The Role of Conceptual Structure in Designing Cellular Automata to Perform Collective Computation

Manuel Marques-Pita[1,2,3], Melanie Mitchell[3], and Luis M. Rocha[1,2]

[1] Indiana University
[2] Instituto Gulbenkian de Ciência
[3] Portland State University

Abstract. The notion of conceptual structure in CA rules that perform the density classification task (DCT) was introduced by [1]. Here we investigate the role of *process-symmetry* in CAs that solve the DCT, in particular the idea of *conceptual similarity*, which defines a novel search space for CA rules. We report on two new process-symmetric one-dimensional rules for the DCT which have the highest "balanced" performance observed to date on this task, as well as the highest-performing CA known to perform the DCT in two dimensions. Finally, we investigate the more general problem of assessing how different learning strategies (based on evolution and coevolution, with and without spatial distribution), previously compared by [2], are suited to exploit conceptual structure in learning CAs to perform collective computation.

1 Introduction

The study of computation in cellular automata (CAs) and related cellular architectures has lately garnered renewed interest due to advances in the related fields of reconfigurable hardware, sensor networks, and molecular-scale computing systems. In particular, cellular array architectures are thought to be appropriate for constructing physical devices such as field configurable gate arrays for electronics, networks of robots for environmental sensing and nano-devices embedded in interconnect fabric used for fault tolerant nanoscale computing [3]. A current stumbling block for CA computing is the difficulty of programming CAs to perform desired computations, due to the decentralized architectures and nonlinear behavior of these systems. One approach is to use genetic algorithms or other evolutionary computation methods to evolve cellular automata transition rules that will perform desired computations. However, this approach has problems of scaling, due to the large search spaces for non-elementary CAs—those with larger than nearest-neighbor cell communication or with multiple states per cell.

In this paper we describe our investigation of reducing the dimensionality of these search spaces by using automatically-discovered *conceptual structures* of rule tables that are common to CAs likely to be successful for a particular computational task. We show that for one well-studied task—two-state density

C.S. Calude et al. (Eds.): UC 2008, LNCS 5204, pp. 146–163, 2008.

classification—a particular conceptual structure of CA rule tables that we call *degree of process symmetry* is correlated with success on the task, and is implicitly increased by genetic algorithms evolving CAs for this task. We also show that process symmetry provides a search space of significantly reduced dimensionality, in which a genetic algorithm can more easily discover high-performing one- and two-dimensional CAs for this task.

2 Cellular Automata

A cellular automaton (CA) consists of a regular lattice of N cells. Each cell is in one of k allowed states at a given time t. Let $\omega \in \{0, 1, ..., k-1\}$ denote a possible state of a cell. Let state $\omega = 0$ be referred to as the *quiescent* state, and any other state as an *active* state. Each cell is connected to a number of neighbors. Let a local neighborhood configuration (LNC) be denoted by μ, and its size by n. For each LNC in a CA an output state is assigned to each cell. This defines a CA rule string, ϕ, the size of which is k^n. In binary CAs, in which only two states are allowed($k = 2$), it is possible to classify individual cell state-updates in three categories: (1) *preservations,* where a cell does not change its state in the next time instance $t+1$; (2) *generations,* state-updates in which the cell goes from the quiescent to the active state; and (3) *annihilations,* state-updates where the cell goes from the active to the quiescent state. The execution of a CA for a number M of discrete time steps, starting with a given initial configuration (IC) of states, is represented as the set Θ containing $M+1$ lattice state configurations.

2.1 The Density Classification Task

The Density Classification Task (DCT) is a widely cited example of collective computation in cellular automata. The goal is to find a one-dimensional binary CA rule (with periodic boundary conditions) that can classify the majority state in a given, random IC (with odd number of cells). If the majority of cells in the IC are in the quiescent state, after a number of time steps M, the lattice should converge to a homogeneous state where every cell is in the quiescent state, with analogous behavior for an IC with a majority of active cells. Devising CA rules that perform this task is not trivial, because cells in a CA lattice update their states based only on local neighborhood information. However, in this particular task, it is required that information be transferred across time and space in order to achieve a correct global classification. The definition of the DCT used in our studies is the same as the one given by [4].

We define the *performance* $\mathcal{P}_N^K(\phi)$ the fraction of K initial configurations on a N-cell lattice that produce correct classifications (all quiescent for a majority of quiescent states in the IC; all active for a majority of active states in the IC).

Nine of the cellular automata rules with highest performance on the DCT were analyzed to determine whether there is *conceptual structure* not explicit in them, and if so, to investigate the possible *conceptual similarity* among them using a cognitively inspired mechanism (Aitana) [1]. Three of these rules were produced