BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

A Prospectus for Ph.D. Dissertation

Minimizing the Intent-to-Reality Gap

By

BASSEL EL MABSOUT

Submitted in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy
In the Department of Computer Science

COMMITTEE MEMBERS:

DR. RENATO MANCUSO *First Reader*

DR. SABRINA NEUMAN

DR. KATE SAENKO

DR. BINGZHUO ZHONG

January 8, 2025

1 Abstract

This dissertation research addresses the intent-to-reality gap in robot learning—the challenge of translating high-level intentions into deployable policies. Practitioners face two key obstacles: expressing their intentions as learning objectives (the intent-to-behavior gap) and maintaining policy behavior when moving from simulation to reality (the sim-to-real gap). Current approaches lack principled structure for specifying objectives and rely on optimization targets that break down during transfer, leading to catastrophic forgetting when policies fail to preserve learned behaviors.

I introduce Expressive Reinforcement Learning as a subfield focused on minimizing the intent-to-behavior gap through structured objective specification and composition. Our early work demonstrated the promise of this design space by developing RE+AL, the first reinforcement learning framework to outperform classical PID controllers on quadrotor attitude control. Countless hours wrestling with brittle reward functions and watching otherwise-promising policies fail in spectacular ways taught us the need for more principled foundations. This led us to develop three complementary frameworks: an Algebraic Q-value Scalarization (AQS) method that minimizes the intent-to-behavior gap through intuitive objective composition (with 600% improvement in sample efficiency), anchor critics for preventing catastrophic forgetting during sim-to-real transfer through multi-objective Q-value optimization, and Conditioning for Action Policy Smoothness (CAPS) that reduces power consumption by 80% while maintaining performance. We further show that efficient deployment is possible through asymmetric actor-critic architectures, reducing model size by up to 99%. Through validation on increasingly complex robotic systems, we show how principled abstractions enable practitioners to directly express and compose their intentions, bridging the gap from intent to reality.

Contents

2 Problem Statement and Objectives

The fundamental challenge in robot learning lies in bridging the intent-to-reality gap—the disparity between a practitioner's intended robot behavior and what is achieved in reality. While reinforcement learning offers powerful tools for developing complex controllers, three critical limitations prevent its widespread adoption in real-world robotics:

First, practitioners struggle to translate their high-level intentions into precise learning objectives. Current approaches rely on brittle linear reward composition and manual tuning, making it difficult to express and balance multiple competing objectives. This intent-to-behavior gap leads to policies that either fail to learn desired behaviors or require prohibitive engineering effort to achieve them.

Second, even when policies perform well in simulation, they often fail to preserve learned behaviors when deployed to real systems. Traditional approaches to sim-to-real transfer treat it as a domain adaptation problem, leading to catastrophic forgetting where policies maintain performance on common scenarios but fail in critical edge cases. This challenge is particularly acute in robotics, where failures in rare but important scenarios can lead to system damage or safety violations.

Third, the computational and energy demands of learned policies often make deployment impractical on real robotic systems. Current methods produce controllers with excessive oscillations that waste energy and stress hardware, while standard neural architectures are unnecessarily complex for deployment. These efficiency challenges compound when attempting to deploy multiple policies or adapt to changing conditions.

We argue that addressing these challenges requires fundamentally rethinking how practitioners express and compose their intentions throughout the learning process. Instead of focusing solely on optimization algorithms, we must develop principled frameworks for:

1. Specifying behavioral objectives in ways that capture intended relationships and trade-offs
2. Preserving learned behaviors during transfer while enabling controlled adaptation
3. Ensuring efficient deployment through resource-aware policy design

Our research addresses these challenges through four interconnected objectives:

1. **Structured Objective Specification:** Develop a principled framework for composing behavioral objectives that enables practitioners to directly express intended relationships and trade-offs. This includes both the mathematical foundations for combining objectives and practical tools for specifying complex behaviors.
2. **Intention-Preserving Transfer:** Create methods that maintain critical behaviors during sim-to-real transfer by treating adaptation as multi-objective optimization over Q-values from both domains. This ensures policies can adapt to reality while preserving behaviors learned in simulation.

3. **Resource-Aware Control:** Design techniques for learning controllers that are efficient in both computation and energy usage. This spans from action smoothness for energy efficiency to architectural optimization for reduced inference cost.
4. **Practical Validation:** Demonstrate the effectiveness of these approaches through systematic evaluation on increasingly complex robotic systems, from quadrotor attitude control to multi-agent scenarios. This includes developing open-source tools and frameworks to enable broader adoption.

Through these objectives, we aim to transform how practitioners develop robotic systems by providing principled ways to express intentions, preserve behaviors, and ensure efficient deployment. The following sections detail our progress toward these goals and outline the remaining work needed to complete this vision.

3 Background and Literature Review

3.1 Definitions

Before diving into related work, we establish the foundational concepts and terminology used throughout this dissertation:

3.1.1 Core MDP Framework

3.1.1.1 Markov Decision Process (MDP): The standard formalization of sequential decision making, defined by a tuple (S, A, P, R, gamma), where S is the state space, A is the action space, P($s_{\{t+1\}}|s_t,a_t$) defines state transition probabilities, R($s_t,a_t,s_{\{t+1\}}$) specifies rewards, and γ is a discount factor. In robot learning, states typically represent sensor readings and physical configurations, while actions correspond to motor commands.

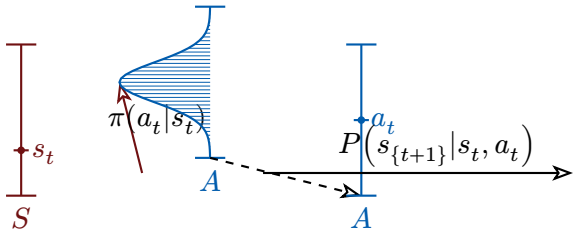

Figure 1: The Markov Decision Process showing how a state s_t in the state space S and action $a_t = \pi(s_t)$ determine the probability distribution $P(s_{\{t+1\}}|s_t, a_t)$ over next states.

3.1.1.2 Policy and Returns: A policy $\pi(a_t|s_t)$ defines the probability of taking action a_t in state s_t. The goal in RL is to find a policy that maximizes expected returns, defined as the discounted

sum of rewards: $E\left[\sum \gamma^t R\left(s_t, a_t, s_{\{t+1\}}\right)\right]$. The discount factor γ determines how much to prioritize immediate versus future rewards.

3.1.1.3 Value Functions and Q-Functions: A value function $V^{\pi(s_t)}$ represents the expected return when following policy π from state s_t. Similarly, a Q-function $Q^{\pi(s_t,a_t)}$ represents the expected return when taking action a_t in state s_t and then following π. These functions are central to our work in two key ways:

1. As a basis for objective composition in AQS, where normalized Q-values enable intuitive specification of behavioral trade-offs
2. As anchors for domain transfer, where simulation Q-values preserve critical behaviors during real-world adaptation

3.1.1.4 Expressive Reinforcement Learning: A subfield of reinforcement learning focused on minimizing the intent-to-behavior gap through structured objective specification and composition. This encompasses:

1. Principled approaches for expressing high-level intentions as learning objectives (e.g., CAPS for smooth control)
2. Methods for preserving behavioral semantics during optimization (e.g., Anchor Critics for sim-to-real transfer)
3. Frameworks for composing multiple objectives while maintaining their intended relationships (e.g., AQS for intuitive objective specification)

The expressiveness of an RL system is measured through three key dimensions: the speed and fidelity of translating practitioner intentions into learned behaviors, the robustness of preserving these behaviors during transfer and adaptation, and the efficiency with which the resulting policies can be deployed on real systems.

3.1.1.5 Behavioral Objectives: We use this term to encompass both the high-level intentions a practitioner wants to achieve and their formal expression as optimization targets. Our work demonstrates this through several key examples: smooth motor control achieved through temporal and spatial action similarity in CAPS, preservation of critical scenario performance through Q-value anchoring, resource efficiency through architectural optimization, and complex objective composition through algebraic Q-value operations in AQS. This structured approach distinguishes our objectives from simple reward functions, emphasizing their principled nature and composability.

3.1.2 Intent Gaps and Expressive RL

3.1.2.1 Intent-to-Behavior Gap: The discrepancy between a practitioner's desired policy behavior and what can be expressed through available specification mechanisms. This gap manifests in two ways: (1) the difficulty of translating complex behavioral intentions into precise mathematical objectives, and (2) the challenge of ensuring learned policies remain faithful to these

intentions during optimization. Minimizing this gap requires both structured specifications and principled composition methods.

3.1.2.2 Intent-to-Reality Gap: The total disparity between a practitioner's intended robot behavior and what is achieved in reality. This encompasses both the intent-to-behavior gap in specifying and learning the desired policy, and the sim-to-real gap that emerges when deploying policies trained in simulation to real systems. Bridging this gap requires not only expressive specifications but also methods for embedding high-level intentions about live adaptation to preserve behaviors during transfer.

3.1.3 Objectives and Policy Properties

3.1.3.1 Behavioral Objectives: We use this term to encompass both the high-level intentions a practitioner wants to achieve (e.g., smooth motor control, efficient resource usage) and their formal expression as optimization targets. This distinguishes them from simple reward functions, emphasizing their structured nature.

3.1.3.2 Objective Composition: The process of combining multiple behavioral objectives into a single learning target. This includes both the mathematical operation of combining value functions and the semantic preservation of each objective's intended role.

3.1.3.3 Policy Faithfulness: We define a policy as faithful when it reliably executes the practitioner's intended behaviors. This encompasses both performance on the primary task and adherence to auxiliary objectives like smoothness and efficiency.

3.1.4 Transfer and Adaptation

3.1.4.1 Environmental Transfer: The process of adapting a policy to work in a new environment, whether from simulation to reality or between different real-world conditions. This is broader than traditional sim-to-real transfer, encompassing any domain shift that requires policy adaptation.

3.1.5 Metrics and Validation

3.1.5.1 Policy Smoothness: A quantitative measure of a controller's action stability over time, capturing both temporal consistency and spatial coherence. Our work introduces principled metrics for measuring smoothness through frequency analysis of control signals and state-action mapping continuity, enabling objective comparison of different control approaches. This addresses a critical gap in the field where smoothness was previously assessed through ad-hoc or qualitative methods.

3.1.5.2 Behavioral Retention: The degree to which a policy maintains its learned behaviors when transferred to new domains or adapted to new conditions. We quantify this through multi-objective evaluation of Q-values across domains, particularly focusing on performance in critical

scenarios that may occur rarely but have high importance. This provides a more nuanced view than traditional average-case metrics.

3.1.5.3 Resource Efficiency: A multi-dimensional assessment of a policy's practical deployability, encompassing:

1. Computational efficiency measured through model size and inference time
2. Energy efficiency quantified through power consumption and control signal analysis
3. Hardware impact evaluated through actuator stress and system wear metrics

These metrics provide a systematic framework for evaluating and comparing different approaches to robot learning, moving beyond simple task performance to consider the full spectrum of deployment considerations.

These definitions emphasize our focus on structured objectives and intention preservation throughout the learning process. They will be referenced throughout our discussion of related work and our proposed approaches.

3.2 Related Work

3.2.1 Reward Specification in RL

4 Published Results

Our published work establishes three key foundations for Expressive Reinforcement Learning:

4.1 Foundations: How to Train Your Quadrotor

Our journey began with a fundamental challenge in robotics: achieving reliable low-level control of quadrotor attitude that could outperform classical PID controllers. While PID controllers were straightforward to use, they couldn't learn from experience or adapt to the environment. Previous attempts to apply reinforcement learning to this problem faced two critical issues: poor transfer from simulation to reality (with only one in dozens of trained agents proving controllable on real drones) and unstable, non-smooth control leading to excessive power consumption and hardware wear.

Through careful analysis, we identified a fundamental limitation in how objectives were being specified: the standard practice of linear reward composition required constant manual tuning based on observed behavior, leading to brittle policies that failed to capture true behavioral objectives. This led us to develop RE+AL, which introduced multiplicative reward composition

as an alternative that better preserved the intent of each objective. Combined with careful state representation and more realistic training signals, this approach provided a more principled foundation for policy learning.

The results were transformative. We achieved the first successful application of reinforcement learning to outperform classical PID controllers on quadrotor attitude control. The approach yielded a 10× reduction in training time, dropping from 9 hours to under 50 minutes. Additionally, we achieved significant improvements in control signal quality, reducing oscillations from 330Hz to 130Hz while maintaining low tracking errors of 4.2 deg/s.

This early work revealed a crucial insight that would shape my subsequent research: the gap between practitioner intent and realized behavior wasn't just about improving simulators—it was about how we specified objectives to the learning system. This realization led me to develop the concepts of intent-to-behavior and intent-to-reality gaps, and the principles of Expressive Reinforcement Learning.

4.2 Resource-Aware Control: CAPS

A critical but often overlooked challenge in deploying learned policies to real robots is the prevalence of oscillatory control responses. While deep reinforcement learning enables training control policies for complex dynamical systems, these policies often exhibit problematic behaviors detrimental to system integrity. These oscillations manifest as:

- Visible physical system oscillations affecting performance
- Increased power consumption and overheating from high-frequency control signals
- Potential hardware failures due to sustained stress

The challenge is particularly acute in continuous control domains, where controller responses can vary infinitely within acceptable output limits. While we can help shape behavior using reward engineering, we identified that certain objectives - like smoothness - can be directly encoded into the policy optimization process itself. This led to a fundamental insight: by explicitly specifying these objectives through policy regularization, we could work in conjunction with reward signals to shape the desired outcomes.

Our Conditioning for Action Policy Smoothness (CAPS) addressed this by directly shaping the neural network's mapping from states to actions through two complementary objectives, implemented as regularization terms:

1. **Temporal smoothness**: Actions should maintain similarity with previous actions to ensure smooth transitions in controller outputs over time
2. **Spatial smoothness**: Similar states should map to similar actions, providing robustness against measurement noise and modeling uncertainties

Key results include:

- 80% reduction in power consumption while maintaining task performance

- 96% improvement in policy smoothness
- Significant reduction in training time
- Successful flight-worthy controllers using simple reward structures

This work demonstrated a key principle: some objectives are better expressed directly through policy structure rather than through reward engineering. This insight would later contribute to our broader vision of how different types of objectives require different mechanisms of expression in reinforcement learning.

4.2.1 Impact on Real-World Deployment

The success of CAPS in achieving smooth control has led to its widespread adoption in real-world applications. Control smoothness proves crucial across multiple dimensions: it extends hardware longevity and reliability, enhances energy efficiency in battery-powered systems, enables safer human-robot interaction, and ensures more predictable behavior during deployment. This work exemplifies how well-structured specifications can simultaneously address multiple aspects of the intent-to-reality gap, from resource constraints to behavioral requirements.

4.3 Efficient Neural Architectures through Actor-Critic Asymmetry

While CAPS addressed energy efficiency through smoother actions, the computational cost of neural network inference remained a significant barrier to deployment. This led us to question a fundamental implicit assumption in actor-critic methods: that actors and critics should share similar neural network architectures.

A key insight emerged when we examined the different objectives these networks need to satisfy: the critic must develop rich representations to understand both system dynamics and reward structures, while the actor simply needs to learn a mapping that maximizes value. This suggested that the representational power needed to satisfy the actor's objectives might be much smaller than what's needed for the critic's objectives - yet the standard practice was to use identical architectures for both components.

Through systematic evaluation across multiple actor-critic algorithms and environments, we demonstrated that:

- Actors can be reduced by up to 99% in size while maintaining performance
- Average reduction of 77% in model parameters across tasks
- Successful validation across 4 popular actor-critic algorithms
- Consistent results across 9 different environments with varying dynamics

This work revealed a fundamental asymmetry in the representational requirements needed to satisfy actor versus critic objectives. Through careful empirical analysis, including a toy problem that demonstrated even a single-neuron actor could learn optimal policies given sufficient critic

capacity, we showed that matching network capacity to objective complexity could dramatically reduce deployment costs. This makes reinforcement learning significantly more practical for resource-constrained applications, particularly when multiple policies need to be deployed simultaneously.

5 Current Work

5.1 Algebraic Q-value Scalarization (AQS)

A fundamental limitation in reinforcement learning is the reliance on linear reward composition, which often proves brittle and unintuitive when designing desired policy behaviors. While practitioners can theoretically achieve any behavior through careful reward engineering, this process is often time-consuming, requires significant domain expertise, and can lead to suboptimal results. This challenge is particularly acute in real-world applications with multiple competing objectives.

To address these limitations, we developed Algebraic Q-value Scalarization (AQS), a novel domain-specific language that allows practitioners to express how different objectives should interact. Rather than focusing on reward engineering, AQS enables direct specification of when one objective should take priority over another based on their current satisfaction levels. Key innovations include:

1. Using the power-mean as a logical operator over normalized Q-values
2. Q-value scalarization instead of traditional reward scalarization
3. Q-value normalization for stable learning across objectives
4. Integration with a new DDPG-based algorithm called Balanced Policy Gradient (BPG)

This approach fundamentally changes how we express objectives in reinforcement learning. Rather than trying to encode complex behaviors through reward engineering, AQS provides an intuitive language for specifying how objectives should be prioritized. For example, if one objective is nearly satisfied while another is not, AQS can naturally express that the unsatisfied objective should take priority - similar to how an AND operator works in Boolean logic, but generalized to continuous values.

The results demonstrated comprehensive improvements across multiple dimensions. We achieved up to 600% improvement in sample efficiency compared to Soft Actor Critic, along with substantial reductions in policy variability. By providing a more principled way to express objective relationships, AQS enables practitioners to focus on what behaviors they want rather than how to engineer rewards to achieve those behaviors.

5.2 Anchor Critics for Robust Transfer

A key challenge in robot learning is maintaining policy behavior when transitioning from simulation to reality. Traditional approaches to sim-to-real transfer often lead to catastrophic forgetting, where policies maintain average performance but fail in critical edge cases. Our Anchor Critics method addresses this by treating transfer as multi-objective optimization over Q-values from both domains.

Key innovations include:

- Using simulation Q-values as anchors for performance metrics
- Multi-objective optimization to balance adaptation and retention
- Integration with on-board inference for real-time control

Initial results demonstrate:

- Near 50% reduction in power consumption
- Successful behavior retention in both sim-to-sim and sim-to-real scenarios
- Development of open-source firmware for on-board inference and real-time control

6 Proposed Methodology

Our proposed work focuses on three key areas that will advance the field of Expressive Reinforcement Learning:

6.1 Structured Objective Specification (AQS)

6.1.1 Preliminary Results

- Novel domain-specific language design
- Power-mean operators and Q-value normalization
- 600% improvement in sample efficiency

6.1.2 Future Development

- Formal semantics and verification
- Safety constraint integration
- Theoretical guarantees

6.2 Intention-Preserving Transfer (Anchor Critics)

6.2.1 Preliminary Results

- Multi-objective Q-value optimization
- Preventing catastrophic forgetting
- Systems Implementation
  - SWANNFlight firmware development
  - On-board inference optimization

6.2.2 Future Development

- System identification integration
- Real-time adaptation strategies
- Hardware-in-the-loop validation

6.3 Objective Composition and Validation

6.3.1 Theoretical Integration

- Unifying Q-value based objectives
  - Compositional properties
  - Theoretical guarantees

6.3.2 Validation Methods

- Behavioral verification metrics
  - Trade-off preservation analysis
  - Compositional property validation

This methodology builds on our preliminary results while advancing both the theoretical foundations and practical applications of expressive reinforcement learning.

7 Timeline and Milestones

The completion of this dissertation involves finalizing two key publications and writing the dissertation itself:

7.1 January 2024

- Submit AQS paper to ICML 2024 (January 30)
  - Complete comparative analysis with recent MORL methods
  - Expand benchmark evaluations
  - Prepare manuscript and code release

7.2 February - March 2024

- Begin dissertation writing
  - Integrate existing published work (RE+AL, CAPS, Actor-Critic)
  - Develop unified theoretical framework
  - Write core chapters

7.3 February 2025

- Submit Anchor Critics paper
  - Refine related works section and introduce the definition of the objectives framework as a contribution
  - Clean up SWANNFlight firmware documentation
  - Prepare manuscript and supplementary materials

7.4 March - April 2025

- Complete dissertation
  - Incorporate all paper contributions
  - Submit dissertation draft to committee
  - Dissertation defense (April)

This timeline builds on our already published work (RE+AL, CAPS, and Actor-Critic papers) while ensuring thorough completion of the remaining publications and dissertation writing.

8 Bibliography