Methods for Autonomously Decomposing and Performing Long-horizon Sequential Decision Tasks

Shubham Pateria (G1701615L)

Supervisor: Assoc Prof. Quek Hiok Chai
Co-supervisor: Prof. Tan Ah-Hwee

School of Computer Science and Engineering
Nanyang Technological University

NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Outline of Presentation

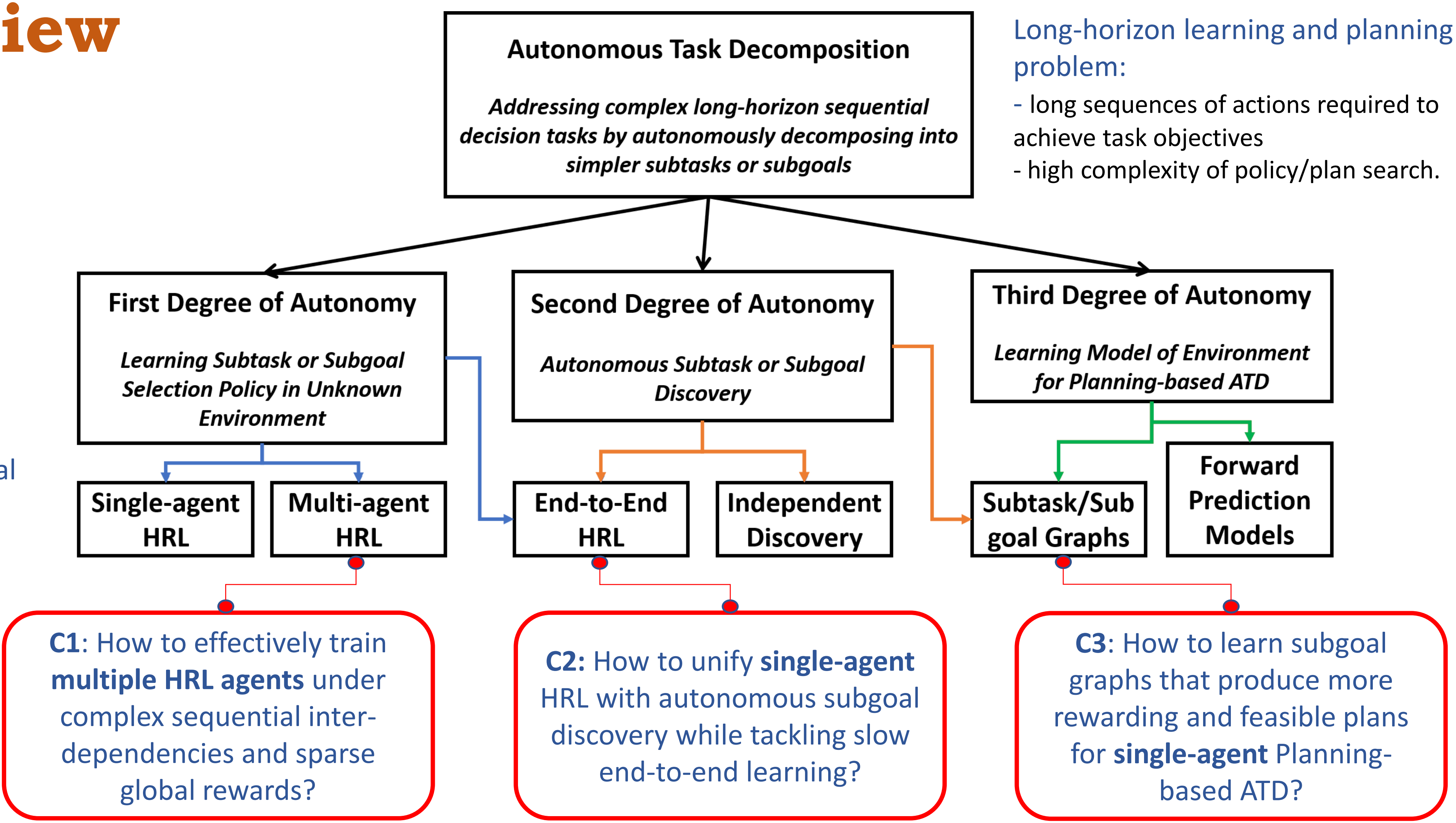
- Introduction
  - Overview of Autonomous Task Decomposition (ATD)
  - Research Challenges
  - My contributions
- Improving Coordinated Multi-agent HRL using Inter Subtask Empowerment Rewards
- Accelerating End-to-End HRL using Integrated Discovery of Salient Subgoals
- Learning Subgoal Graphs using Value-based Subgoal Discovery and Automatic Graph Pruning
- Conclusion
  - Summary of work
  - Major future work direction

Introduction

Overview

Long-horizon learning and planning problem:

- long sequences of actions required to achieve task objectives
- high complexity of policy/plan search.



Autonomous Task Decomposition

Addressing complex long-horizon sequential decision tasks by autonomously decomposing into simpler subtasks or subgoals

First Degree of Autonomy

Learning Subtask or Subgoal Selection Policy in Unknown Environment

Second Degree of Autonomy

Autonomous Subtask or Subgoal Discovery

Third Degree of Autonomy

Learning Model of Environment for Planning-based ATD

Single-agent HRL

Multi-agent HRL

End-to-End HRL

Independent Discovery

Subtask/Subgoal Graphs

Forward Prediction Models

HRL: Hierarchical Reinforcement Learning

Research Challenges

C1: How to effectively train **multiple HRL agents** under complex sequential inter-dependencies and sparse global rewards?

C2: How to unify **single-agent** HRL with autonomous subgoal discovery while tackling slow end-to-end learning?

C3: How to learn subgoal graphs that produce more rewarding and feasible plans for **single-agent** Planning-based ATD?

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Contributions

Challenge: How to effectively train multiple HRL agents for coordination under complex sequential inter-dependencies and sparse global rewards?

Method: Multi-agent HRL using Inter Subtask Empowerment Rewards (auxiliary rewards based on inter-dependencies).

- Better system performance than standard multi-agent HRL method that primarily relies on joint global rewards.

- A starting point for exploring principled approaches for training multiple HRL agents by learning their inter-dependencies and mutual effects on each other.

- Can be applied in multi-agent logistics, disaster response operations, warehouse management etc.

Contributions

Challenge: How to unify single-agent HRL with autonomous subgoal discovery while tackling slow end-to-end learning?

Method: Single-agent End-to-End HRL using Integrated Discovery of Salient Subgoals (explicit subgoal discovery)

- Better performance compared to state-of-the-art vanilla end-to-end HRL method that simply uses a large continuous subgoal space as the output space of the subgoal-selection policy.

- Significance as a starting point for researching and integrating various advantageous subgoal discovery heuristics into end-to-end HRL.

- Can be applied in goal-based navigation and robot manipulation tasks.

NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Contributions

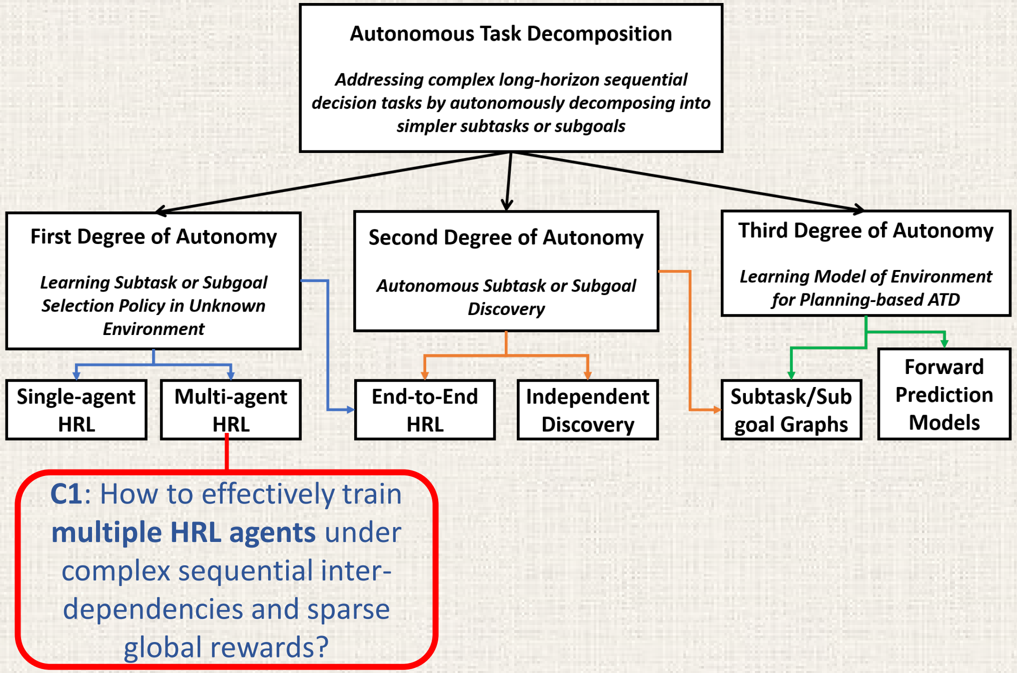
Challenge: How to learn subgoal graphs that produce more rewarding and feasible plans for single-agent Planning-based ATD?

Method: Value-based Subgoal Discovery and Automatic Graph Pruning to learn Subgoal Graphs for Single-agent Planning-based ATD

- Better performance compared to state-of-the-art subgoal graph methods that rely on simplistic ad-hoc heuristics for subgoal sampling and might also be prone to edge prediction errors.

- Can be used as a basis for developing reward-conforming heuristics for subgoal/state space abstraction and learning sparse models for higher-level planning using subgoals.

- Can be applied in goal-based navigation and robot manipulation tasks.



Autonomous Task Decomposition

Addressing complex long-horizon sequential decision tasks by autonomously decomposing into simpler subtasks or subgoals

First Degree of Autonomy

Learning Subtask or Subgoal Selection Policy in Unknown Environment

Second Degree of Autonomy

Autonomous Subtask or Subgoal Discovery

Third Degree of Autonomy

Learning Model of Environment for Planning-based ATD

C1: How to effectively train **multiple HRL agents** under complex sequential inter-dependencies and sparse global rewards?

Improving Coordinated Multi-agent HRL using Inter Subtask Empowerment Rewards

S. Pateria, B. Subagdja and A. Tan, "Multi-agent Reinforcement Learning in Spatial Domain Tasks using Inter Subtask Empowerment Rewards," 2019 **IEEE Symposium Series on Computational Intelligence** (SSCI), 2019

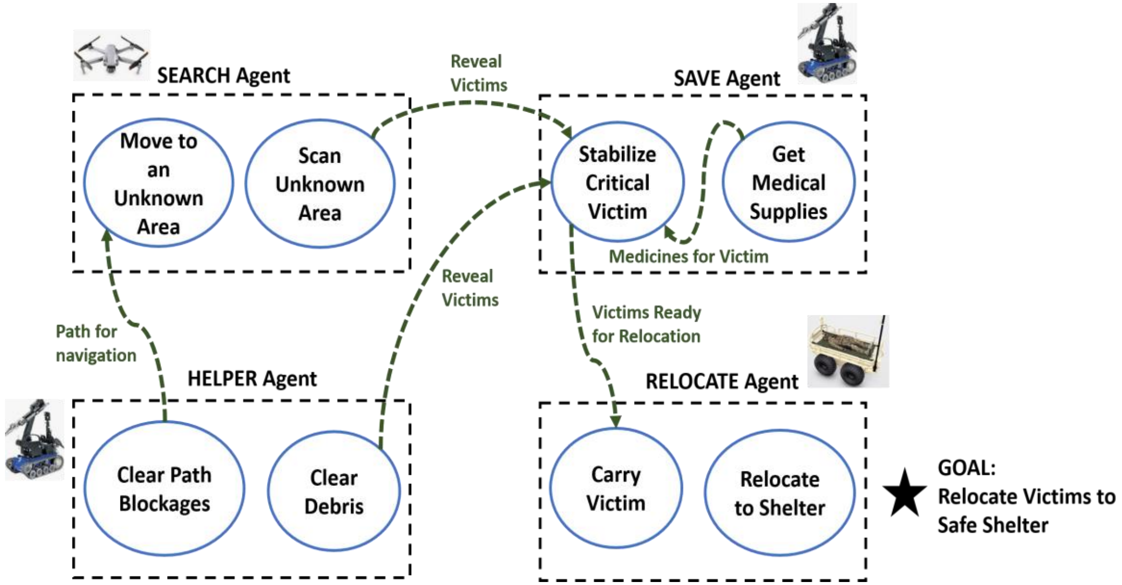
Scope

- Context: Decentralized HRL agents, centralized learning for coordination to perform a long-horizon joint task.

Issues and Challenges:

- Primary: Difficulty of learning coordination across different HRL agents due to complex sequential inter-dependencies and sparse global reward.

- Secondary: Fixed termination conditions or rules for various subtasks might result in sub-optimal performance due to a non-stationary environment consisting of multiple agents



ISEMO: Inter Subtask Empowerment based Multi-agent Options

Heterogenous + Homogenous HRL agents with **implicit** sequential inter-dependencies

Define a set of subtasks for each agent. Each subtask is defined as an **Option*** with the following components:

- **Handcrafted preconditions with respect to various global state features**

- Predefined primitive action policy

- **Learnable termination function**

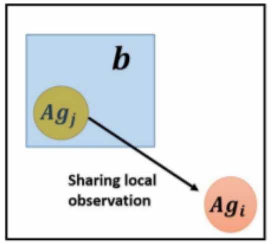
*Sutton, Richard S., Doina Precup, and Satinder Singh. "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning." *Artificial intelligence* 112.1-2 (1999).

During training process:

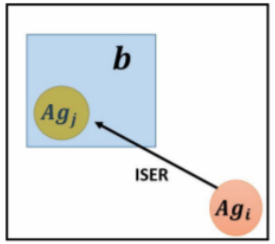
- Generate auxiliary rewards for individual agents.

- The auxiliary reward is called **Inter Subtask Empowerment Reward** (ISER)

- ISER is given to an agent if it enables the preconditions for subtask execution of another agent during the training process.



Agent Ag_j modifies certain environment feature in a region b



Agent Ag_i detects a precondition enabled due to change in the environment feature in region b and communicates ISER to the proximal agent Ag_j

$$\boldsymbol{R^j} = w_1 \times R_{ext} + w_2 \times \text{ISER}$$

During training process:

- Also learn the termination function of each subtask using the training data and **termination gradients**.

- Termination gradients adapted from single-agent **Option-Critic architecture*.**

*Bacon, Pierre-Luc, Jean Harb, and Doina Precup. "The option-critic architecture." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. No. 1. 2017.

Experiment Highlights

- Experiments performed using a custom (simulated) Search & Rescue task involving heterogeneous and homogeneous agents.

- Comparison with a standard multi-agent HRL method called Cooperative HRL (CoHRL). – CoHRL uses only the external global reward, does not account for inter-dependencies, and uses fixed termination rules/boundaries.

- Performance measure in terms of the number of victim deaths at the end of the task.

- The auxiliary reward ISER improves the performance even when added to CoHRL itself, **reducing the number of deaths by almost 50%** compared to CoHRL alone.

- ISEMO achieves higher performance through the combination of both the ISER and the adaptive termination functions (**almost 70% fewer deaths** compared to CoHRL alone).

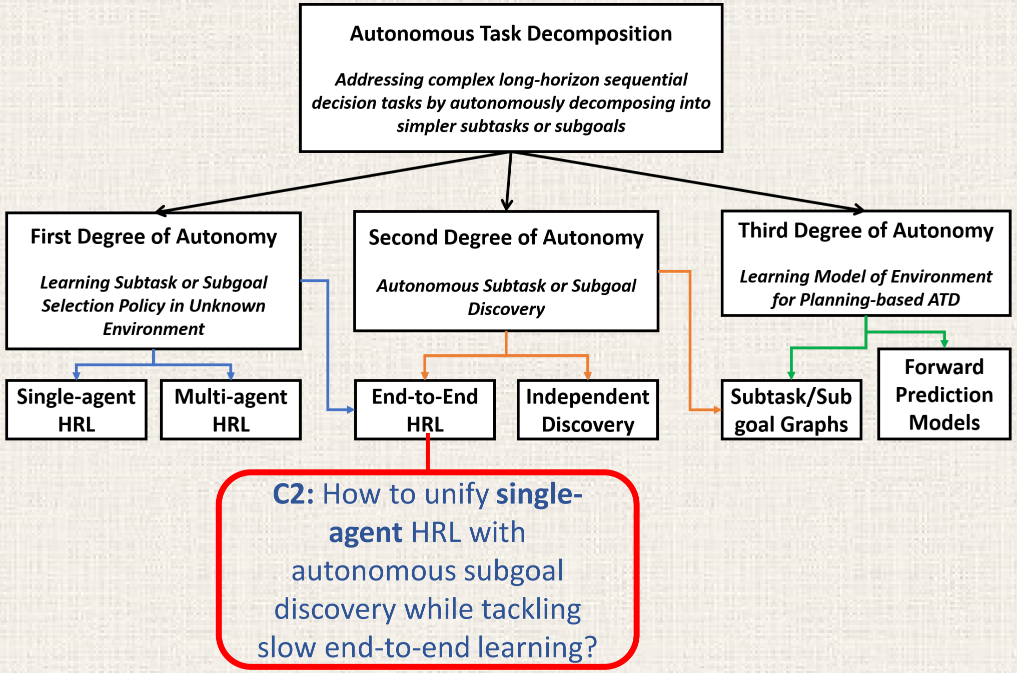
- With ISER, agents learn faster to select and execute those subtasks that enable the operation of other agents, resulting in better overall performance.

Assumptions and Limitations

- Agents must have global observability, ability to communicate, and *factored* state representation.

- **Subtasks are handcrafted**, not autonomously discovered.
  - Work on subtask/subgoal discovery in multi-agent HRL is nascent and highly challenging.

- The auxiliary reward ISER is derived using **handcrafted preconditions** (of subtasks).
  - How can the preconditions and ISER be learned?
    Discussed at the end of the presentation (Future Work).


Autonomous Task Decomposition
Addressing complex long-horizon sequential decision tasks by autonomously decomposing into simpler subtasks or subgoals
First Degree of Autonomy
Learning Subtask or Subgoal Selection Policy in Unknown Environment
Second Degree of Autonomy
Autonomous Subtask or Subgoal Discovery
Third Degree of Autonomy
Learning Model of Environment for Planning-based ATD
Single-agent HRL
Multi-agent HRL
End-to-End HRL
Independent Discovery
Subtask/Subgoal Graphs
Forward Prediction Models

C2: How to unify **single-agent** HRL with autonomous subgoal discovery while tackling slow end-to-end learning?

Accelerating End-to-End HRL using Integrated Discovery of Salient Subgoals

1. S. Pateria, B. Subagdja, A. -H. Tan and C. Quek, "End-to-End Hierarchical Reinforcement Learning With Integrated Subgoal Discovery," in **IEEE Transactions on Neural Networks and Learning Systems**.
2. Shubham Pateria, Budhitama Subagdja, and Ah Hwee Tan. 2020. Hierarchical Reinforcement Learning with Integrated Discovery of Salient Subgoals. In Proceedings of the 19th **International Conference on Autonomous Agents and Multi Agent Systems** (AAMAS '20).

Scope

- **Context**: Learning to reach long-horizon goals in single-agent navigation or control tasks, using end-to-end HRL.

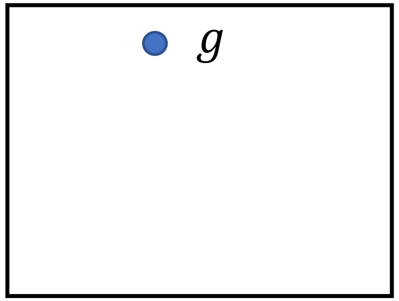
Issues and Challenges:

- Vanilla end-to-end HRL methods use a **large continuous subgoal space*** **as the output space** of subgoal selection policy so that the policy implicitly discovers useful subgoals.

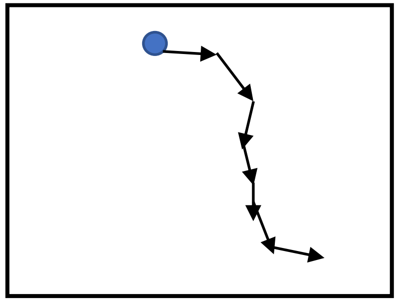
  This leads to the slow end-to-end learning issue.

Imagine a continuous 2D subgoal space

Continuous-output subgoal-selection policy: The output of the policy is initially random.

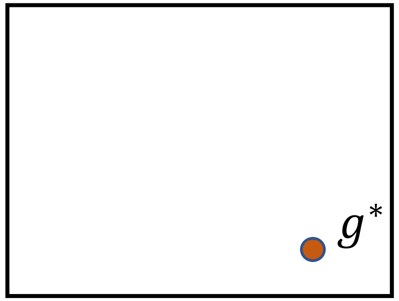


Over multiple iterations of exploration and training, the agent learns the Q-value function.



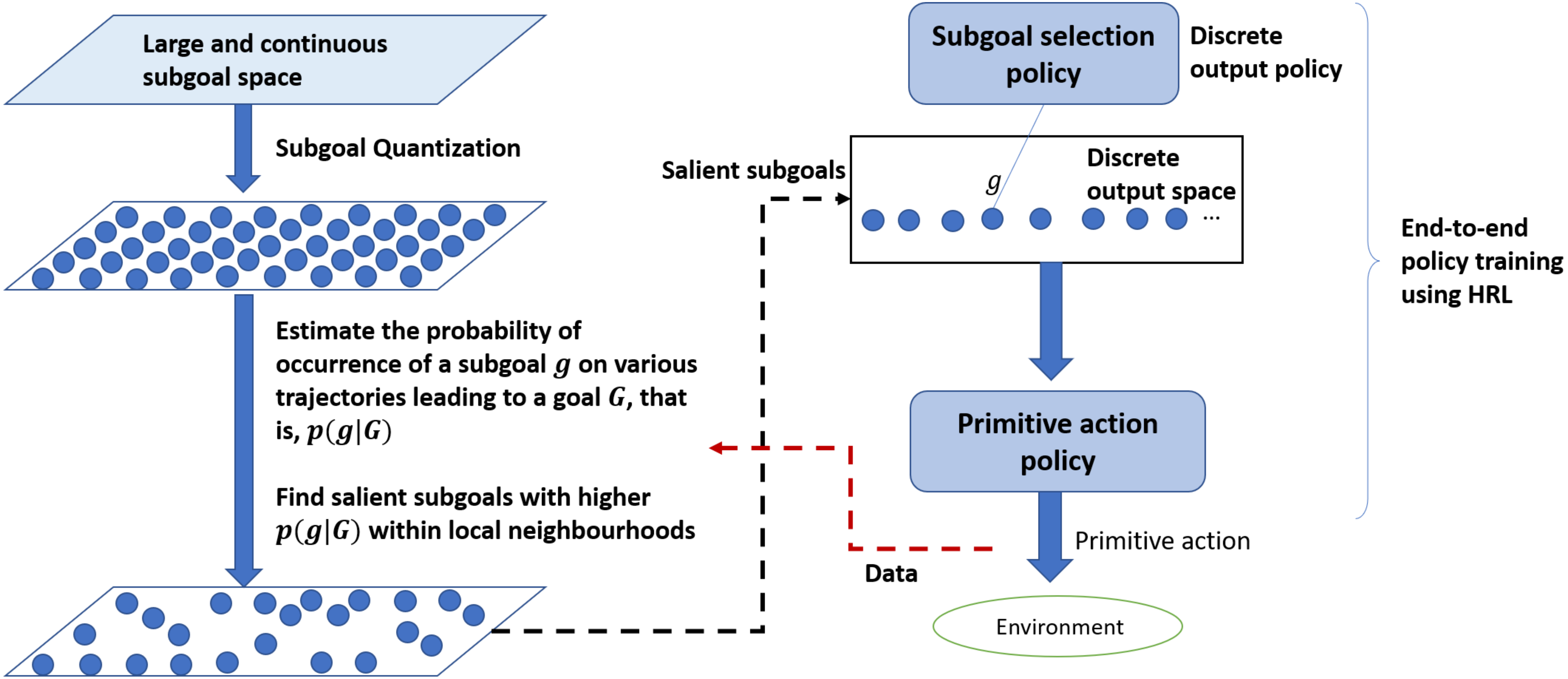
The gradients of the Q-value function are simultaneously used to learn the subgoal-selection policy

The output of the policy finally converges to the optimal subgoal(s)



*A discussion on parametric subtask spaces would be more complicated and out of scope for this work.

LIDOSS: End-to-End Hierarchical Reinforcement Learning with Integrated Discovery Of Salient Subgoals



Large and continuous subgoal space
Subgoal Quantization
Estimate the probability of occurrence of a subgoal g on various trajectories leading to a goal G, that is, $p(g|G)$
Find salient subgoals with higher $p(g|G)$ within local neighbourhoods
Salient subgoals
Subgoal selection policy
Discrete output policy
g
Discrete output space
…
Primitive action policy
Primitive action
Data
Environment
End-to-end policy training using HRL

NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

Experiment Highlights

- MuJoCo task domains. Continuous state and action spaces. Navigation to goals while avoiding obstacles.

- Comparison with Hierarchical Actor Critic (HAC): A state-of-the-art (vanilla) end-to-end HRL without explicit subgoal discovery. **Uses continuous subgoal space and parametric continuous-output subgoal-selection policy.**

- LIDOSS (discrete salient subgoals) outperforms HAC by achieving **10-40% higher success rates** of reaching the goals, across different experiments.
  - Both methods learn Q-value distribution at a similar pace. LIDOSS discovers subgoals faster than HAC's subgoal-selection policy learns & converges on useful subgoals. LIDOSS does not need gradient-based training for non-parametric ϵ-greedy subgoal-selection policy.

- **Is simple discretization/quantization of the subgoal space enough?**
  LIDOSS outperforms a quantization-only variant by achieving **6-15% higher success rates** of reaching the goals, across different experiments. Observations:
  - The quantization-only agent chooses a long and haphazard sequence of subgoals in a few episodes, possibly due to closely packed subgoals. This might affect exploration and learning.
  - Quantization might also include unreachable subgoals, such as those lying within walls.

Assumptions and Limitations

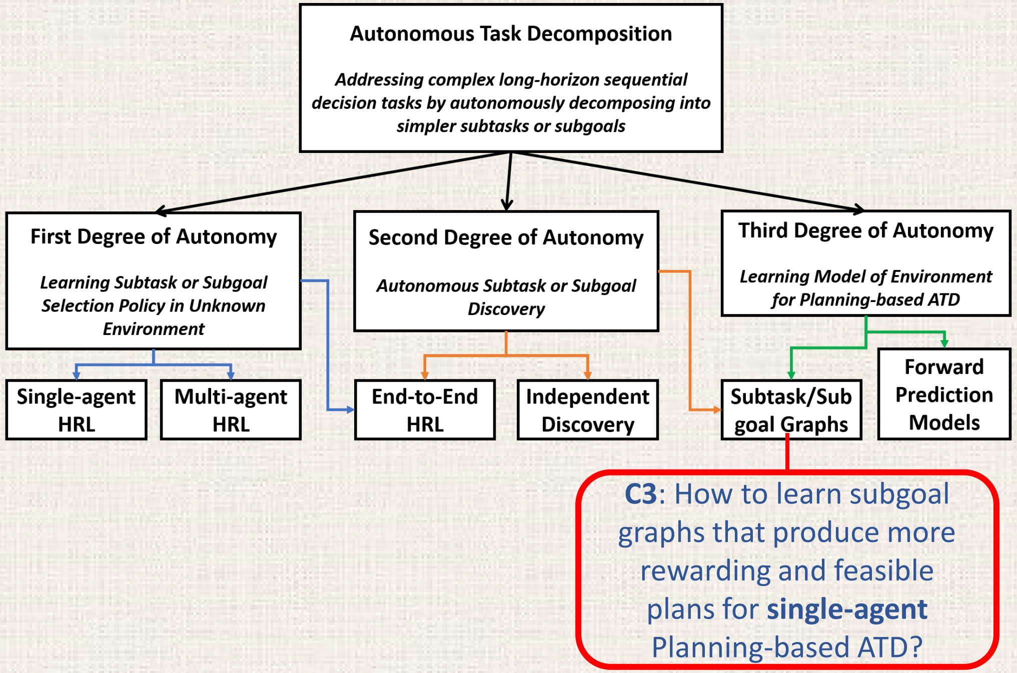
- LIDOSS requires **subgoal space quantization** for probability estimation, which might be **challenging in very high-dimensional subgoal spaces**.

- The subgoal discovery heuristic **does not take reward distribution into consideration**. It is suitable for tasks with terminal rewards generated only at the end of an episode.

- Pros and Cons of discrete salient subgoals compared to continuous subgoal space:

- Pros: Simplified output space of the subgoal selection policy.
  Use of discrete-output non-parametric policy which does not require slow gradient-based training (directly uses Q-values).

- Cons: **Restricts the ability to sample new subgoals from previously unseen/unexplored regions of the subgoal space**, a generalization that is possible with continuous subgoal space and continuous output policy (such as in HAC).
  Non-trivial to apply subgoal discovery in large state spaces.



Autonomous Task Decomposition

Addressing complex long-horizon sequential decision tasks by autonomously decomposing into simpler subtasks or subgoals

First Degree of Autonomy

Learning Subtask or Subgoal Selection Policy in Unknown Environment

Single-agent HRL

Multi-agent HRL

Second Degree of Autonomy

Autonomous Subtask or Subgoal Discovery

End-to-End HRL

Independent Discovery

Third Degree of Autonomy

Learning Model of Environment for Planning-based ATD

Subtask/Subgoal Graphs

Forward Prediction Models

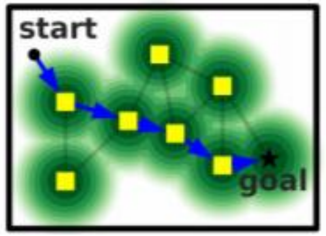
C3: How to learn subgoal graphs that produce more rewarding and feasible plans for **single-agent** Planning-based ATD?

Learning Subgoal Graphs using Value-based Subgoal Discovery and Graph Pruning

Full paper submitted to IEEE Transactions on Neural Networks and Learning Systems, on 05-Oct-2021. The manuscript is under review.

Scope

- Context: Learning to reach long-horizon goals in single-agent navigation or control tasks using subgoal graph-based goal decomposition and planning.



Issues and Challenges:

- The existing methods do not learn a subgoal graph that conforms to the reward distribution. Hence, they are not suitable for environments with non-uniform distribution of rewards across different regions of the state space

- The existing methods might plan infeasible sequences of subgoals (e.g. transition across obstacles) due to erroneously predicted connections (edges) between certain pairs of subgoals (nodes).

LSGVP: Learning Subgoal Graph using Value-based Subgoal Discovery and Automatic Pruning

Exploration and Training:

- Explore the environment. Save episodic data. Save observed states in a memory buffer.

- Learn **primitive action policy** π and **Q-value function** Q^π using shorter-horizon (pseudo) goals. π & Q **frozen after training**.

Inter-state **Distance Function (D)**

$$\mathcal{D}(s_i, s_j) = Q^+ - \max_\pi Q^\pi(s_i, a|s_j)\Big|_{a = \pi(a|s_i, s_j)}$$

Subgoal Discovery:

- Define a **subgoal dissimilarity measure** based on the distance function.
- Subgoals that are dissimilar beyond a threshold are discovered as salient subgoals.

Graph Construction: Construct a subgoal graph by adding directed edges between subgoals (nodes) if the inter-subgoal distance, estimated using D, is below a threshold.

Graph Pruning: Do back-and-forth traversals across the edges in the graph to test if predicted distances are correct, otherwise prune erroneous edges.

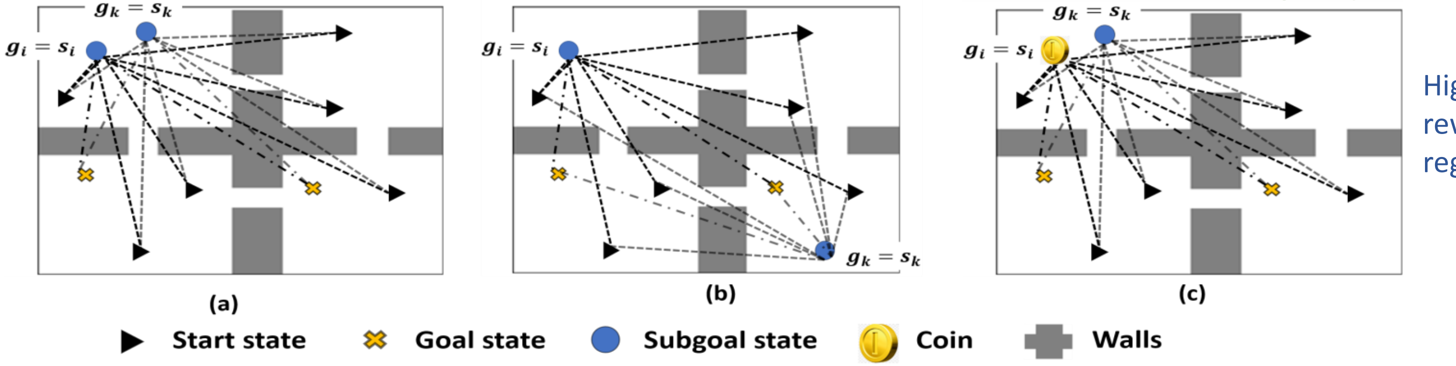
Testing:

- Dijkstra's shortest path planning to find a sequence of subgoals between a start state and a given long-horizon goal.
- Traverse to subgoals using π.

Low subgoal dissimilarity for all start, goal pairs

High subgoal dissimilarity for at least one start, goal pair

High subgoal dissimilarity for at least one start, goal pair



Higher positive reward at the Coin region

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Experiment Highlights

- Experiment 1: Testing the effect of value-based subgoal discovery.

  - Custom Coin Gather task. Continuous state and action spaces. Navigation to goal(s) while avoiding obstacles. Intermediate positive reward at the Coins.

  - Performance metrics: Average Positive Cumulative Rewards (APCR) and Average Success Rate (of reaching goals) across testing episodes.

  - Average Success Rates are found to be the same for all the compared methods.

- LSGVP achieves **~42% higher APCR** compared to a state-of-the-art subgoal graph-based ATD method called **Search on the Replay Buffer** (SoRB). SoRB performs **uniform subgoal sampling** and does not prune the graph**.** For this experiment, graph pruning was added to SoRB.

- LSGVP achieves **39-46% higher APCR** compared to two other subgoal sampling/discovery methods: **Farthest Point Sampling and Bottleneck discovery**. Graph pruning was added to both methods for this experiment.

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Experiment Highlights

- Experiment 2: Testing the effect of graph pruning.

- Comparison with state-of-the-art subgoal graph-based ATD methods:
  - **Search on the Replay Buffer** (SoRB), in two-dimensional navigation domains (Point Four Rooms and Maze).
  - **Semi Parametric Topological Memory** (SPTM), in higher-dimensional VizDoom navigation game.
  - Both methods do not prune the subgoal graphs.

- LSGVP achieves higher goal-reaching success rates (**percentage**) than SoRB and SPTM, for similar levels of subgoal graph sparsity

Point Four Rooms		**Point Maze**		**VizDoom**	
LSGVP **96.8 +- 3.6**	**SoRB** **52.3 +- 4.7**	**LSGVP** **69.2 +- 5.5**	**SoRB** **20.7 +- 4.3**	**LSGVP** **74.5 +- 9.6**	**SPTM** **52.4 +- 12.3**

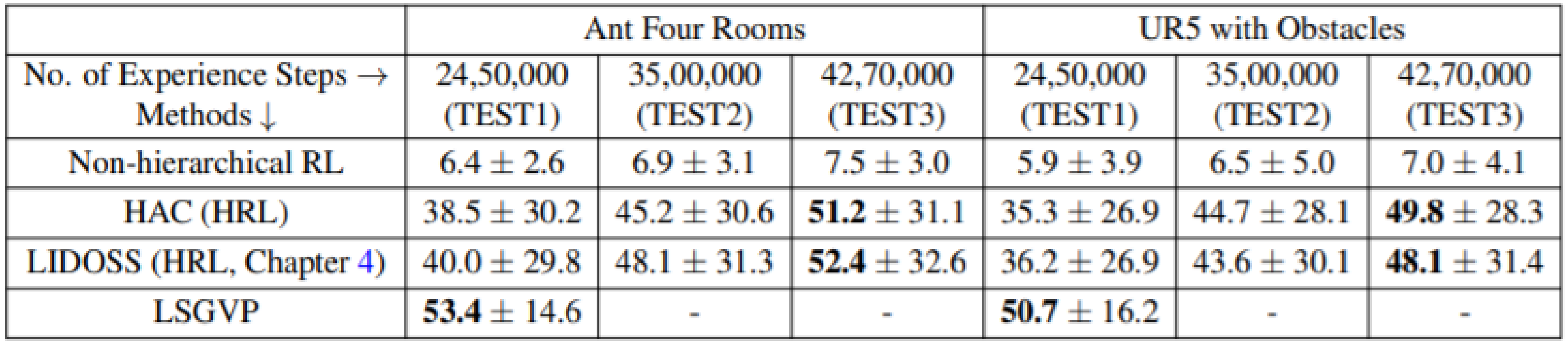
Average success rate across ten trials.

Experiment Highlights

- Experiment 3: Testing data efficiency compared to model-free HRL.
  - Comparison with LIDOSS and HAC, both with 2-level policy hierarchy.

  - MuJoCo continuous control domains, for navigation (Four Rooms) and robot arm-control (UR5).

- **LIDOSS and HAC require training with more data** (larger number of experience steps) to reach similar success rates as LSGVP.



	Ant Four Rooms			UR5 with Obstacles		
No. of Experience Steps → Methods ↓	24,50,000 (TEST1)	35,00,000 (TEST2)	42,70,000 (TEST3)	24,50,000 (TEST1)	35,00,000 (TEST2)	42,70,000 (TEST3)
Non-hierarchical RL	6.4 ± 2.6	6.9 ± 3.1	7.5 ± 3.0	5.9 ± 3.9	6.5 ± 5.0	7.0 ± 4.1
HAC (HRL)	38.5 ± 30.2	45.2 ± 30.6	**51.2** ± 31.1	35.3 ± 26.9	44.7 ± 28.1	**49.8** ± 28.3
LIDOSS (HRL, Chapter 4)	40.0 ± 29.8	48.1 ± 31.3	**52.4** ± 32.6	36.2 ± 26.9	43.6 ± 30.1	**48.1** ± 31.4
LSGVP	**53.4** ± 14.6	-	-	**50.7** ± 16.2	-	-

Average success rates, across fifty trials.

Assumptions and Limitations

- LSGVP works under the **assumption that the agent can completely explore the environment** during the initial training phase, before subgoal discovery, to learn a stationary subgoal graph.

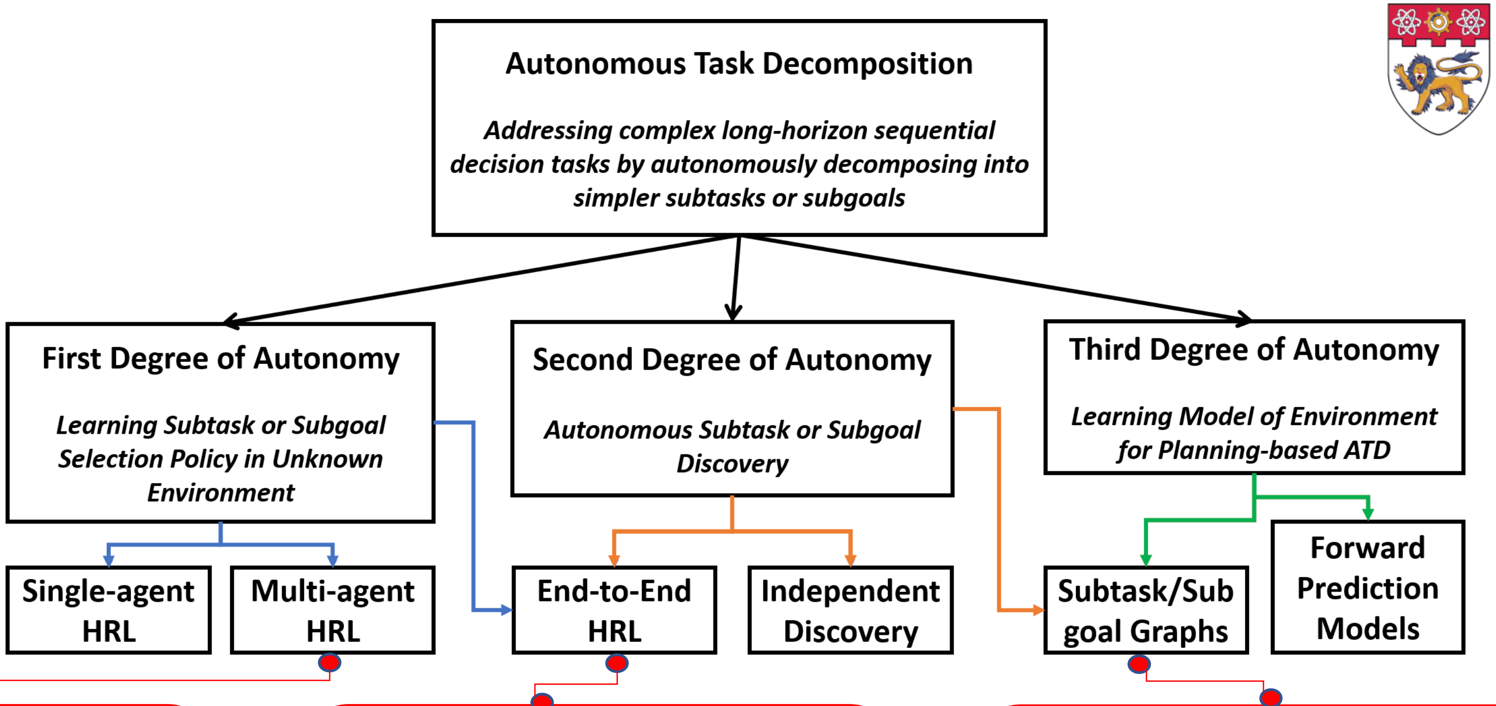
- Therefore, LSGVP is **not suitable for unbounded or infinite state spaces** which need to be continuously explored or for non-stationary environments.

- LSGVP requires a default reward of -1 at each action step, apart from any intermediate positive reward.

  This is to ensure that physical traversal distances can also be included in the distance function based on the Q-values.

Conclusion

Summary



Autonomous Task Decomposition

Addressing complex long-horizon sequential decision tasks by autonomously decomposing into simpler subtasks or subgoals

First Degree of Autonomy

Learning Subtask or Subgoal Selection Policy in Unknown Environment

Second Degree of Autonomy

Autonomous Subtask or Subgoal Discovery

Third Degree of Autonomy

Learning Model of Environment for Planning-based ATD

ATD Challenges

C1: How to effectively train **multiple HRL agents** under complex sequential inter-dependencies and sparse global rewards?

C2: How to unify **single-agent** HRL with autonomous subgoal discovery while tackling slow end-to-end learning?

C3: How to learn subgoal graphs that produce more rewarding and feasible plans for **single-agent** Planning-based ATD?

Introduced ATD Methods

ISEMO: improves multi-agent HRL under complex inter-dependencies among agents, **using auxiliary rewards** based on one agent enabling others' subtasks.

LIDOSS: accelerates end-to-end HRL by simplifying the output space of subgoal-selection policy, **using a probability-based subgoal discovery heuristic**.

LSGVP: introduces cumulative reward-based subgoal discovery and automatic pruning of erroneous connections in the subgoal graph, leading to **higher cumulative rewards and goal-reaching success rates.**

Major Future Work

Towards building general agents capable of solving diverse long-horizon tasks, by learning skill repositories and hierarchical transition models

- Humans learn, store, and compose skills to solve a variety of complex tasks.
- Learning and storing subtask-solving policies as lifelong skills.
- Incremental skill discovery; relation to curriculum learning and imitation learning.

- Learning hierarchical models of skill transitions, for long-horizon planning.

- Affordances: learning preconditions for skill initiation. In a multi-agent context, learned affordances can also enable learning auxiliary rewards such as ISER.

  - **Affordances** are clues in the environment that **indicate possibilities for action**
  - ISER can be derived by detecting a positive response from an affordance classifier corresponding to a subtask.
  - Learning affordance in RL: http://proceedings.mlr.press/v119/khetarpal20a/khetarpal20a.pdf
  - Affordances with respect to long-term outcomes: https://arxiv.org/pdf/2011.08424.pdf

Publications and Papers Under Review

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Publications:

- S. Pateria, B. Subagdja, A. -H. Tan and C. Quek, "End-to-End Hierarchical Reinforcement Learning With Integrated Subgoal Discovery," in **IEEE Transactions on Neural Networks and Learning Systems** (June 2021, Early Access), doi: 10.1109/TNNLS.2021.3087733. **(Impact Factor close to 9.0)**

- Shubham Pateria, Budhitama Subagdja, and Ah Hwee Tan. 2020. Hierarchical Reinforcement Learning with Integrated Discovery of Salient Subgoals. In Proceedings of the 19th **International Conference on Autonomous Agents and Multi Agent Systems** (AAMAS '20), Richland, SC, 1963–1965.

- Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. 2021. Hierarchical Reinforcement Learning: A Comprehensive Survey. **ACM Comput. Surv**. 54, 5, Article 109 (June 2021), 35 pages. **(Impact Factor > 10)**

- S. Pateria, B. Subagdja and A. Tan, "Multi-agent Reinforcement Learning in Spatial Domain Tasks using Inter Subtask Empowerment Rewards," 2019 **IEEE Symposium Series on Computational Intelligence** (SSCI), 2019, pp. 86-93, doi: 10.1109/SSCI44817.2019.9002777.

Under Review:

- "Value-based Subgoal Discovery and Path Planning for Reaching Long-Horizon Goals" is submitted as a full paper to **IEEE Transactions on Neural Networks and Learning Systems** and is **currently under review** (1st round). Submitted on 05-Oct-2021.

Thank You!

Reach me by email at: SHUBHAM007@e.ntu.edu.sg

The credit-assignment problem for a complex learning system (Minsky, 1961) is the problem of properly assigning credit or blame for overall outcomes to each of the learning system's internal decisions that contributed to those outcomes. In many cases the dependence of outcomes on internal decisions is mediated by a sequence of actions generated by the learning system. That is, internal decisions affect which actions are taken, and then the actions, not the internal decisions, directly influence outcomes. In these cases it is sometimes useful to decompose the credit-assignment problem into two subproblems: 1) the assignment of credit for outcomes to actions, and 2) the assignment of credit for actions to internal decisions. The first subproblem involves determining *when* the actions that deserve credit were taken, and the second involves assigning credit to the internal structure of actions. Accordingly, the first subproblem is called the *temporal credit-assignment problem*, and the second the *structural credit-assignment problem*.

Reward Shaping for Multi-agent Credit Assignment

Difference Rewards

$$D_i(z) = G(z) - G(z_{-i})$$

System performance

System performance without the action of agent i

Assigning auxiliary reward **ISER can be viewed as a form of credit assignment for the following special case**:

- Better system performance depends on enabling the agents to execute the subtasks required to achieve the joint task.

- If an agent enables the preconditions for subtask execution of other agent(s), it can potentially improve the system performance. Thus, ISER assignment is a form of credit assignment for eventually better system performance.

Made the mistake of not explaining this properly in the thesis.

Usage of this terminology (‘credit assignment’) can be avoided without diminishing the contribution and outcome of the work.