

Lab Project Proposal: Socially NEAT

Evolutionary Learning for Social Robot Navigation

3 MSc Students, 9 ECTS Each

1 Project Overview

This lab project investigates the combination of neuroevolution, imitation learning, and reinforcement learning for socially compliant mobile robot navigation. The project is designed for three MSc students working in parallel over one semester, with an expected workload of approximately 270 hours per student. The three students develop largely independent work packages with clearly defined standalone outcomes and subsequently integrate their components into a common social-navigation pipeline. All controller training, reinforcement learning, and evolutionary optimization are performed in simulation. Real-world experiments are restricted to deployment and evaluation of already trained controllers on a Husarion mobile robot.

The overall concept is

social-navigation teacher $\rightarrow$ imitation learning
 $\rightarrow$ lifetime reinforcement learning $\rightarrow$ evolutionary selection.

The central research question is whether evolution can produce compact neural controllers that acquire socially compliant behavior from a hand-designed teacher and subsequently improve that behavior through lifetime reinforcement learning. A secondary question is whether evolutionary selection can favor controller structures or initializations that are particularly easy to adapt during the lifetime of an agent.

2 Learning and Evolutionary Setting

A classical social-navigation controller, for example a Social Force Model, provides demonstrations of basic collision avoidance and socially compliant motion. A neural controller first learns this behavior using supervised imitation learning. The controller is then improved using reinforcement learning in simulation. Evolution acts across generations by selecting controller architectures, initializations, or other inherited parameters according to navigation performance after lifetime learning.

For a Baldwinian formulation, the learned lifetime parameters are not directly inherited. Instead, evolution favors genotypes that produce controllers which learn effectively during their lifetime. The resulting lifecycle can be written as

$$G_g \rightarrow \theta_0 \rightarrow \text{imitation learning} \rightarrow \text{reinforcement learning} \rightarrow F \rightarrow G_{g+1}.$$

A Lamarckian extension, in which selected learned parameters are inherited by the next generation, may be investigated if time permits.

3 Common System Interface

All work packages shall use a common simulator-independent observation and action interface. A possible observation representation is

$$o_t = \left[g_t, v_t, \omega_t, \{p_i, v_i\}_{i=1}^N, o_t^{\text{obs}} \right],$$

where g_t represents goal information, v_t and ω_t represent the robot state, p_i and v_i denote relative pedestrian states, and o_t^{obs} represents obstacle observations. The controller output shall remain compatible with the mobile robot interface,

$$a_t = [v_t, \omega_t].$$

This interface shall be used both in simulation and for deployment on the Husarion platform.

4 Work Package 1: Social Navigation Environment, Teacher, and Robot Interface

Responsible student: Student 1

Estimated workload: 270 hours

Objective

Develop the common social-navigation environment, implement an interpretable teacher controller, define the benchmark scenarios and evaluation metrics, and provide the simulation-to-robot interface required by all work packages.

Tasks

1. Set up a lightweight social-navigation simulation environment with a differential-drive mobile robot, static obstacles, moving pedestrians, and goal-conditioned navigation.
2. Implement a classical social-navigation teacher, preferably based on the Social Force Model or a comparable reactive method.
3. Implement goal attraction, pedestrian avoidance, static-obstacle avoidance, and velocity and acceleration constraints.
4. Define and document the common observation and action interface used by all work packages.
5. Generate demonstration trajectories from the teacher controller for supervised learning.
6. Implement reproducible evaluation scenarios including single-person crossing, head-on encounters, overtaking, corridor interaction, and multi-pedestrian scenes.
7. Implement common evaluation metrics including navigation success, collision rate, navigation time, path length, minimum pedestrian distance, personal-space violations, and control smoothness.
8. Implement the ROS 2 interface required to deploy trained controllers on the Husarion mobile robot.
9. Demonstrate the classical teacher or baseline controller on the real robot using a controlled navigation scenario.

Standalone Outcome

This work package shall produce a complete social-navigation benchmark, teacher controller, demonstration dataset, and robot interface independently of the evolutionary and reinforcement-learning components.

5 Work Package 2: Neuroevolution and Imitation Learning

Responsible student: Student 2

Estimated workload: 270 hours

Objective

Develop a compact NEAT-inspired evolutionary representation for neural navigation controllers and investigate whether evolved controller structures or initializations improve imitation learning and subsequent navigation performance.

Tasks

1. Implement or integrate a NEAT-style evolutionary representation for compact neural controllers.
2. Allow evolution of selected architectural properties such as hidden nodes, connectivity, initial weights, and optionally recurrent connections.
3. Implement mutation operators such as weight perturbation, addition or removal of connections, and addition of hidden nodes.
4. Implement supervised behavior cloning using the demonstration dataset generated in Work Package 1.
5. Compare fixed neural architectures with evolved architectures in terms of imitation accuracy, controller complexity, training speed, and navigation performance.
6. Introduce a complexity term or regularizer, for example

$$J_{\text{evo}} = J_{\text{nav}} - \lambda_N N_{\text{nodes}} - \lambda_E N_{\text{edges}}.$$

7. Visualize evolved controller graphs and structural changes across generations.
8. Deploy at least one trained evolved controller on the Husarion robot and demonstrate closed-loop navigation using the common observation and action interface.

Standalone Outcome

The student shall be able to compare a fixed-architecture imitation-learning controller against an evolved-architecture imitation-learning controller using a fixed demonstration dataset, without requiring the reinforcement-learning component.

6 Work Package 3: Lifetime Reinforcement Learning and Evolutionary Inheritance

Responsible student: Student 3

Estimated workload: 270 hours

Objective

Develop reinforcement-learning-based lifetime adaptation and investigate how navigation performance after lifetime learning can be incorporated into evolutionary selection.

Tasks

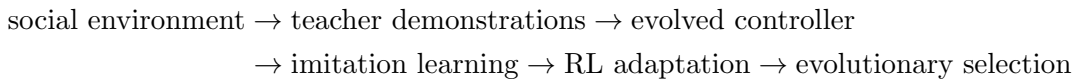
1. Implement reinforcement-learning fine-tuning of a social-navigation policy using a suitable algorithm such as PPO or SAC.
2. Define a reward function incorporating goal achievement, collision avoidance, social distance, progress, path efficiency, and smooth control.
3. Evaluate reinforcement learning with a fixed neural architecture before integration with neuroevolution.
4. Compare random initialization followed by reinforcement learning with imitation initialization followed by reinforcement learning.
5. Implement a Baldwinian evolutionary loop in which lifetime-learned parameters are not directly inherited but post-learning performance determines evolutionary fitness.
6. Evaluate learning speed, post-learning performance, and generalization across unseen social-navigation scenarios.
7. If time permits, implement a Lamarckian extension in which selected learned parameters are inherited by the next generation.
8. Deploy at least one reinforcement-learning-adapted controller on the Husarion robot and demonstrate closed-loop navigation in a controlled real-world scenario.

Standalone Outcome

The student shall be able to compare a fixed neural controller trained from scratch using reinforcement learning against a controller initialized from the social-navigation teacher and subsequently refined using reinforcement learning.

7 Integrated System

During the integration phase, the three work packages are combined into the complete pipeline



Work Package 1 provides the simulation environment, teacher, demonstrations, benchmark scenarios, evaluation metrics, and robot interface. Work Package 2 provides the evolutionary controller representation and imitation-learning component. Work Package 3 provides lifetime reinforcement learning and the evolutionary inheritance mechanism.

The integrated system shall be evaluated in simulation on both training and unseen social-navigation scenarios. All evolutionary optimization and reinforcement-learning training shall remain in simulation. Real-world experiments shall use frozen trained controllers and shall focus on demonstrating transfer to the Husarion platform.

8 Common Experimental Comparison

The final project report should, where feasible, compare the methods shown in Table 1.

Method	Evolution	Imitation	RL
Social Force Model	–	–	–
Fixed neural behavior cloning	–	Yes	–
Fixed neural BC+RL	–	Yes	Yes
NEAT-style evolution + BC	Yes	Yes	–
NEAT-style evolution + BC+RL	Yes	Yes	Yes
Baldwinian evolution + BC+RL	Yes	Yes	Yes

Table 1: Suggested comparison of navigation controllers.

9 Minimum, Expected, and Desired Outcomes

Minimum

- A common social-navigation simulation environment with reproducible scenarios.
- A classical teacher controller and generated demonstration dataset.
- A fixed-network behavior-cloning baseline.
- A reinforcement-learning fine-tuning baseline.
- A basic NEAT-style topology or architecture evolution mechanism.
- A Baldwinian evolutionary evaluation loop.
- A common set of evaluation metrics and a simulator-independent observation and action interface.
- Integration of the three software components in simulation.

Expected

- A complete integrated pipeline combining teacher demonstrations, imitation learning, neuroevolution, reinforcement learning, and Baldwinian evolutionary selection.
- Evaluation on multiple training and unseen social-navigation scenarios.
- Analysis of navigation performance, controller complexity, learning effort, and generalization.
- ROS 2 integration with the Husarion mobile robot.
- Real-world demonstration of selected trained controllers from each work package.
- At least one controlled social-interaction scenario on the real robot, such as a single-person crossing, head-on encounter, or corridor passing scenario.

Desired

- A full end-to-end demonstration of the integrated evolved and lifetime-adapted controller on the Husarion robot.
- Quantitative comparison of multiple frozen controllers on identical real-world scenarios.
- Evaluation with multiple pedestrians or more complex social interactions.
- Systematic analysis of the simulation-to-real performance gap.

- Comparison between Baldwinian and Lamarckian inheritance.
- Investigation of attention-based or recurrent pedestrian representations.
- Optional use of CMA-ES for continuous parameter optimization within the evolutionary framework.

10 Suggested Semester Plan

For a semester of approximately 15 working weeks, the project may be organized as follows.

Weeks	Activities
1–2	Literature review, simulator setup, common interfaces, benchmark definition, and software architecture.
3–7	Independent development of the three work packages.
8–10	Standalone experiments, debugging, and freezing of common interfaces.
11–13	Integration of evolution, imitation learning, reinforcement learning, and robot interface.
14–15	Common experiments, real-robot demonstrations, ablations, final report, and presentation.

The nominal workload is approximately 18 hours per student per week over 15 weeks.

11 Final Deliverables

Each student shall provide the implementation and documentation of their assigned work package, standalone experimental results, and an individual description of their contribution. The team shall additionally provide an integrated codebase, a common evaluation protocol, simulation results for the complete system, at least the expected real-robot demonstrations where feasible, and a joint final presentation demonstrating the integrated social-navigation system.