Biologically-Inspired Distributed Control Algorithms

Problem Statement: Division-of-labor algorithms break up large problems into smaller, more manageable components that can be solved in parallel. These techniques are especially practical for solving distributed control problems, such as coordinating robot swarms or sensor networks [5]. Indeed, solutions that employ many simple agents are often more effective and more robust to noise than monolithic controllers. However, designing distributed control algorithms remains difficult: Humans are poor at predicting emergent, group-level behavior, as minor alterations in how agents interact can have cascading effects at the group level [5]. Worse, these distributed control problems often involve uncertain or noisy environments making a centralized controller unreliable and further reducing human intuition for distributed algorithm design.

Elegant control algorithms that divide labor among distributed agents are common in nature. We need look no further than the algorithms encoded in the DNA of multicellular organisms or colonies of eusocial insects; in both cases, agents use noisy sensory information and localized signals to coordinate specialized roles. **I propose to explore how evolution produces effective forms of division and to design distributed control algorithms using these principles.** I will add new approaches to our toolbox for engineering distributed control algorithms.

Background: Dynamic control algorithms must allow agents to alter their behavior in response to uncertain conditions; in biology, this ability is called *phenotypic plasticity* [1]. *Distributed* control algorithms require more complex forms of plasticity to ensure each individual contributes to group-level problem solving. While these topics are challenging to study in biological systems due to the long time scales at which evolution operates, computational evolving systems have allowed for substantial progress in understanding the evolution of both plasticity (*e.g.*, [4]) and division of labor (*e.g.*, [2]). Digital organisms in these experiments are self-replicating computer programs in a Turing-complete language. They allow powerful studies of evolution, and the resulting programs can be analyzed, decomposed, and used to advance computational theory [3].

While plasticity and division of labor go hand-in-hand, previous studies have initialized groups with non-plastic agents, forcing role assignment mechanisms to evolve from scratch [2].

Hypothesis I: Evolution will co-opt ancestral plasticity mechanisms as building blocks toward more complex coordination in distributed control algorithms.

Hypothesis II: Evolutionary processes can be guided to produce substantially more effective control algorithms with the careful choice of ancestral state, customized to the problem type.

Approach: I will study obligate groups of agents evolving under conditions known to promote division of labor [2]. Groups can replicate by performing collective tasks, replacing a random competing group. I will vary the types and complexity of the initial groups as well as the tasks they must perform. For example, initial plasticity mechanisms will include:

Signal-based plasticity relies on exogenous signals to determine the appropriate task, such as attempting to metabolize a resource only when it is present in the environment. I expect this form of plasticity to produce signal-based coordination, common in developmental biology.

Location-based plasticity uses an agent's position to resolve the appropriate task (*e.g.*, grooming in a den, while foraging in the open). Resulting division-of-labor strategies may involve agents moving around, performing local tasks that will contribute to the overall goal.

Probabilistic plasticity employs a form of bet-hedging, randomly picking the next task and counting on positive results on average; commonly used to desynchronize from competitors or predators. I expect this technique to be valuable for division of labor when a range of tasks must be performed, but communication and environmental signals are unreliable.

I will evolve these groups of agents in a set of environmental challenges that will require organisms to either perform mathematical functions (*e.g.*, Boolean XOR), navigate a landscape (*e.g.*, foraging), or form patterns (*e.g.*, stripes or a segmented body). Some environments will require groups to solve many tasks, others will be a single task that can be solved efficiently if decomposed and divided among the group. In all cases, I will stress the group to promote robust algorithms: I will impose noise on sensor inputs, trigger environmental changes, or kill agents.

I will use four metrics to characterize evolved algorithms:

Quality - How close is the result of this algorithm to an optimal (or the best known) solution?

Efficiency - How does the run time of the algorithm scale as problem sizes increase?

Scalability - How well does each algorithm perform as I increase the number of agents?

Robustness - How well does an algorithm handle unexpected environmental changes, agent knock-outs, and additional environmental noise?

Due to tradeoffs in these metrics, all algorithms along the pareto front will warrant analyses. In each case, I will step through the evolved code, classify phenotypes, and perform multi-scale analyses to identify how individual, neighborhood, and global behaviors interact. Instruction-level knock-out analyses will help me link code to function, as will information-theoretic analyses on sensor inputs and communications. Finally, I will create simple simulations of individual behaviors to tease apart their role in the global dynamics.

Expected Outcomes: I expect evolution to co-opt pre-existing plasticity for distributed controllers once groups are formed, driving the type of distributed control algorithm that evolves. As such, I expect a wide range of control algorithms to evolve. Additionally, I expect different pre-existing mechanisms for plasticity to succeed or fail at different problem types, revealing the merits and faults of various distributed controller design practices for certain types of problems.

For some environmental challenges, I may find that groups do not evolve that can solve the problem. If so, the problem environment may be poorly defined or I may be failing to properly reward intermediate solutions. Identifying this pitfall is a matter of systematically simplifying the problem until it can be solved, and then slowly scaling it back up until the problem is clear.

Broader Impacts: My research will contribute to both computer science and evolutionary biology. As such, I will publish my findings in peer-reviewed journals and present at conferences in both fields, facilitating further collaborations between the fields. In collaboration with the Digital Evolution Lab, I am contributing to the development of the next generation of Avida, a widely used open source computational evolution platform that I will use to carry out my proposed research. In addition to adding new features (*e.g.* more advanced, physics-based environments), the platform will use asm.js to run efficiently on the web, increasing its accessibility to other scientists as well as the general public. In addition to sharing my research findings through journal articles and talks, I will publish web pages to automatically load my experiments and data visualizations with explanations for the general public, allowing anyone to explore distributed control problems.

References: **(1)** Ghalambor *et al.* Behavior as Phenotypic Plasticity. *Evo. Behav. Ecology.* **(2)** Goldsby *et al.* Task-switching costs promote the evolution of division of labor and shifts in individuality. *Proc. Nat. Acad. Sci.* **(3)** Knoester *et al.* (2008) Cooperative network construction using digital germlines, *Proc. GECCO 2008.* **(4)** Lalejini & Ofria (2016) The evolutionary origins of phenotypic plasticity. *Proc. ALife 2016.* **(5)** Ye *et al.* (2016) A survey of self-organisation mechanisms in multi-agent systems. *IEEE Trans. Systems.*