Deep Reinforcement Learning for Combat System-of-Systems Architectural Path Selection

Zhemei Fang

zmfang2018@hust.edu.cn

School of Artificial Intelligence and Automation
Huazhong University of Science and Technology

Author’s Bio Info

Zhemei Fang received her Ph.D. degree from the School of Aeronautics and Astronautics Engineering at Purdue University in 2017. She is currently an Assistant Professor in the School of Artificial Intelligence and Automation in Huazhong University of Science and Technology in Wuhan, China.

Her primary research interests center on developing theories and methods for system-of-systems architecture design and evolution. The methods include but are not limited to model-based systems engineering, discrete event simulation, approximate dynamic programming, and reinforcement learning.

Outline

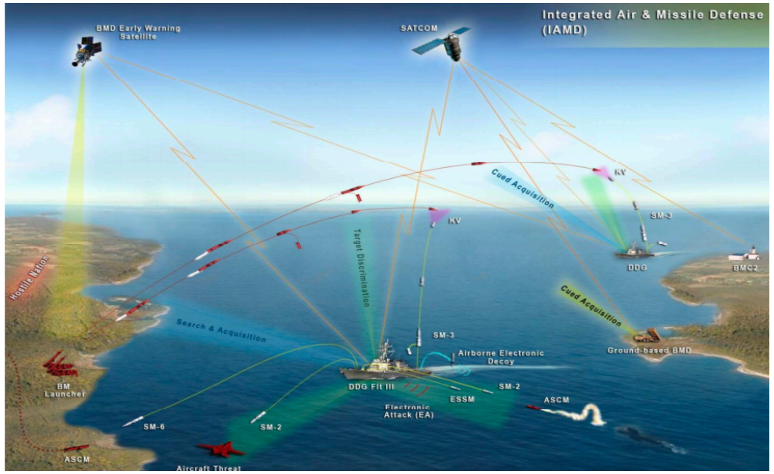
1 Motivation & Research Objective

2 Proposed Approach

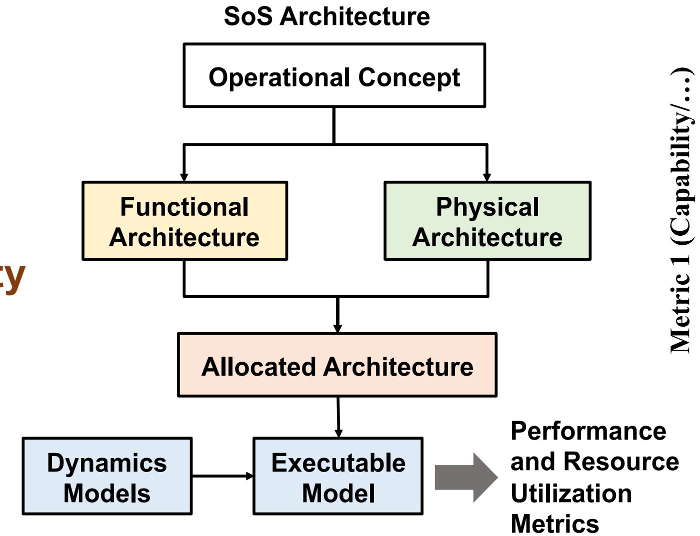
3 Illustrative Study

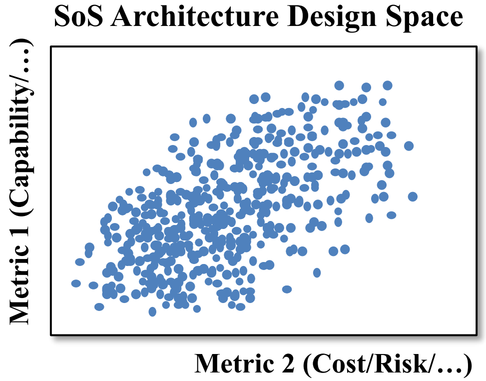
4 Conclusion & Future Work

Motivation & Research Objective



Complexity





Architecture Selection

Difficulties

Interdependency | **Evolution + Uncertainty**

Research Objective: Develop a learning-based framework for managing interdependency-incorporated System-of-Systems (SoS) architecture path selection

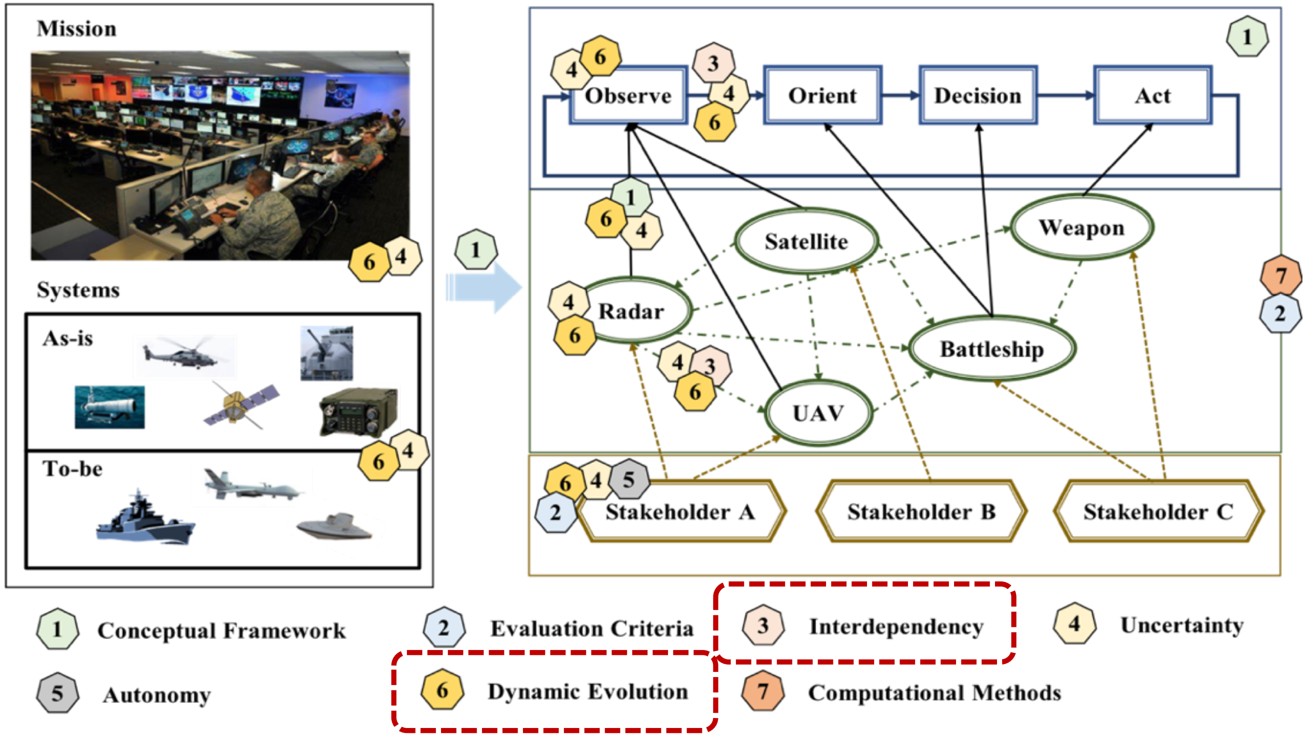
Related Work

SoS Architecture Selection

Simulation-based Methods
- Agent-based simulation
- Petri net based simulation
- Parametric models

Trade-off Space Exploration
- Multi-attribute tradespace exploration (MATE)

Optimization-based Methods
- Mathematical optimization: Evolutionary algorithms
- Modern portfolio theory



Related Work

Impact of Interdependency

Simulation-based Methods
- Message passing between different agents
- Token flows between places and transitions
- Dependency parameters; Conditional probability

Optimization-based Methods
- Synergy value
- Compatibility/prerequisite constraints
- Co-variance

Architectural Path Selection

Dynamic Strategic Planning based Methods
- Real options analysis
- Epoch-era analysis
- Time-expanded decision network

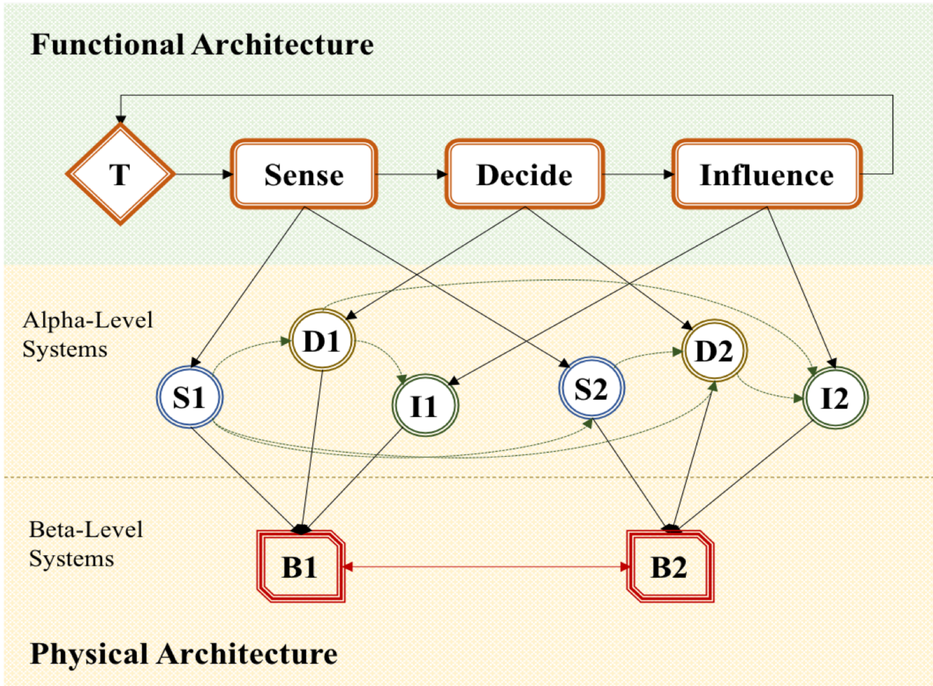
Algorithm based Methods
- Evolutionary algorithms
- **Approximate dynamic programming (previous work)**

Issues:
- **Interdependency in the optimal architecture selection problems has not been well captured**
- **Inadequate support for multi-stage architecture development**

Weighted sum of capability → inadequate
Linear value function approximation → inadequate

SoS Architectural Path Selection Problem

Representation of SoS Architecture



Functional Architecture

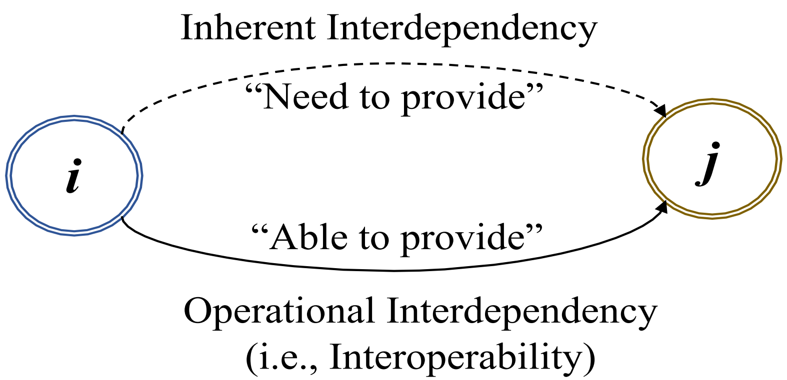
- Adopts "Information Age Combat Model (IACM)" proposed by Jeffrey Cares
- IACM includes elements: Target (T), Sensor (S), Decider (D), Influencer (I)

Physical Architecture

- Alpha-level systems: sensor entities, Command and Control (C2) entities, and firing entities
- Beta-level systems: integrated platform including sensor, C2, and firing entities

SoS Architectural Path Selection Problem

Impact of Interdependency



$$cap_j = \xi_{ij} \cdot \delta_{ij} \cdot cap_i + \left(1 - \xi_{ij}\right) \cdot SC_j$$

Inherent Interdependency

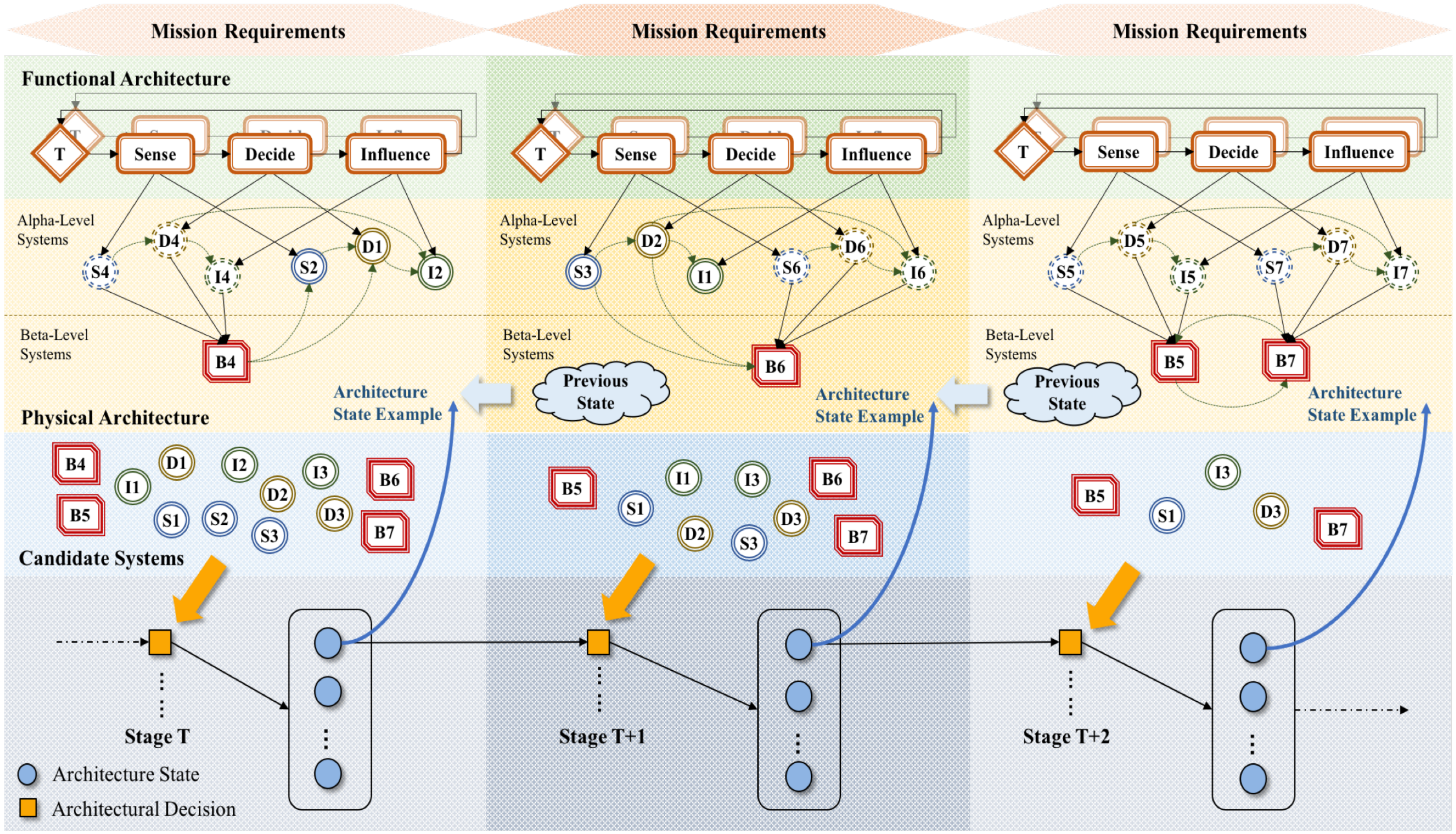
- Inherent functional need that centers on required information exchange between two nodes
- ξ_{ij}: degree of dependency
  - $\xi_{ij} = 0$: node j can work well without any support of node i
  - $\xi_{ij} = 1$: node j cannot work at a;; without information from node i

Operational Interdependency

- Functional capability of conducting satisfactory information exchange between two nodes
- δ_{ij}: interoperability level
  - $\delta_{ij} = 1$: full interoperability
  - $\delta_{ij} = 0$: zero interoperability

SoS Architectural Path Selection Problem

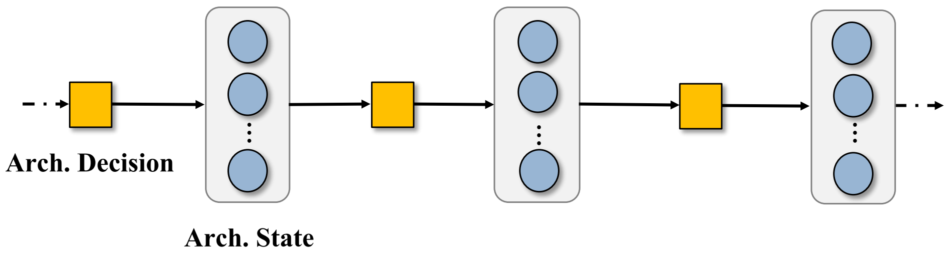
SoS Architectural Path Selection



Path Selection Process

- Sequential allocation of candidate systems to functional architecture
- Objective: Maximize combat mission capability
- Assumption
  - Candidate systems are given
  - Functional architectures are given and fixed

Mathematical Formulation



Markovian Property Satisfaction

- An SoS architecture state (analogous to MDP state) contains full information of the architectural history
- An architecture state depends only on the architecture state of the previous decision stage

MDP tuple

$$M = < S, A, P, R, \gamma >$$

Bellman Equation

$$Q_t(S_t, X_t) = \mathrm{E}\left[cap_{kt}^{I} + \gamma \max_{X_{t+1}} Q_{t+1}(S_{t+1}, X_{t+1})\right] \Leftarrow \max_{\{x_{it}^{S}, x_{jt}^{D}, x_{kt}^{I}\}} \sum_{t=1}^{T} cap_{kt}^{I}$$

State variables

$$S_t = \{R_t, SC_t, \xi_t, \delta_t, cost_t, CapReq_t, BG_t\}$$

Decision variables

$$X_t = \{x_{it}^{S}, x_{jt}^{D}, x_{kt}^{I}\}$$

Reward function refers to the operational capability of node Influence cap_{kt}^{I} that indicates the threat neutralization capability at each decision stage.

Reward function

$$cap_{kt}^{I} = \xi_{jkt} \cdot \delta_{jkt} \cdot (\xi_{ijt} \cdot \delta_{ijt} \cdot SC_{it}^{S} \cdot x_{it}^{S} + (1 - \xi_{ijt}) \cdot SC_{jt}^{D} \cdot x_{jt}^{D}) + (1 - \xi_{jkt}) \cdot SC_{kt}^{I} \cdot x_{kt}^{I}$$

MDP: Markov Decision Process

Mathematical Formulation

Same for D and I

Transition function

$$SC_{i,t+\tau}^{S} = \min\left(SC_{i,t}^{S} + x_{i,t}^{S} \cdot \sum_{m=t}^{t+\tau-1} \widehat{dev}_{i,t}^{S}, SCU_{i}^{S}\right), \tau \in [1, T-t]$$

$$BG_{t+1} = BG_{t} - \left(\sum_{i \in N_t^S} cost_{it}^{S} \cdot x_{it}^{S} + \sum_{j \in N_t^D} cost_{jt}^{D} \cdot x_{jt}^{D} + \sum_{k \in N_t^I} cost_{kt}^{I} \cdot x_{kt}^{I}\right) + BG_{t+1}^{new}$$

Architecture Constraints

Developmental budget constraint

$$\sum_{i \in N_t^S} cost_{it}^{S} \cdot x_{it}^{S} + \sum_{j \in N_t^D} cost_{jt}^{D} \cdot x_{jt}^{D} + \sum_{k \in N_t^I} cost_{kt}^{I} \cdot x_{kt}^{I} \leq BG_t$$

Intermediate capability demand

$$\sum_{t} cap_{kt}^{I} \geq CapReq_t$$

Must-selection constraint

$$\sum_{i \in N_t^S} x_{it}^{S} = 1 \qquad \sum_{j \in N_t^D} x_{jt}^{D} = 1 \qquad \sum_{k \in N_t^I} x_{kt}^{I} = 1$$

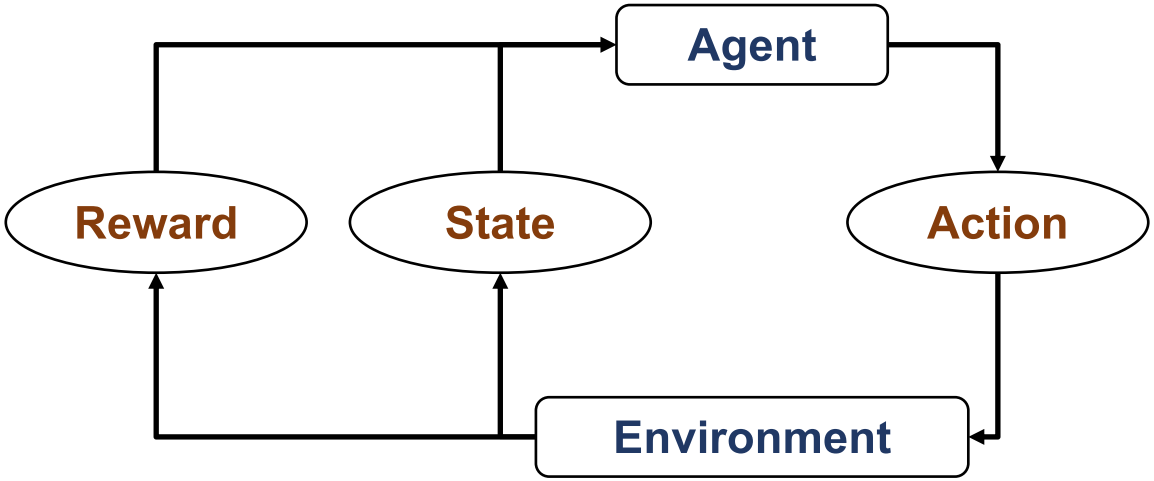
Beta-level system selection constraint

$$\left(x_{it}^{S,beta} \cdot x_{jt}^{D,beta}\right) + \left(1 - x_{it}^{S,beta}\right) \cdot \left(1 - x_{jt}^{D,beta}\right) \geq 1$$

$$\left(x_{it}^{S,beta} \cdot x_{kt}^{I,beta}\right) + \left(1 - x_{it}^{S,beta}\right) \cdot \left(1 - x_{kt}^{I,beta}\right) \geq 1$$

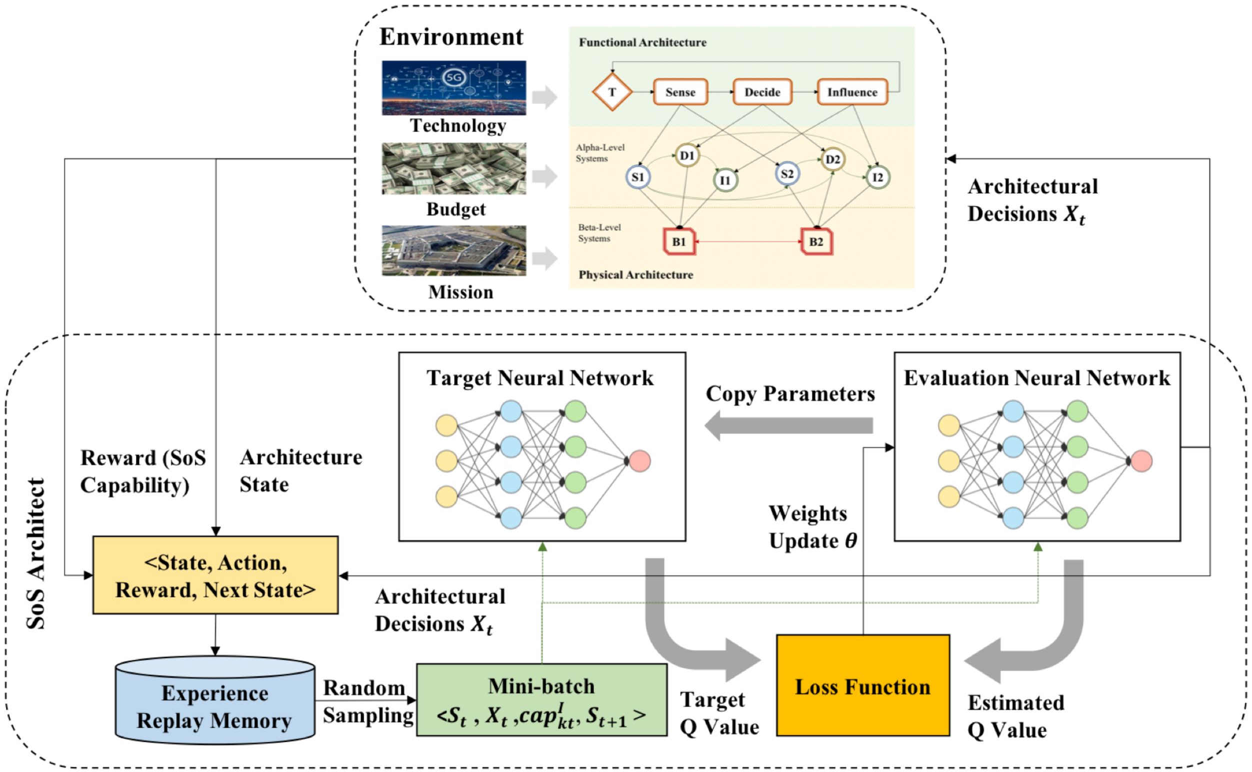
Approach: Deep Reinforcement Learning (DRL)

Brief Introduction to Reinforcement Learning



- No supervisor → Learning based on feedback and improvement
- Sequential decision-making
- Q learning
  - Value-based learning algorithm that updates the action-value function Q(S,X) based on Bellman Equation
  - Simple problems can directly use Q-Table Q(S,X) to calculate and store the maximum expected future rewards for action at each state
  - For high-dimensional problems, Q-Table is insufficient

Approach: Deep Reinforcement Learning (DRL)



$$Loss(\theta_n) = \mathrm{E}\left[\left(\left(cap^I_{kt} + \gamma \max_{X_t} \bar{Q}_{t+1}(S_{t+1}, X_{t+1}; \theta_n^-) - \bar{Q}_t(S_t, X_t; \theta_n\right)\right)^2\right]$$

Bellman Equation

$$Q_t(S_t, X_t) = \mathrm{E}\left[cap^I_{kt} + \gamma \max_{X_{t+1}} Q_{t+1}(S_{t+1}, X_{t+1})\right]$$

Techniques: Deep Q Network (DQN)

- Q learning
- Functional approximator: Convolutional Neural Network (CNN)
- Experience replay: remove correlation between observation sequences
- Target network + evaluation network
- Parameter noise: support exploration

Illustrative Study

Mosaic Warfare

- Decompose monolithic multi-mission units into a larger number of smaller elements
- Distributed maritime operations
  - ISR USV, C2 USV and AMD USV with different TRLs
- Interdependency
  - Left: inherent (L = 0.1, M = 0.4, H = 0.8)
  - Right: interop (L = 0.1, M = 0.5, H = 1)

Candidate systems and input parameters

Example	Sys	Self-Cap Limit	Initial Self-Cap	Dev Rate	Cost (m$)
ISR USV Class A Radar I	S1	40	40	(0,0)	80
ISR USV Class B Radar I	S2	40	40	(0,0)	120
ISR USV Class A Radar II	S3	50	40	(1,1)	110
ISR USV Class B Radar II	S4	50	40	(1,1)	130
ISR USV Class A Radar III	S5	60	36	(2,1)	150
ISR USV Class B Radar III	S6	60	36	(2,1)	180
ISR USV Class C Radar III	S7	70	35	(3,1)	160
……	…	…	…	…	…

Interdependency between sensor-decider system pairs

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14
S1	HL	HL	HL	HL	HL	HL	ML	ML	ML	ML	ML	ML	LL	LL
S2	HL	HM	HL	HM	HM	HM	MM	MM	MM	MM	MM	MM	LM	LM
S3	HL	HL	HL	HL	HL	HL	ML	ML	ML	ML	ML	ML	LL	LL
S4	HL	HM	HL	HM	HM	HM	MM	MM	MM	MM	MM	MM	LM	LM
S5	HL	HM	HL	HM	MM	MM	MM	MM	MM	MM	LM	LM	LM	LM
……	…	…	…	…	…	…	…	…	…	…	…	…	…	…

ISR: Intelligence, Surveillance and Reconnaissance; AMD: Air and Missile Defense; USV: Unmanned Surface Vehicle; TRL: Technology Readiness Level

Illustrative Study

Hyperparameters	Values
Mini-batch size	512
Discount factor	0.9
Q learning rate (stepsize)	0.8
Experience replay memory size	10000
Target network update frequency	500
CNN network layer	4
CNN learning rate	0.001
Optimizer	Adam
Activation function	ReLU
Noise	N(0,1)

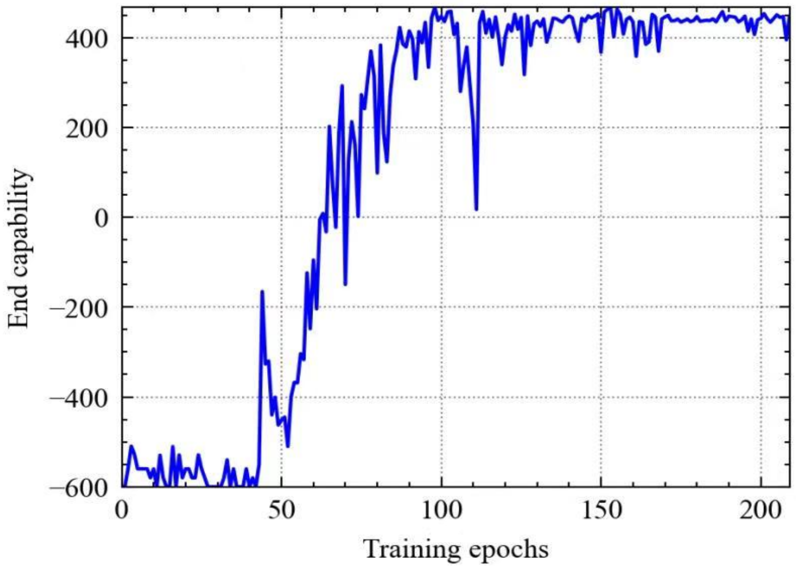
Additional Input

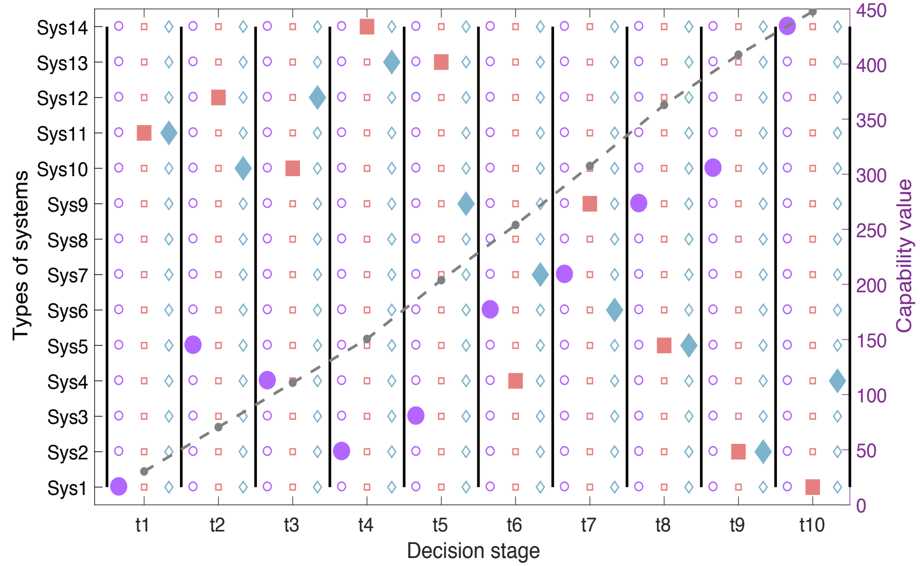
- Horizon: 10 stages
- Budget: 1000 m$ initially; receives additional 500 m$ each stage
- Intermediate requirement: Stage 3 >= 100 units; Stage 6 >= 250 units

Environment

- Python 3.7
- PyTorch machine learning library
- AMD Ryzen-2600X CPU
- 16GB 3000MHZ memory
- GeForce RTX-2080 GPU
- WIN 10 operating system

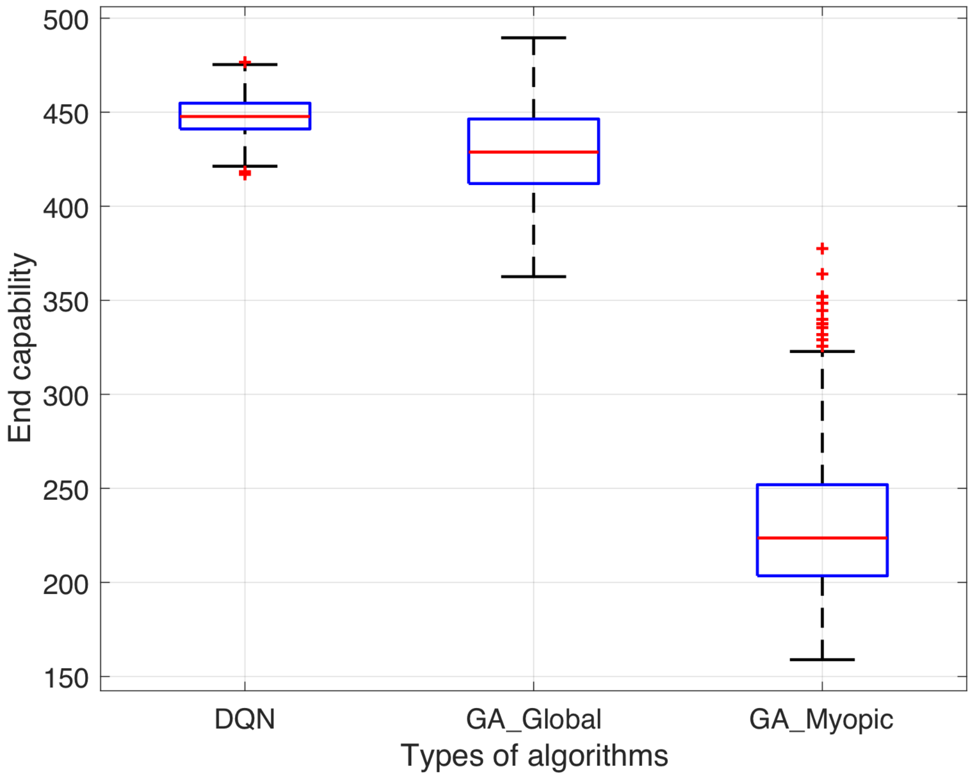
Illustrative Study





- Left: training curve tacking total capability of combat SoS.
  - Capability of the entire SoS architecture converges after around 150 training epochs.
  - Fluctuation still exists but remains stable and the MSE loss function stays around 2.
- Right: most often suggested architectural path and resultant capability (circle: S; square: D; diamond: I).
  - Decider systems and influencer systems with advanced technologies are often selected at the beginning whereas sensor systems with high readiness level are selected.
  - SoS architect agent is smart enough to choose those systems with large potentials in the future.

Illustrative Study



- **DQN algorithm** provides the largest average capability value of 448 units and lowest standard deviation of 10 units.

  → DQN provides more stable solution in an uncertain environment
- **GA_Global** provides slightly lower average capability value (429 units) and larger standard deviation (24 units) than DQN algorithm.
- **GA_Myopic** offers lowest capability value of 232 units since it only sees the current situation without any consideration of future possibility.

- **GA_Global:** assumes full predictability of future information and establishes a big optimization problem containing all architectural path decisions at ten stages.
- **GA_Myopic:** only considers now and does not account for any information from the future.

Conclusion & Future Work

- Conclusion
  - Developed DQN algorithm to support the SoS architectural path selection under uncertainty;
  - Built a simple parametric model to capture the impact of interdependency on SoS capability;
  - Applied the method to a synthetic USV-centered naval AMD SoS.
- Future Work
  - Assumption relaxation, such as one system for one type of function;
  - Development of more effective DRL algorithms and techniques;
  - Robust and resilient architectural path selection.

References

[1] "DARPA Tiles Together a Vision of Mosaic Warfare: Banking on cost-effective complexity to overwhelm adversaries," Defense Advanced Research Projects Agency (DARPA), 2018. [Online]. Available: https://www.darpa.mil/work-with-us/darpa-tiles-together-a-vision-of-mosiac-warfare. [Accessed 2020].

[2] B. Clark, D. Patt and H. Schramm, "Mosaic Warfare: Exploiting Artificial Intelligence and Autonomous Systems to Implement Decision-Centric Operations," Center for Strategic and Budgetary Assessments, 2020.

[3] M. Maier, "Architecting Principles for Systems-of-Systems," Systems Engineering, vol. 1, no. 4, pp. 267-284, 1998.

[4] A. P. Sage and C. D. Cuppan, "On the Systems Engineering and Management of Systems of Systems and Federations of Systems," Information, Knowledge, Systems Management, vol. 2, no. 4, pp. 325-345, 2001.

[5] E. Crawley, B. Cameron and D. Selva, System Architecture: Strategy and Product Development for Complex Systems, Pearson Education Limited, 2016.

[6] A. H. Levis and L. W. Wagenhals, "C4ISR Architectures: I. Developing a Process for C4ISR Architecture Design," Systems Engineering, vol. 3, no. 4, pp. 225-247, 2000.

[7] R. Kenley, D. Timothy, W. Paul and D. DeLaurentis, "Synthesizing and Specifying Architectures for System of Systems," in 24th Annual INCOSE International Symposium, Las Vegas, NV, USA, 2014.

[8] A. K. Raz, C. R. Kenley and D. DeLaurentis, "A System-of-Systems Perspective for Information Fusion System Design and Evaluation," Information Fusion, vol. 35, pp. 148-165, 2017.

[9] I. Lluch and A. Golkar, "Architecting Federations of Systems: A Framework for Capturing Synergy," Systems Engineering, vol. 22, pp. 295-312, 2019.

[10] H. Chen, A. Lapin, C. Lei, K. Ho and T. Ukai, "Optimization for Large-Scale Multi-Mission Space Campaign Design by Approximate Dynamic Programming," in AIAA Space Forum, Orlando, FL, USA, 2018.

[11] Z. Fang, N. Davendralingam and D. DeLaurentis, "Multistakeholder Dynamic Optimization for Acknowledged System-of-Systems Architecture Selection," IEEE Systems Journal, vol. 12, no. 4, pp. 3565-3576, 2018.

[12] Z. Fang, X. Zhou and A. Song, "Architectural Models Enabled Dynamic Optimization for System-of-Systems Evolution," Complexity, vol. 7534819, pp. 1-14, 2020.

[13] V. Mnih, K. Kavukcuoglu, D. Silver and e. al., "Human-Level Control Through Deep Reinforcement Learning," Nature, vol. 518, no. 7540, pp. 529-533, 2015.

[14] D. A. DeLaurentis, W. A. Crossley and M. Mane, "Taxonomy to Guide Systems-of-Systems Decision-Making in Air Transportation Problems," Journal of Aircraft, vol. 48, no. 3, pp. 760-770, 2011.

[15] W. C. Baldwin, B. Sauser and R. Cloutier, "Simulation Approaches for System of Systems: Events-Based versus Agent Based Modeling," in Conference on Systems Engineering Research (CSER), Hoboken, NJ, USA, 2015.

[16] P. T. Biltgen, "A Methodology for Capability-based Technology Evaluation for Systems-of-Systems," Georgia Institute of Technology, 2007.

[17] J. C. Domercant, "ARC-VM: An Architecture Real Options Complexity-Based Valuation Methodology for Military Systems-of-Systems Acquisitions," Georgia Institute of Technology, 2011.

[18] J. V. Iacobucci, "Rapid Architecture Alternative Modeling (RAAM): A Framework for Capability-based Analysis of System of Systems Architectures," Georgia Institute of Technology, 2012.

[19] K. Jacobson, "The Littoral Combat Ship (LCS) Surface Warfare (SUW) Module: Determining the Best Mix of Surface-to-Surface and Air-to-Surface Missiles," Naval Postgraduate School, 2010.

[20] C. H. Popa, S. P. Stone, E. H. Aw, C. P. J. Teo, L. E. Cai and e. al., "Distributed Maritime Operations and Unmanned Systems Tactical Employment," Naval Postgraduate School, 2018.

References

[21] P. J. Winstead, "Implementation of Unmanned Surface Vehicles in the Distributed Maritime Operations Concept," Naval Postgraduate School, 2018.

[22] A. Mour, C. R. Kenley, N. Davendralingam and D. DeLaurentis, "Agent-based Modeling for Systems of Systems," in INCOSE International Symposium, Philadelphia, PA, USA, 2013.

[23] L. W. Wagenhals and A. H. Levis, "Service Oriented Architectures, the DoD Architecture Framework 1.5, and Executable Architectures," Systems Engineering, vol. 12, no. 4, pp. 312-343, 2009.

[24] Z. Fang, D. DeLaurentis and N. Davendralingam, "An Approach to Facilitate Decision Making on Architecture Evolution Strategies," in Conference on Systems Engineering Research (CSER), Atlanta, GA, USA, 2013.

[25] R. Wang, S. Agarwal and C. Dagli, "Executable System of Systems Architecture using OPM in Conjunction with Colored Petri Net: A Module for Flexible Intelligent and Learning Architectures for System of Systems," in INCOSE International Symposium, Cape Town, South Africa, 2015.

[26] S. Han, Z. Fang and D. DeLaurentis, "Acquisition Management for System-of-Systems: Requirement Evolution and Acquisition Strategy Planning," in Proceedings of the Ninth Annual Acquisition Research Symposium, Monterey, CA, USA, 2012.

[27] Q. Zhao, S. Li, Y. Dou, X. Wang and K. Yang, "An Approach for Weapon System-of-Systems Scheme Generation based on a Supernetwork Granular Analysis," IEEE Systems Journal, vol. 11, no. 4, pp. 1971-1982, 2017.

[28] P. R. Garvey and C. A. Pinto, "Introduction to Functional Dependency Network Analysis," in Second International Symposium on Engineering Systems, Cambridge, MA, USA, 2009.

[29] C. Guariniello and D. DeLaurentis, "Supporting Design via the System Operational Dependency Analysis Methodology," Research Engineering Design, vol. 28, pp. 53-69, 2017.

[30] P. D. Vascik, A. M. Ross and D. H. Rhodes, "A Method for Exploring Program and Portfolio Affordability Tradeoffs under Uncertainty using Epoch-Era Analysis: A Case Application to Carrier Strike Group Design," in Proceedings of 12th Annual Acquisition Research Symposium, 2015.

[31] J. W. Dieffenbacher, "Managing Portfolios of Complex Systems with the Portfolio-Level Epoch-Era Analysis for Affordability Framework," Massachusetts Instittue of Technology, 2018.

[32] S. E. Gillespie, R. E. Giachetti, A. Hernandez, P. T. Beery and E. P. Paulo, "System of Systems Architecture Feasibility Analysis to Support Tradespace Exploration," in 12th System of Systems Engineering Conference (SoSE), Waikoloa, HI, USA, 2017.

[33] G. Brown, R. Dell, H. Holtz and A. Newman, "How US Air Force Space Command Optimizes Long-Term Investment in Space Systems," INFORMS: Interfaces, vol. 33, no. 4, pp. 1-14, 2003.

[34] R. Burk and G. Parnell, "Portfolio Decision Analysis: Lessons from Military Applications," in Portfolio Decision Analysis: Improved Methods for Resource Allocation, 2011, pp. 333-357.

[35] D. Konur, H. Farhangi and C. H. Dagli, "A Multi-Objective Military System of Systems Architecting Problem with Inflexible and Flexible Systems: Formulation and Solution Methods," OR Spectrum, vol. 38, pp. 967-1006, 2016.

[36] M. Li, M. Li, K. Yang, B. Xia and C. Wan, "A Network-based Portfolio Optimization Approach for Military System of Systems Architecting," IEEE Access, vol. 6, pp. 53452-53472, 2018.

[37] K. Shafi, S. Elsayed, R. Sarker and M. Ryan, "Scenario-Based Multi-Period Program Optimization for Capability-based Planning using Evolutionary Algorithms," Applied Soft Computing, vol. 56, pp. 717-729, 2017.

[38] H. Markowitz, "Portfolio Selection," The Journal of Finance, vol. Vol.7, no. No.1, pp. pp.77-91, 1952.

[39] N. Ricci and A. Ross, "Developing a Dynamic Portfolio-Based Approach for Systems-of-Systems Composition," Systems Engineering Advancement Research Initiative, 2012.

[40] N. Davendralingam, M. Mane and D. DeLaurentis, "Capability and Development Risk Management in System-of-Systems Architectures: A Portfolio Approach to Decision-Making," in Proceedings of Ninth Annual Acquisition Research Symposium, 2012.

[41] N. Davendralingam and D. DeLaurentis, "A Robust Portfolio Optimization Approach to System of System Architectures," Systems Engineering, vol. 18, no. 3, pp. 269-283, 2015.

[42] J. Dahmann, "System of Systems Pain Points," in INCOSE International Symposium, 2014.

[43] M. Ouyang, "Review on Modeling and Simulation of Interdependent Critical Infrastructure Systems," Reliability Engineering and System Safety, vol. 121, pp. 43-60, 2014.

[44] N. Davendralingam and D. DeLaurentis, "A Robust Portfolio Optimization Approach to System of System Architectures," Systems Engineering, vol. 18, no. 3, pp. 269-283, 2015.

[45] S. A. Selberg and M. A. Austin, "Toward an Evolutionary System of Systems Architecture," in INCOSE International Symposium, 2008.

References

[46] R. de Neufville, "Dynamic Strategic Planning for Technology Policy," International Journal of Technology Management, vol. 19, pp. 225-245, 2000.

[47] J. Buurman, S. Zhang and V. Babovic, "Reducing Risk Through Real Options in System Design: The Case of Architecting a Martime Domain Protection System," Risk Analysis, vol. 29, no. 3, pp. 366-379, 2009.

[48] M. E.Fitzgerald and A. M.Ross, "Mitigating Contextual Uncertainties with Valuable Changeability Analysis in the Multi-Epoch Domain," in The 6th Annual IEEE International Systems Conference, 2012.

[49] M. E.Fitzgerald and A. M.Ross, "Sustaining Lifecycle Value: Valuable Changeability Analysis with Era Simulation," in The 6th Annual IEEE International Systems Conference, 2012.

[50] M. Silver and O. De Weck, "Time-Expanded Decision Networks: A Framework for Designing Evolvable Complex Systems," Systems Engineering, vol. 10, no. 2, pp. 167-186, 2007.

[51] A. Siddiqi, E. Rebentisch, S. Dorchuck, Y. Imanishi and T. Tanimuchi, "Optimizing Architecture Transitions using Decision Networks," Journal of Mechanical Design, vol. 142, no. 121702, pp. 1-14, 2020.

[52] P. Davison, B. Cameron and E. F. Crawley, "Technology Portfolio Planning by Weighted Graph Analysis of System Architectures," Systems Engineering, vol. 18, no. 1, pp. 45-58, 2015.

[53] W. Tan, B. J. Sauser, J. E. Ramirez-Marquez and R. B. Magnaye, "Multiobjective Optimization in Multifunction Multicapability System Development Planning," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 43, no. 4, pp. 785-799, 2013.

[54] P. Acheson, L. Pape, C. Dagli, N. Kilicay-Ergin, J. Columbi and K. Haris, "Understanding System of Systems Development using an Agent-Based Wave Model," in Complex Adaptive Systems, 2012.

[55] N. Kilicay-Ergin and C. H. Dagli, "Incentive-based Negotiation Model for System of Systems Acquisition," Systems Engineering, vol. 18, no. 3, pp. 310-321, 2015.

[56] W. B.Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, 2 ed., John Wiley & Sons, Inc., 2010.

[57] M. Maier, "Research Challenges for Systems-of-Systems," in IEEE International Conference on Systems, Man and Cybernetics, 2005.

[58] Z. Fang and D. DeLaurentis, "Dynamic Planning of System of Systems Architecture Evolution," in Conference on Systems Engineering Research (CSER), Redondo Beach, CA, USA, 2014.

[59] Z. Fang and D. DeLaurentis, "Multi-Stakeholder Dynamic Planning of System of Systems Development and Evolution," in Conference on Systems Engineering Research (CSER), Hoboken, NJ, USA, 2015.

[60] N. Davendralingam and D. DeLaurentis, "Promoting Affordability in Defense Acquisitions: A Multi-Period Portfolio Approach," in Proceedings of the 11th Acquisition Research Symposium, 2014.

[61] J. R. Cares, "An Information Age Combat Model," in 9th International Command and Control Research and Technology Symposium, Copenhagen, Denmark, 2004.

[62] D. DeLaurentis, "Understanding Transportation as System-of-Systems Design Problem," in 43rd AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2005.

[63] E. A. J. Wyatt, "A Reliability-Based Measurement of Interoperability for Conceptual-Level Systems of Systems," Georgia Institute of Technology, 2014.

[64] Y. Li, "Deep Reinforcement Learning," arXiv: 1810.06339, 2018.

[65] L. Xiao, X. Lu, T. Xu, X. Wan, W. Ji and Y. Zhang, "Reinforcement Learning-based Mobile Offloading for Edge Computing against Jamming and Interference," IEEE Transactions on Communications, vol. 68, no. 10, pp. 6114-6126, 2020.

[66] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. Yogamani and P. Perez, "Deep Reinforcement Learning for Autonomous Driving: A Survey," IEEE Transactions on Intelligent Transportation System, pp. 1-18, 2021.

[67] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction (Second Edition), The MIT Press, 2017.

[68] M. Fortunato, M. G. Azar, B. Piot, J. Menick and e. al., "Noisy Networks for Exploration," 2017. [Online]. Available: https://arxiv.org/pdf/1706.10295v1.pdf. [Accessed 2020].

[69] S. Savitz, I. Blickstein, P. Buryk, R. W. Button, P. DeLuca, J. Dryden and e. al., "U.S. Navy Employment Options for Unmanned Surface Vehicles (USVs)," RAND Corporation, 2013.

[70] C. Paul, C. P. Clarke, B. L. Triezenberg, D. Manheim and B. Wilson, "Improving C2 and Situational Awareness for Operations in and Through the Information Environment," RAND Corporation, 2018.