitivities of a dense, heterogeneous system to control strategies and uncertainties.
- A component in a toolbox for monitoring system controllability, efficiency, and safety, regardless of the overall control strategy.

Concluding Remarks

- The idea: reduce uncertainty in reaching successful decisions via “conditioning” the system to be simple enough to allow each agent to solve its problem in time.
- Cautiously optimistic about proof of concept:
  - For a centralized or distributed solution, the onset of complexity is detectable in advance
  - Reconfigurability can be controlled automatically
  - For practical solution, need a minimum necessary set of information requirements for any viable architecture
  - In the case or air traffic, few problem formulations will be used
  - Caution: uncertain whether patterns will emerge for general NLP
- Authority allocation aspect:
  - Reconfiguration may involve transfer of authority to alternative decision-maker
- DOE of more realistic scenarios in development for data collection and complexity model construction.

Acknowledgments

The author thanks the NASA LaRC CIF/IRAD Program for supporting this work.