Learning to Control Reconfigurable Multiagent Systems

Doctoral Consortium

Manuel Agraz Vallejo
Collaborative Robotics and Intelligent Systems Institute
Oregon State University
Corvallis, United States of America
agrazvam@oregonstate.edu

ABSTRACT

Reconfigurable multiagent systems (RMSs) are composed of individual agents capable of docking and undocking together to form composite agents with new capabilities that can adapt to diverse tasks. One such system, salp-inspired agents (simple thrust-based agents that can form chains for locomotion) have the potential for scientific monitoring in topologically intricate underwater habitats, such as caves, overhangs, and confined openings. Current approaches focus on controlling the locomotion of single salp units or small salp chains that are limited to operating on a fixed-size structure. However, this limits the scalability and robustness inherent to the salp chain design and requires new controllers for new chain configurations. In our work, we introduce a set of graph-based neuro-controllers whose structures directly map onto salp chains of arbitrary length. By representing salp units as nodes in a graph, we integrate the chain's structure directly into the controller's architecture, eliminating the need to redesign the controller whenever the chain grows or shrinks. Our results show that graph-based controllers retain up to 90% performance under zero-shot settings, demonstrating scalability and robustness to salp-unit failures.

KEYWORDS

Reinforcement learning; Graph-based control; Continuous control;

ACM Reference Format:
Manuel Agraz Vallejo. 2026. Learning to Control Reconfigurable Multiagent Systems: Doctoral Consortium. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 3 pages. https://doi.org/10.65109/UPSM9650

1 INTRODUCTION

Salps are marine organisms that propel themselves using multi-jet propulsion and self-assemble into large chains to move efficiently underwater [2, 16]. Salp-inspired agents [4, 9, 21] offer similar advantages for long-range underwater motion, making them well-suited for extended tasks such as ocean monitoring or environmental surveys [5]. Three main challenges stem from the modular design of salp chain agents. Firstly, the use of multi-jet propulsion limits the locomotion of individual salp-units to one DOF. Thus, locomotion of the chain requires a high level of coordination between salp-units. Secondly, the varying degrees of freedom of adding or removing salp-units require a new controller for every chain configuration. Finally, salp-units along the chain can fail, which requires the controller to adapt and retain performance under failure.

Seminal research in salp-inspired agent locomotion focuses on the control of single salp-units [4, 9, 22] and small salp chains (two to three units) [21, 23]. While these controllers are effective for small chains, the primary benefits of salp locomotion come from their ability to operate as larger chains. Additionally, as the chain grows in size, so does the risk of failure of individual salp-units. Thus, the challenge is how to maintain effective control of the salp chain as units are added, removed, or even fail.

Reinforcement Learning (RL) methods have shown success in training neural network (NN) controllers that handle tightly coupled, non-linear dynamics [7, 11, 14, 20]. Among the different NN architectures, graph neural networks (GNNs) [6, 18] in particular are well suited for controlling intricate structures with many interconnected joints [1, 8, 19]. They do this by aggregating information from neighboring nodes to determine the control action for individual nodes. However, for a salp chain, this means that the selection of graph topology or aggregation mechanism can have a significant impact on the salp chain's robustness and adaptability.

In our work, we introduce a set of graph-based controllers that embed the structure of the salp chain into a controller's architecture. To test the controllers' robustness and scalability, we perform zero-shot experiments across varying salp chain sizes and levels of salp-unit failure. To train the controllers, we introduce the Salp Chain Locomotion Domain (SCLD). Empirical results in the SCLD environment show that graph-based controllers maintain 90% of their performance on salp chains of up to twice the length used during training. Furthermore, we find that both local and global aggregation mechanisms can produce graph-based controllers with strong zero-shot performance.

2 BACKGROUND

Work on single salp units started with the emulation of salp jet propulsion by Bujard et al. [3]. They built a pulse-jet swimming agent whose cost-of-transport matches the one seen in efficient pulse-jet swimmers such as jellyfish and salps. Tackling the problem of linking multiple salp units into a chain, the authors [21] developed a ground-based version of a 3-unit salp chain and developed a controller using geometric mechanics. Finally, iterating on their origami-inspired salp-unit, Yang et al. [23] connected multiple salp units into a 2-unit chain, and showed that it provides a significant increase in speed underwater than a single unit. To control the salp chain, they use a fixed jet pulse sequence at a known rate.

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/UPSM9650

To leverage reinforcement learning methods [17], we model the salp chain locomotion problem as a Markov Decision Process (MDP). An MDP is a sequential decision-making process defined by five components: a set of states $\mathcal{S}$, actions $\mathcal{A}$, a reward function $\mathcal{R}(s)$, a transition function $\mathcal{P}(s'|s,a)$ and a policy $\pi(a|s)$. At each time step, given a state $s \in \mathcal{S}$, the agent takes an action $a \in \mathcal{A}$ following policy π. Afterwards, it receives the next state s' from the transition function $\mathcal{P}$ and finally obtains a reward from $\mathcal{R}(s') \rightarrow r$. The use of policy π induces a value function $V(s_t) = \mathbb{E}[R_t|s_t]$, where $R_t = \sum_{i=0}^{\infty} \gamma^i r_{i+1}$ is the discounted return from the episode.

Graph Neural Networks (GNN) are a class of neural network architectures that can operate on any graph, both directed and undirected [6]. A graph is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of all nodes, and $\mathcal{E}$ is the set of all edges. When it comes to locomotion tasks, often the structure of the agent is partitioned into nodes, and the edges between them define their kinematic relation [1, 13, 19]. For example, in [13], the authors partitioned a legged agent such that the state of each joint became a node and each connection between joints an edge in a graph.

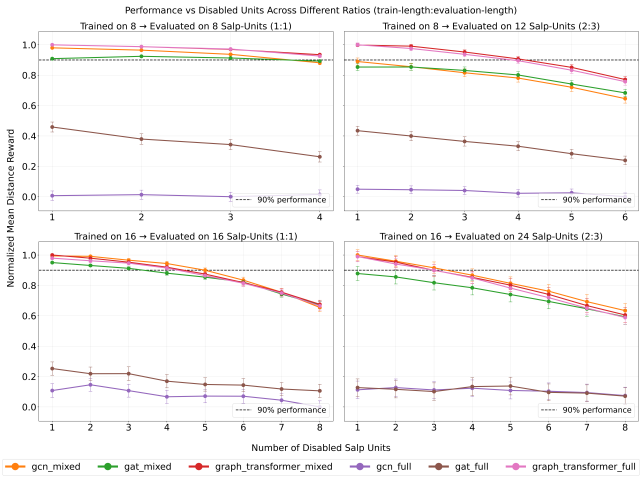
3 LEARNING SALP-INSPIRED LOCOMOTION

The Salp Chain Locomotion Domain (SCLD) simulates a 2D chain of linked salp-units. The main goal in this domain is to translate the entire chain to a target pose that changes in shape and position each episode. In the SCLD, each salp-unit can only generate forces to propel itself along one degree of freedom through forward or backward force. The links between salps connect them in a chain through revolute joints. Each salp-unit is assigned a reachable individual target position that is part of the complete target pose. The main locomotion challenge comes from the restricted motion of each salp-unit and the requirement to coordinate the salp-units' forces to produce both translation and rotation of the entire chain.

We use the SCLD to evaluate how different graph-based models perform on unseen salp chain lengths and under varying levels of salp-unit failure. We train graph-based policies (GCN, GATv2, GT) using a mixed and fully connected graph structure, and an MLP policy as a performance baseline using PPO. Two controllers are trained for each model-topology pair, one trained on an 8-salp-unit chain and another on 16, resulting in 6 total graph-based controllers and 1 baseline MLP controller. We perform zero-shot evaluation using policies trained on 8 and 16 salp-unit chains operating on an increasing number of disabled salp-units and unseen chain lengths. The maximum number of disabled units is 50% of the trained chain length, and the maximum length on evaluation is up to 40 units.

Our results show that training our GNN controllers on a fixed chain length, particularly the ones using the GT architecture, demonstrate varying levels of both scalability and robustness to failure under zero-shot conditions as seen in both Figures 1a and 1b. Additionally, we found that increasing the size of the chain length used for training also increases the scalability of our controller by retaining 90% performance on longer chains. Additionally, we found that the inverse relation is true when aiming for robustness. As the ratio of train-length to zero-shot-length increases, the GT controller trained on the longer 16 salp-unit configuration saw a 6.25% drop in the amount of disabled salp-units it could handle. On the other hand, at a ratio of 2:3, the GT controller trained on the 8 salp-unit configuration showed the same robustness as when tested with a 1:1 ratio. This implies that training on longer configurations increases scalability at a higher rate than robustness.

(a)



(b)

Figure 1: Zero-shot evaluations of policies trained on 8 and 16 salp units across varying salp chain lengths (Fig. 1a) and across increasing disabled units (Fig. 1b). A normalized mean distance reward of 1.0 implies that the salp chain was able to match the target pose fully. We can observe that as the ratio increases, the 90% performance threshold is crossed sooner.

4 PROPOSED RESEARCH

So far in our work, we've shown that GNN models can provide scalability and robustness in a salp chain locomotion setting. However, more complex tasks can require a sequence of docking and undocking of salp units to do obstacle avoidance, or adapt the salp chain shape to enclose or capture an objective, which cannot be solved with just locomotion control. In this respect, we propose abstracting the low-level locomotion into macro-actions to focus on the higher-level decision-making of when to dock/undock, and which chain configuration to select based on the task. For this, we intend to leverage the concept of skill chaining [10, 15] in RL. By using pre-trained skills such as spreading out, converging at a point, and breaking into subchains, among others. We aim to train the agent's policies using multiagent RL [12, 24] methods to learn to sequence these skills to solve a target enclosure task, where the agents have to break into units to avoid obstacles, and then regroup in a shape that can enclose an irregularly shaped target.

ACKNOWLEDGMENTS

I would like to thank my advisor, Kagan Tumer; my friends at OSU, and my labmates at the AADI Lab.

REFERENCES

[1] Charlie Blake, Vitaly Kurin, Maximilian Igl, and Shimon Whiteson. 2021. Snowflake: Scaling GNNs to High-Dimensional Continuous Control via Parameter Freezing. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*. 23983–23992. https://proceedings.neurips.cc/paper/2021/file/c952ce98517ac529c60744ac28364b03-Paper.pdf

[2] Q. Bone and E. R. Trueman. 1983. Jet propulsion in salps (Tunicata: Thaliacea). *Journal of Zoology* 201, 4 (1983), 481–506. https://doi.org/10.1111/j.1469-7998.1983.tb05071.x

[3] Thierry Bujard, Francesco Giorgio-Serchi, and Gabriel D. Weymouth. 2021. A Resonant Squid-Inspired Robot Unlocks Biological Propulsive Efficiency. *Science Robotics* 6, 50 (January 2021), eabd2971. https://doi.org/10.1126/scirobotics.abd2971

[4] Xue Dong, Hao Chen, Zhitong Zhou, Chen Ouyang, Lei Hu, Fang Zhang, Bo Chen, and Zhen Gan. 2024. Salpot: A Jet Propulsion Swimmer with Scissor Structure and Bilateral Apertures. *IEEE Robotics and Automation Letters* 9, 8 (Aug. 2024), 7102–7109. https://doi.org/10.1109/LRA.2024.3418278

[5] Dennis P. Gordon, Jennifer Beaumont, Alison MacDiarmid, Donald A. Robertson, and Shane T. Ahyong. 2010. Marine Biodiversity of Aotearoa New Zealand. *PLoS ONE* 5, 8 (2010), e10905. https://doi.org/10.1371/journal.pone.0010905

[6] Marco Gori, Gabriele Monfardini, and Franco Scarselli. 2005. A New Model for Learning in Graph Domains. In *Proceedings of the 2005 IEEE International Joint Conference on Neural Networks (IJCNN '05)*, Vol. 2. IEEE, Montreal, QC, Canada, 729–734. https://doi.org/10.1109/IJCNN.2005.1555942

[7] Yaqi Guo and Haijun Peng. 2021. Full-Actuation Rolling Locomotion with Tensegrity Robot via Deep Reinforcement Learning. In *2021 5th International Conference on Robotics and Automation Sciences (ICRAS)*. 51–55. https://doi.org/10.1109/ICRAS52289.2021.9476651

[8] Wenlong Huang, Igor Mordatch, and Deepak Pathak. 2020. One Policy to Control Them All: Shared Modular Policies for Agent-Agnostic Control. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 4455–4464. https://proceedings.mlr.press/v119/huang20d.html

[9] Ali Jones and J. R. Davidson. 2024. Underwater Salp-Inspired Soft Structure Contraction with Twisted Coiled Actuators. In *Proceedings of the 7th IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, San Diego, CA, USA, 504–510. https://doi.org/10.1109/RoboSoft60065.2024.10522044

[10] George Konidaris and Andrew Barto. 2009. Skill Discovery in Continuous Reinforcement Learning Domains using Skill Chaining. In *Advances in Neural Information Processing Systems*, Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta (Eds.), Vol. 22. Curran Associates, Inc.

[11] Ashish Kumar, Zhongyu Li, Jun Zeng, Deepak Pathak, Koushil Sreenath, and Jitendra Malik. 2022. Adapting Rapid Motor Adaptation for Bipedal Robots. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 1161–1168. https://doi.org/10.1109/IROS47612.2022.9981091

[12] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529–533. https://doi.org/10.1038/nature14236

[13] Gaukhar Nurbek. 2024. *Exploring Graph Neural Networks in Reinforcement Learning: A Comparative Study on Architectures for Locomotion Tasks*. Master's Thesis. The University of Texas Rio Grande Valley, Edinburg, TX, USA. https://scholarworks.utrgv.edu/etd/1493

[14] Junyao Shi, Tony Dear, and Scott David Kelly. 2020. Deep Reinforcement Learning for Snake Robot Locomotion. In *21st IFAC World Congress*. 9688–9695. https://doi.org/10.1016/j.ifacol.2020.12.2581

[15] Martin Stolle and Doina Precup. 2002. Learning Options in Reinforcement Learning. In *Abstraction, Reformulation, and Approximation*, Sven Koenig and Robert C. Holte (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 212–223.

[16] Kelly R. Sutherland and Daniel Weihs. 2017. Hydrodynamic advantages of swimming by salp chains. *Journal of the Royal Society Interface* 14, 133 (2017), 20170298. https://doi.org/10.1098/rsif.2017.0298

[17] Richard S. Sutton, David A. McAllester, Satinder P. Singh, and Yishay Mansour. 1999. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Advances in Neural Information Processing Systems 12*. MIT Press, 1057–1063.

[18] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations* (2018). https://openreview.net/forum?id=rJXMpikCZ

[19] Tingwu Wang, Renjie Liao, Jimmy Ba, and Sanja Fidler. 2018. NerveNet: Learning Structured Policy with Graph Neural Networks. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*. https://openreview.net/forum?id=S1sqHMZCb

[20] Yuliu Wang, Ryusuke Sagawa, and Yusuke Yoshiyasu. 2024. Learning Advanced Locomotion for Quadrupedal Robots: A Distributed Multi-Agent Reinforcement Learning Framework with Riemannian Motion Policies. *Robotics* 13, 6 (May 2024), 86. https://doi.org/10.3390/robotics13060086

[21] Yanhao Yang, Nina L. Hecht, Yousef Salaman-Maclara, Nathan Justus, Zachary A. Thomas, Farhan Rozaidi, and Ross L. Hatton. 2025. Geometric Data-Driven Multi-Jet Locomotion Inspired by Salps. *arXiv preprint arXiv:2503.08817* (2025). https://doi.org/10.48550/arXiv.2503.08817 arXiv:2503.08817 [cs.RO]

[22] Zhiyuan Yang, Dongsheng Chen, David J. Levine, and Cynthia Sung. 2021. Origami-inspired robot that swims via jet propulsion. *IEEE Robotics and Automation Letters* 6, 4 (2021), 7145–7152. https://doi.org/10.1109/LRA.2021.3097757

[23] Zhiyuan Yang, Yipeng Zhang, Matthew Herbert, M. Ani Hsieh, and Cynthia Sung. 2025. Effect of Jet Coordination on Underwater Propulsion with the Multi-Robot SALP System. In *2025 IEEE 8th International Conference on Soft Robotics (RoboSoft)*. 1–8. https://doi.org/10.1109/RoboSoft63089.2025.11020967

[24] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. 2022. The Surprising Effectiveness of PPO in Cooperative, Multi-Agent Games. arXiv:2103.01955 [cs.LG] https://arxiv.org/abs/2103.01955