Contemporary Approaches to Artificial General Intelligence

Cassio Pennachin and Ben Goertzel

AGIRI – Artificial General Intelligence Research Institute
1405 Bernerd Place, Rockville, MD 20851, USA
cassio@agiri.org, ben@agiri.org - http://www.agiri.org

1 A Brief History of AGI

The vast bulk of the AI field today is concerned with what might be called "narrow AI" – creating programs that demonstrate intelligence in one or another specialized area, such as chess-playing, medical diagnosis, automobile-driving, algebraic calculation or mathematical theorem-proving. Some of these narrow AI programs are extremely successful at what they do. The AI projects discussed in this book, however, are quite different: they are explicitly aimed at artificial *general* intelligence, at the construction of a software program that can solve a variety of complex problems in a variety of different domains, and that controls itself autonomously, with its own thoughts, worries, feelings, strengths, weaknesses and predispositions.

Artificial General Intelligence (AGI) was the original focus of the AI field, but due to the demonstrated difficulty of the problem, not many AI researchers are directly concerned with it anymore. Work on AGI has gotten a bit of a bad reputation, as if creating digital general intelligence were analogous to building a perpetual motion machine. Yet, while the latter is strongly implied to be impossible by well-established physical laws, AGI appears by all known science to be quite possible. Like nanotechnology, it is "merely an engineering problem", though certainly a very difficult one.

The presupposition of much of the contemporary work on "narrow AI" is that solving narrowly defined subproblems, in isolation, contributes significantly toward solving the overall problem of creating real AI. While this is of course true to a certain extent, both cognitive theory and practical experience suggest that it is not so true as is commonly believed. In many cases, the best approach to implementing an aspect of mind in isolation is very different from the best way to implement this same aspect of mind in the framework of an integrated AGI-oriented software system.

The chapters of this book present a series of approaches to AGI. None of these approaches has been terribly successful yet, in AGI terms, although several of them have demonstrated practical value in various specialized domains (narrow-AI style). Most of the projects described are at an early stage of engineering development, and some are still in the design phase. Our aim is not to present AGI as a mature field of computer science – that would be

impossible, for it is not. Our goal is rather to depict some of the more exciting ideas driving the AGI field today, as it emerges from infancy into early childhood.

In this introduction, we will briefly overview the AGI approaches taken in the following chapters, and we will also discuss some other historical and contemporary AI approaches not extensively discussed in the remainder of the book.

1.1 Some Historical AGI-Related Projects

Generally speaking, most approaches to AI may be divided into broad categories such as:

- symbolic;
- symbolic and probability- or uncertainty-focused;
- neural net-based;
- evolutionary;
- artificial life;
- program search based;
- embedded;
- integrative.

This breakdown works for AGI-related efforts as well as for purely narrow-AI-oriented efforts. Here we will use it to structure a brief overview of the AGI field. Clearly, there have been many more AGI-related projects than we will mention here. Our aim is not to give a comprehensive survey, but rather to present what we believe to be some of the most important ideas and themes in the AGI field overall, so as to place the papers in this volume in their proper context.

The majority of ambitious AGI-oriented projects undertaken to date have been in the symbolic-AI paradigm. One famous such project was the General Problem Solver [42], which used heuristic search to solve problems. GPS did succeed in solving some simple problems like the Towers of Hanoi and *crypto-arithmetic*,[1] but these are not really general problems – there is no learning involved. GPS worked by taking a general goal – like solving a puzzle – and breaking it down into subgoals. It then attempted to solve the subgoals, breaking them down further into even smaller pieces if necessary, until the subgoals were small enough to be addressed directly by simple heuristics. While this basic algorithm is probably necessary in planning and goal satisfaction for a mind, the rigidity adopted by GPS limits the kinds of problems one can successfully cope with.

[1] Crypto-arithmentic problems are puzzles like `DONALD + GERALD = ROBERT`. To solve such a problem, assign a number to each letter so that the equation comes out correctly.