Desiderata for Incorporating Coordination Design into a Designer's Desktop for DAI (Extended Abstract)

Raphael Malyankar*
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85287-5406, USA.
Email: rmm@acm.org

This paper discusses certain requirements for including facilities for coordination design in a designer's desktop for intelligent agent systems (IAS). The term 'coordination design' is used to mean the selection of suitable agent interaction techniques from a technique toolbox. The aim is to allow optimal coordination to be designed into an IAS. Other essential features for a designer's desktop, such as an agent description language, interface definitions and communications standards, etc., are not discussed here.

A designer's desktop is defined as a library of agent prototypes and other self-contained components for agent systems, presented within a GUI that enables interactive assembly of agent prototypes and components into an architecture for a specific agent based system (ABS). Three basic building blocks considered necessary for such a design tool are described. Note that the concepts presented are intended for system architectures, and not agent architectures.

A description of the blocks and current work on them follows. A brief summary of their roles in system design is also provided.

Taxonomy of Intelligent Agent Systems

The first block is a taxonomy of system types, based on classifying the interactions between agents. These interactions cause the structuration of the IAS by generating conceptual emergent structures (which are sometimes realized as, and facilitated by, software artifacts). A taxonomy of intelligent agent systems can be constructed based on types and subtypes of emergent structures arising within a system. There is a large body of DAI research in such coordinative structures, but this has largely been in the context of exploring the individual concepts. (There do exist other classifications of ABS, e.g., the DPPS/MA and FA/C–CA/NA dichotomies, and domain type classifications.) Three fundamental system types based on coordinative interaction structures have been tentatively identified. A multi-level hierarchy of types and subtypes (e.g., genus, species, sub-species, etc.) is being developed. Descriptions of the basic types follow.

In a **team** system, a number of agents cooperate on achieving a common goal. The raison d'etre of a team is the achievement of the common goal, unlike the coalition or alliance (discussed later). Teams are characterized by a common intention, a well-defined scheme of communication between members, joint action, and full participation by members, (i.e., the team members' actions are all directed at achieving or supporting the common goal).

In a **coalition** or **alliance** system, cooperation is a matter of convenience for individual agents. The basic feature is utility maximization for individuals through joint action by the coalition or alliance. This type is subdivided into coalitions and alliances, based on the rationalizations for joining or leaving, the means of synthesizing the common goals, desires and intentions of individuals within the coalition or alliance, etc. A cost may be associated with joining, leaving, or staying in an alliance.

A **market** system is characterized by transactions between agents (e.g., in the form of bids and offers) and by a means of exchange (i.e., a currency). It is defined by the mechanisms that that bring together parties interested in transacting 'business'. Further subdivision of the market consists of:

- A 'commerce' market, where traders buy and sell items (securities, goods, commodities, consumer items, etc.). Transactions here are basically buy/sell transactions.
- A 'contract' market, where the participants consist of bidders, offerors, and contractors (plus ancillary agents such as contract managers, coordinators, etc.). Transactions here are basically contracts between agents to perform services or provide solutions to specified problems.

Conceptually, any of the emergent structures involved in the above classification is capable of functioning as a virtual agent. For example, a team may be regarded (particularly by non-members) as a single entity with its own intentions and goals. This feature allows abstraction of agent models. For example, if some agents are known to be working in an team, then — for most purposes — other agents need not consider the team's members individually.

These types may themselves be subdivided into various categories, and different subtypes will be appropriate for different kinds of IAS — e.g., coalitions that are helpful in a supply chain system system may be harmful (and perhaps illegal) in an electronic commerce system. Such subdivision will further develop the taxonomy of intelligent agent systems. Other emergent structures, such as global plans, are

*This work was partially supported by the National Science Foundation under grant IRI-9530271.

temporarily ignored for the purposes of this classification, but lead to subspeciation within the above broad categories. For example, an auction system is a commercial market that uses the auction as the primary mode of interaction; distributed planning systems are species within the Team genus.

Note that the possibility of a single system consisting of interacting subsystems of different species is not excluded by this paradigm. Indeed, such hybrid systems may give rise to interesting issues not yet explored in DAI.

Models of Optimal Coordination

Means of defining and measuring the optimality of coordination are needed in order to allow selection of the best coordination methods. A system designer may be able to decide, sometimes subjectively, that one kind of coordination mechanism will be better than another. Even so, unresolved questions about satisficing of coordination may remain (e.g., determining the optimal number of agents to contact with a specific type of query). We are currently formulating models of coordination for different classes of systems. This requires a formal or empirical theory of coordination. Current work by my group involves both formal and empirical models. The formal model involves modeling an IAS in a multi-dimensional space defined by characteristics such as the general level of autonomy of individual agents, size of agent neighborhoods (i.e., how many agents they interact with), aspiration levels of satisficing models of other agents, etc. The empirically oriented approach consists of two steps: (i) compositing input variables that define properties such as the structure of the society (e.g., branching factor for a tree-structured society), connectivity (the number of agents with which an agent can directly interact), resource availability, agent capabilities (in terms of the number of tasks an agent can accomplish relative to the task domain), etc., into 'emergent variables' by forming mathematical expressions comprised of two or more of these property descriptors; (ii) deriving relationships between these composite 'emergent variables' and a QoC (quality of coordination) measure. The definition of the QoC will be different for different classes of systems. Current work is concentrated on the distributed manufacturing domain, and the QoC measure we are using is a weighted average (over all agents) of a composite WIP (work in process) and tardiness measure.

Coordination Patterns

Much work on various coordination mechanisms has been done in DAI. A coordination pattern can be defined analogously to a design pattern in software engineering, as a description of the intent, context, applicability, structure, roles, collaborations, and consequences of a coordination process (or mechanism). (Aridor and Lange provide a similar definition in their Aglets work.) To this is added a 'profile', defined as a description of coordination regimes and pathologies expected under stated conditions. The conditions are expressed in terms of a formal or empirical model of coordination. The profile thus provides a link between the pattern and the relevant model of coordination. The 'coordination pattern' construct encompasses and extends Decker's concept of a library of coordination mechanisms and Barbuceanu's 'conversations'.

The key to using such coordination patterns in a designer's desktop is twofold: first, a formal language for expressing and encoding these patterns so they can be symbolically processed and manipulated by tools; and second, software artifacts embodying the patterns and the coordination mechanisms they contain. We are currently working on defining the software artifacts representing coordination patterns, given the relevant models of coordination for different system species. These artifacts operate either embedded within agents (as has nearly always been the case in DAI) or separably from (external to) the agents. The emphasis is on separately operating artifacts, which embody system patterns (as opposed to patterns for parts of individual agents, which can be implemented by the embeddable artifacts). The artifacts can be directly used for coordination design in a manner similar to the templates in the Aglets workbench, described by Lange and associates. These 'smart' (but not quite 'intelligent') artifacts are proposed as a means of distributing complexity between agents and their environs (the environment, communications links, etc.). (Complexity' here means both design complexity and time and space costs of computation involved in interaction.) Internal (embeddable) artifacts are widely studied in DAI (as various coordination mechanisms) and will not be discussed further here.

An abbreviated description of the envisaged design process follows, with a description of the roles played by the blocks.

DAI System Design

The system designer begins by selecting the appropriate species from the full ABS taxonomy described earlier. (Hybrid systems may involve identifying multiple ABS species.) Next, legacy elements (agents, interfaces and other components the designer is compelled to use) are identified and encoded in the description languages used by the tool. Following this, the constraints acting on the system are defined. There follows a stage of interleaved system and agent design, in which agents are designed and coordination patterns and artifacts are selected in interleaved steps. This stage consists of searches through the two spaces of agent and system designs. These interlinked searches are guided by the tradeoffs between appropriate models of coordination (evaluated by their coordination regimes and pathologies) and feasible and optimal designs for individual agents. The tradeoffs are determined by the constraints and criteria imposed by the domain, and by the environmental conditions specific to the system. This fourth stage (interleaved agent and system design) will, it is hoped, converge to an ABS in which the agents and coordination mechanisms are mutually suitable, and whose behavior under different operating conditions is predictable.