

A Pandemonium Can Have Goals

Giovanni Pezzulo (pezzulo@ip.rm.cnr.it)

Institute of Cognitive Science and Technology – CNR. Viale Marx, 15 – 00137 Roma, Italy
 and Università degli Studi di Roma “La Sapienza”, Piazzale Aldo Moro, 9 – 00185 Roma, Italy

Gianguglielmo Calvi (calvi@noze.it)

Institute of Cognitive Science and Technology - CNR
 Viale Marx, 15 – 00137 Roma – Italy

Abstract

AKIRA is an open-source framework for agent-based cognitive and socio-cognitive modeling and simulation. In this introductory paper we explain the underlying theoretical assumptions that lead to its central features, e.g. hybridism of the components, access to a common pool of resources, homeostasis of the system. In order to show the potentiality of AKIRA for cognitive modeling, we provide and example of a Goal Directed Agent. However, we are not focused on a single agent model, mechanism or computational tool; we use AKIRA as an “experimental laboratory” for modeling and implementing many cognitive functions (e.g. belief and goal dynamics, epistemic actions, anticipation, attention), exploring how higher-order cognition emerges from the interplay of many specialized agents and coalitions that compete, cooperate and learn how to exploit each other.

Introduction

The open source project AKIRA is a multi agent framework for cognitive and socio-cognitive modeling and simulations. The framework is inspired by the work of many cognitive scientists and philosophers (Minsky, 1986; Dennett, 1991), sharing some features with related computational models (Kokinov, 1994; Franklin, 1995; Sloman, 1999; Hofstadter, 1995), but retaining some unique ones.

Although our first interest is in high-order cognition, mainly in BDI-like (Rao & Georgeff, 1995) style, it is possible to use AKIRA for implementing a range of agents and architectures having different levels of complexity: reactive, logic-based, layered and autonomous agents; constraint satisfaction and connectionist networks. AKIRA is suitable for socio-cognitive, situated simulation, where cognitively rich agents are required in order to model e.g. trust and reputation dynamics (Falcone, 2004; Conte, 2001).

In order to explain the basic structure of AKIRA, we assume the Pandemonium metaphor (Jackson, 1987), describing its agents as Daemons that cooperate and compete, form Coalitions and communicate through a Blackboard. In order to show how Daemons and Coalitions can be exploited for cognitive modeling we explore a range of built-in features, such as the hybridism of the agents and the homeostasis of the system (Cannon, 1939).

Assuming *goal-directness* (Castelfranchi, 1995) as the basic functioning of an autonomous agent, we describe the facilities for implementing e.g. goal and belief dynamics. We also show the interplay between epistemic structures,

motivations and activity in the autonomous agents, adopting a constructivist and situated perspective (Piaget, 1975).

The Framework

The multi-agent framework AKIRA (<http://akira-project.org/>) is a run-time C++ multithreading environment for building and executing Agents and a web/system development platform to model their behavior and their interaction, as well as for interacting with the environment. AKIRA is implemented using state-of-the-art tools and design: this allows to build applications that are scalable and solid.

AKIRA provides a MACRO language and many templates for building Agents of different complexity (e.g. reactive, BDI-like). The whole system is written in C++ and integrates many different open source libraries. A number of soft computing technologies are included as basic features, e.g. Fuzzy Logic and Fuzzy Cognitive Maps (Kosko 1986).

BDI constructs are among the facilities embedded within the framework and reliable for agent programming. A strong multithread model ensures parallel, distributed computation.

Fig. 1. sketches the dynamics of the agents in AKIRA. The agents pass energy through an energetic network and communicate via a blackboard. Their computational resources (priority and memory space) can change during execution. Resources are an index of *contextual relevance*: more relevant agents have more resources for their operations and introduce a stronger *pressure* over the system, e.g. activating or inhibiting other agents.

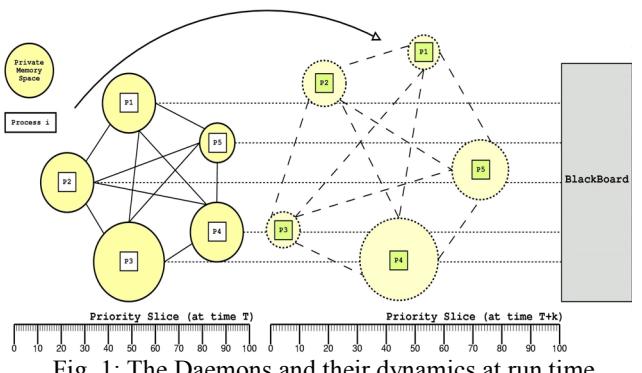


Fig. 1: The Daemons and their dynamics at run time.

AKIRA follows the Pandemonium metaphor (Jackson, 1987); its components are: the **Pandemonium** (kernel), the **Daemons** (micro agents) and the **Coalitions** (Daemon sets);

the **Blackboard** (XML stream); the **Energy Pool** (an abstraction for the computational resources).

The Pandemonium

The Pandemonium is the system kernel, the main process that instances the threads that are necessary to execute the Daemons (agents) and that executes all the monitoring and control operations over the single components. It is the “father process” that during start-up identifies the agents to load (accordingly to the constraints in the initial configuration) and during execution monitors the content of the XML Stream and of the single agents.

Demons

The **Daemons** (micro agents) are the minimal computational elements, each carrying its code, that are implemented as single threads. They are instantiated and executed by the Pandemonium during the system lifetime. Daemons have some features: a **priority** (set by the programmer), that gives a measure of absolute relevance; a **current activation level** (updated at each cycle), that gives a measure of contingent, contextual relevance; a **tap power**, that sets the access to the concurrent resources; a **symbolic operation**, that is the functional body; a **labeled link list**, that points to some other daemons and is the medium for spreading activation with a mechanism inspired by the Slipnet (Hofstadter 1995): the links are strengthened if the concepts associated to the labels are more contextually relevant.

Daemons are hybrid (Kokinov, 1994): they have both a symbolic and a connectionist side. With respect to their **symbolic side**, they can: *execute* the symbolic action they carry on, that can be performed if the contextual conditions are met (like productions) and if the energy is sufficient; *shout*, notifying their current activity and status to the other Daemons via the Blackboard. With respect to their **connectionist side**, they can *tap* energy from the Energy Pool; *spread*, giving it to linked Daemons; *release* it to the Energy Pool; *join* other agents in order to form complex structures called Coalitions.

Coalitions

The **Coalitions** are communities of cooperating daemons that can be created on-the-fly. Their purpose is to solve together complex, non atomic tasks. For example, in a composite pattern matching problem, each Daemon carries the code for matching a part (e.g. the subject, the sender and the address of an email). Differently from pure connectionist dynamics, Coalitions can cooperate and coordinate exchanging messages (e.g. for interactive tasks requiring explicit coordination). Coalitions arise and die in a dynamic way during computation as Daemons shift between them.

Coalitions can be nested and they can result both from bottom-up (*Bands*) and top-down (*Hordes*) pressures.

Bands. Bands arise when some Demons start to Shout in order to find help for a non atomic problem, in which its single symbolic operation is a non sufficient part; other Demons can respond to the Shout and start to Join. When two or more Demons Join in this way, a Band arises.

A Band is the resultant of auto-organization in a bottom-up fashion; its semantic is mainly driven from similarity and proximity (via the Link List) and concomitant activation (only active Daemons can join Coalitions). The topological structure of a Band carries semantic information about the Role that each Daemon assumes into it (as explained later).

Hordes. Hordes arise in a more top-down way and have a more structured, hierarchical shape. Normally special purpose Demons called **Archons**, start to Shout in order to recruit Demons. What's new is that they carry a non-atomic *Structure*, in which specific *Roles* for single Demons have to be fit. While in Bands the aggregation starts to build a structure in a “blind” way, in Hordes the prototypical skeleton of the structure is carried by Archons. In order to execute its symbolic operation, Archons try to recruit other Daemons, spreading them some energy, and taking advantage of their joining or of their symbolic operations.

Archons can be seen as active focuses recruiting Demons for their goal oriented operations; normally they are fueled by higher-level Daemons, e.g. representing internal drives.

The Blackboard

The communication medium is a shared data structure (Blackboard), divided into blocks containing XML packets, where the messages are concurrently written and read. Daemons can address their messages to specific Daemons, or to classes of Daemons, or to everything. Daemons actions and activation levels are notified to the Blackboard, too.

The Energy Pool

Energy is not a component; however, its dynamics are central in explaining the functionalities of the system. It exist in AKIRA as a global variable, *energy*, shared by all the agents, that gives a measure of the available computational resources. So all the daemons reside in an intrinsically concurrent environment with limited resources.

For each agent, more energy means more resources (e.g. more computational time). The total energetic level of the system, summing up all the priorities for all the agents, their tap power and the Pandemonium energy, gives an upper bound to the possible activation sequences of the threads during their lifetime. Energy can migrate between the agents, driving the sequences of activation of the threads.

Constraints for Cognitive Modeling

A main requirement for the AKIRA platform is to allow high-order cognitive modeling, mainly exploiting BDI-like constructs. AKIRA allows to implement *goal-directness* exploiting both the top-down control structures (e.g. from goals to plans, subgoaling) and the dynamics of parallel and concurrent computation (e.g. reactivity, behavioral modules, emergence and self organization).

The general AKIRA paradigm is MAS, but the level of complexity of an agent can be set at different levels. While for many kind of simple agents (e.g. reactive) it is sufficient a single daemon, a cognitive agent is conceived as a *macro-agent*, composed by many cooperating and concurrent *micro-agents* (Daemons), each representing e.g. a goal, or a

belief, etc. The environment, with its logic and dynamics, can be modeled as an agent, too, interacting with the other agents within the framework.

Here we describe the underlying theoretical assumptions.

Hybridism and Locality Principle

AKIRA agents have both a *symbolic* and a *connectionist* component. The **symbolic component** involves the set of operations an agent can perform. The symbolic operations can range from simple operations to very complex ones (e.g. involving reasoning; AKIRA furnishes to each agent a set of modules and facilities). However, accordingly to the underlying distributed, MAS approach, complex tasks are more likely performed by complex-agents or by Coalitions of cooperating agents. The **connectionist component** involves the activation level of the agent as well as the energy exchanges between the agents. A major difference exist with many neural-like architectures: the whole system is *homeostatic* and there are limited resources shared by all the agents, stored into the Energy Pool; the agents can tap or release energy. Spreading energy to another agent means losing it (like in Behavior Networks; Maes, 1990).

We also rely upon a **Locality Principle**: every interaction between the agents, both connectionist and symbolic, is implemented locally; the agents interact globally only with the Energy Pool. The medium for all the operations is the Blackboard; however, it is only a functional abstraction.

Hybridism is a central property of the system; it allows to perform a continuum of computation styles, ranging from centralized, hierarchical control to distributed, emergent computation. The *connectionist side* of AKIRA endorses the emergent and distributed properties of cognitive and socio-cognitive phenomena: the MAS perspective allows complex patterns of actions using autonomous and specialized computational units, i.e. Daemons. The *symbolic side* allows both to introduce top-down drives and structure and to manage operations requiring semantic compositionality.

Two Metaphors

The agents dynamic follows an **Energetic Metaphor** (Kokinov, 1994): greater activation corresponds to a greater computational power, i.e. speed. Each agent has an amount of computational resources (energy) that is proportional to its activation level (and is a measure of its relevance, both absolute and contextual, in the current computation). More active agents have a priority in their symbolic operations and their energetic exchange, and more frequent access to the Energetic Pool. This mechanism allows to model a range of cognitive phenomena such as context and priming effects, as explored mainly by Kokinov (1994).

At the same time, the system implements also a **Physical Work Metaphor**: performing symbolic operations has a cost in energy, that is paid by the performing agent to the Energetic Pool: this keeps the system conservative.

Energetic and Physical Work Metaphors let Demons compete for energy and for the access to the resources (e.g. access to the effectors). They also allow to characterize the interesting concept of *Temperature* (as introduced by Hofstadter, 1995) as an emergent property of the system.

Temperature of the system is represented by the currently used energy; it can increase and decrease over time and it is proportional to how far it is from a “solution”, if we make the assumption that many Daemons and Coalitions can compete as concurrent “hypothesis” in order to fit the data.

As in Copycat (Hofstadter, 1995) a *hot system* is far from stabilization and performs quick-and-dirty computation, with a rapid hypothesis shift; a *cold system* means stability and successful solutions, computing in a more accurate and conservative way. We can also give a meaning to some local dynamics; e.g. interesting, unsolved problems lead to hot Coalitions, with many Daemons joining it (see Baars, 1988).

Resources, Urgency and Pressures

A Demon or Coalition can act (i.e. *execute*) only if it has sufficient energy, because each symbolic operation has a cost. The cost can be seen as inverse to *urgency* of the behavior: less cost means more easily activated. So, urgent behaviors, like stimulus-response behaviors (as well as *alarms*, Sloman, 1999), can be represented with very low-cost operations; more complex cognitive operations are slower: they need to recruit a lot of energy, or exploit operations by other Daemons, or wait for one or more *join*.

As a consequence of system dynamics, each Daemon introduces a *pressure* over the computation in virtue of its presence. The system shows also implicit, contextual forces and pressures (e.g. set points) that may lead to Coalitions.

Daemons and Coalitions introduce contextual pressures in many ways: perceptual, goal-driven, cultural, conceptual, memory contexts are among those possible. As an effect of the Archons work, Demons that are somewhat related to the contexts are able to recruit more energy; this is true even if they are not able to join the Horde. But this is also true of Bands, where the pre-existing link topology is charged to embed in an implicit way a “similarity” semantic.

Learning and Coalitions. Daemons can exploit connectionist learning (involving the Link List) as well as symbolic learning. For example, a new Archon can be created by bottom-up pressures (e.g. if a Band shows persistence) or by top-down motivational pressures (e.g. via analogy). The prototypical structure of an Archon can be modified interacting with new exemplars of situations, as in Case Based Reasoning. These processes have the potential to model *assimilation* and *accommodation* (Piaget, 1975).

AKIRA Agents and Agent Societies

All the features hence described represent what we call the *constraints* given by the framework to any possible agent implementation: they are built-in and they cannot be violated, setting the “expressive power” of the framework. AKIRA allows to implement a MAS that is *homeostatic* and endorses *intrinsic concurrence between the agents*. Which kind of Agents and Agent Societies can be so implemented?

We define an *AKIRA Agent Society* as a set of agents working under a common Energetic Pool and competing for its limited resources; an *AKIRA Agent* is the unity which has a single, concurrent access to the Pool, proportionally to its activation/pertinence (both absolute and contextual). All the

energetic exchanges are local between the agents; all the symbolic operations have a cost in energy. So the system is conservative and it allows emergent phenomena. It is also possible to have many kernels implementing multiple societies that interact; to interface external components (e.g. agents or objects); and to implement agents persistence.

Although it is out of the boundaries of the present paper to argument the cognitive plausibility of the constraints embedded in AKIRA, they are not accidental but rather an essential part of the cognitive model: the expressive power of the each architecture is bounded by its implementation.

The first aim in designing AKIRA constraints and peculiarities is making it well suited for modeling a large set of cognitive and socio-cognitive phenomena. In this phase we do not commit to a specific architecture or model; we plan to use AKIRA as a framework for implementing and testing a number of functions and mechanisms, and trying to find interesting ways to let them interact and cooperate; this path to high-order cognition is inspired by the Society of Mind (Minsky, 1986). As an example of AKIRA's peculiarities for cognitive modeling, here we describe a Goal Directed Agent integrating BDI-like features.

Goal Directed Systems

According to Castelfranchi (1995), an autonomous, goal-directed agent is: able to generate its own goals, to select between multiple alternative goals to pursue, and to decide to adopt goals from others (e.g. for its own purposes).

We use AKIRA as a “cognitive modeling laboratory”: we have included many modeling tools (including BDI) as libraries, making it possible to implement and test many cognitive agent architectures. Moreover, AKIRA allows to situate the agents, building environments that constrain their actions and representations. We provide an example of an agent model, showing AKIRA's potential and expressive power. We sketch its motivational apparatus (involving Desires and Goals) and some of its epistemic features; we discuss its goal dynamics and belief building capabilities.

Desires and Goals. Desires are internal, top-level drives and the source of activation for Goals: an unique feature of self-motivated systems. Desires are *maintain* conditions (e.g. fuzzy variables to be kept into a given interval); they are stronger and they spread more activation if they are farer from the condition. Under certain conditions they can also build new Goals. Sometimes Desires intervention can be seen as normative (“adverbial”), in the sense that they constrain how the goals are performed (e.g. behave *politely*). So, they are well suited for implementing a (poor) version of Norms.

Goals have a condition to satisfy (e.g. in the form of a fuzzy variable); they become stronger the more they are close to the satisfying conditions; this accounts for a form of implicit commitment. Goal dynamics can be implemented by Archons; their concurrence is a built-in feature and there is no need of an external interpreter. Archons structure can represent several features: for example a temporal sequence or a control scheme (useful for Plans); or they can assume a role into complex structures. Goals activate Plans whose effects are positive for them and inhibit Plans whose effects

are negative. A Goal can be activated by Desires, as well as by bottom-up pressures, by Plans and even by Beliefs.

Planning and Subgoaling. Plans are control structures with Preconditions and Effects. They activate Goals that correspond to Preconditions to met; in this way an automatic subgoaling mechanism is achieved. Plans can activate and inhibit other Plans as well as Goals.

Plans do not need to be fully represented in the system; they can either be a part of an Archon, or only partially pre-determined: in this case, as in Behavior Networks (Maes, 1990), planning and control result from the dynamics of the system. Pre-planning everything is not always an advantage.

A Bridge between Knowledge and Action

Explicit Plans are action schemes, intrinsically calling procedures (*heuristics*) for dealing with the current situation as represented in the Archon structures. Such heuristics can work at the structural, descriptive level rather than the semantic one. Depending from the situation, given the same Plan structure, many strategies are suitable, such as: serial vs. parallel subgoal processing; direct goal exploitation vs. facilitation of side conditions, etc. In order to explain how to bridge Plans activity with representations, we introduce the epistemic role of the Coalitions and their structures, using the *Description & Situations* formalism (Gangemi, 2003).

Descriptions and Situations

Coalitions are not simply aggregates of Demons: they can have *structures*. We call these structures **Descriptions**; they represent a prototype of a concept, a situation, a theory, the abstract form of a solved problem. They contain also slots for *Roles* to be filled in e.g. by Daemons joining the Coalition. Descriptions carry on also the functional counterpart of the problem itself, i.e. *heuristics*: they are prototypical operations for dealing with them. Heuristics can not be directly applied to Descriptions (that are abstract entities), but to their concrete counterpart: a reified Description (e.g. in a Horde) represents a **Situation**.

Descriptions can be e.g. *hypothesis* that compete for explanation of phenomena. Following Activity Theory, representational activity is “organizing for use” but also *constraining* it: an active structure, once instantiated, constraints the way stimuli can be successively perceived and dealt with. This process is not only bottom-up, stimuli-driven: a cognitive agent has its internal drives. i.e. the axiological and operational counterpart is driven by the motivational apparatus, which embodies the desires of the system that compete in order to activate their functions.

Situations pair intelligibility with action possibility. Problem-solving capabilities are embodied in situations as their functional-effective counterpart. For instance, some Demons can be specialized in building or dealing with sequences, regardless of the objects that they are putting in the sequence, thus performing operations on the structures themselves. The constructive operation “put-in-sequence” builds a situation that is associated to some “plans-for-sequencing” and some “plans-to-deal-with-objects-in-a-sequence”. There is a functional link between situations and

objects that can take a *role* into, as well as there are constraints for the (structural) *heuristics* the system can apply to them. Therefore, constructive operations contribute to build a “cognitive map” of the environment, fully grounded in the internal, functional structure of the system.

Epistemic activity. Starting from perception, a stimulus is an “open problem” for a cognitive system: many Demons try to interact with it, singularly or in Coalitions (for more complex operations). A Demon that interacts with a stimulus “cuts it from the noise”; this constructive operation results in applying a *description* and building a *situation*. For example a Demon that carries a pattern matching operation only matches a certain pattern -and it is constrained to describe data in that way. Representing is not mirroring the environment, but the constructive operation of fitting stimuli into a schematic, functional structure: the constructive operation of perceiving a stimulus fits it into a description and constrains how the system can deal with it.

Since they carry little structural information, active Bands are conceived as a (mainly reactive) auto-organization in response to the stimuli of their environment, as unorganized or proto-organized data. At the contrary, Hordes can be seen as more structured attempts to extract (or impose) an interpretation, e.g. formulating an hypothesis. In both cases the epistemic apparatus is constrained by action, either in a reactive fashion or retaining the goal-oriented perspective of the top-down pressures. Due to the functional counterparts of the structures, knowledge and action are both endorsed by the same structure; this also results in a connection between procedural and declarative knowledge.

A Pandemonium Without Satan. Since all the information (e.g. about Daemons activity) is notified to the Blackboard, it can be exploited as data by the other agents, allowing *introspection*: some Daemons can be specialized in monitoring, interpreting and applying their functions to patterns of activity of other Daemons. It has to be pointed out that introspective agents have no special access to the private memory and resources of the other agents: they only interpret the results of their actions as meaningful data (they only have to know what the agents do, not how they do it).

Meta-reasoning is not a distinct module, but it is implemented by structures interpreting other structures as data, in a constructive way. The approach is **non-modular**: many active agents and coalitions, e.g. different *points of view* or conflicting Descriptions, as well as specialized cognitive functions, can be active (or partially active) at the same time, communicating, blending, exploiting and interrupting each other. Conflict management can be left to the energetic dynamics or solved by specialized agents, too.

Goals Dynamics: the Watchdog Example

Here we present an example of goal dynamics describing an agent having a set of Goals and a Norm.

The Watchdog agent patrols a house; it has a Norm: *stay always close to the house*, and some active Goals: #1 *walk around the house*; #2 *bark if you see an intruder*; #3 *chase and follow the intruders*; etc. Goals inhibit each other, too.

In order to fulfill goal #1 respecting the Norm, the Watchdog will follow circular trajectories around the house, standing always *close* (in fuzzy terms) to it. When an intruder arrives, in order to fulfill goal #2, the Watchdog will bark; if the intruder tries to fly out, in order to fulfill goal #3 the Watchdog has a pressure to follow it. In this case the Norm and Goal #3 are two contrasting pressures: the first to stay close to the house, the second to leave the house. The Watchdog trajectory results from a mix of those factors. Moreover, the internal dynamic of the system will follow some built-in rules for Goals and Norms: the goal becomes stronger the closer it is from its realization; the norm becomes stronger the farther it is from its realization; both become stronger as the Watchdog follows the intruder.

Assuming a slightly higher priority for the Norm, the Watchdog will follow the intruder, until: either the dog reaches him, or it goes too far from the house and the pressure of the Norm becomes stronger. The behavior of the Watchdog simply results from diverging pressures: the trajectory as well as the exact point where it comes back home are not pre-calculated. However, the effect can be amplified by a symbolic operation; e.g. after a while (when its clause is far from realization) the Norm can activate another goal: #4 *come back to the house*. An explicitly planned activity can also intervene: a Goal can activate a Plan involving a rigid “sentinel routine” e.g. follow a certain trajectory that includes each corner, bark each minute, etc.

The Watchdog behavior thus emerges from the *interplay between top-down and bottom-up components and pressures*. It can start as a reactive, stimuli-driven action, be modulated by contextual pressures, activate a Goal and shift to a proactive, top-down control sequence regulated by a Plan; all is done without a central interpreter.

The Watchdog behavior can also follow epistemic drives, e.g. proactively performing *epistemic actions* (Kirsh, 1994).

A Case Study: Constructive Belief Building

Let's sophisticate the Watchdog example, furnishing it the ability to perform *epistemic actions*. These are actions explicitly aimed at acquiring information (e.g. look at the world, waiting for) and building or revising a belief.

Building a Belief is a constructive, goal-driven action: normally a Belief is built because a Goal requires it (e.g. as a precondition). Rather than only collecting input data, the agent can proactively “ask questions to” (the environment, a set of data, another agent); questions always embeds the “point of view” and the intention of the asker, i.e. the (goal driven) Descriptions, as well as some contextual pressures.

In a cognitive perspective, Beliefs are *explicit* epistemic atoms (mainly declarative¹). In a specific implementation, building a belief can mean e.g. assigning a value to a fuzzy variable: this operation carries on the schematic structure of the goal (e.g. the Description it carries and the metric for that specific fuzzy variable). It also introduces a number of top-down constraints derived by Daemons dynamics, e.g. selective attention (driven by the activity), priming and contextual effects, interference with other Daemons. Using

¹ The hybrid formalism allows also to model *implicit knowledge* as an epistemic pressure, distributed and not explicitly represented.

fuzzy logic a Belief, like any predicate, can express *degrees* (i.e. this place is *quite* far). A belief has also an associated *strength*, a degree of epistemic certainty (i.e. I am *rather sure* that this place is quite far). According to Castelfranchi (1995), the strength of a belief is a function of its sustaining sources: how many sources I have queried; how much accurate and reliable they are. Thus, building a belief consists in a set of epistemic actions towards a number of sources; since the sources can be of different kinds, this can lead to different pragmatic actions: e.g. consider perceptual or stored data; ask a question to another agent. This activity is similar to a “detective” that formulate, confirms and falsifies hypothesis; however it has to be stressed that the main drive of the epistemic action is action-oriented. All the information has to be unified into a single Belief: it is a complex cognitive activity that exploits a set of heuristics (e.g. for mixing converging and diverging sources, for managing contradictions) and that subdues to many bias and contextual influences (Castelfranchi, 1995). As a consequence of the constructive procedure, a *degree* and a *strength* are always associated to any Belief. In the same way, exploiting more abstract Descriptions, we can build complex epistemic structures (e.g. causal explanations), in which beliefs take a Roles (e.g. “the core of the theory”)².

This belief building procedure is broad enough to comprehend even perception: it is both data-driven and proactive, depending on the current hypothesis of the agent.

Giving the Watchdog the possibility to perform epistemic actions and to proactively “ask questions” allows it to enrich its epistemic apparatus (formulating and comparing hypothesis, e.g. “this corner is now secure”). It can also perform new behaviors (e.g. search and follow the footprints of the intruder). We use the belief building mechanism for modeling anticipatory, proactive Watchdogs, exploiting constructive perception: they build expectations and “asks questions” in order to confirm or falsify them. They do not simply react to input data, but they “look at the world” with a set of hypothesis, thus reacting to the discrepancy between expectations and observations, i.e. to *surprise*.

Conclusions

We have presented AKIRA, a framework for cognitive and socio-cognitive modeling. We have described some of its features and assumptions, including hybridism, energetic and physical work metaphor, allowing to model and implement a range of agent architectures. We have described how to model Goal Directed Agents in AKIRA.

Currently we are using AKIRA for implementing a set of cognitive models and functions (mainly developed at ISTC-CNR) including: plans for delegation, monitoring and control; a quantification of the strength of the beliefs as a function of the trust in their sources (Falcone, 2004);

2 In computational terms another important question is: when does an agent consider a belief “solid enough”, i.e. when does it stops asking the sources? For this dimension, we are investigating the *certainty* parameter (Pezzulo, 2004): it involves *ignorance* (how much things I do know that I don’t know), *perceived contradiction* (the degree of contradiction in my data), and *uncertainty* (that involves a comparison with competing hypothesis).

expectations and epistemic actions (Pezzulo, 2004); uncertainty and belief revision (Pezzulo, 2004). They will be included into the framework as a set of functions for agent modeling; allowing to explore how their interplay realizes high-order cognition. We plan to include the functionality of simulators (Barsalou, 1999); to interface the foundational ontology DOLCE (Gangemi, 2003); and to implement some architectural features such as data analysis tools, programming interfaces and data exchange protocols.

Even if our models and functions are not yet mature for empirical verification, in order to validate and refine them we are performing a set of experiments involving human subjects: e.g. how do they mix different, possibly discordant information; how do they build and revise trust; how do they manage uncertainty. Allowing empirical testing will be crucial in order to exploit AKIRA for cognitive modeling.

References

Baars B. J.. *A Cognitive Theory of Consciousness*. Cambridge University Press, Cambridge, MA, (1988).

Barsalou L.W., Perceptual symbol systems, *Behavioral and Brain Sciences*, 22, 577-609, 1999.

Cannon W. B.. *The wisdom of the body*. Norton, New York, 1939.

Castelfranchi C.. Guarantees for autonomy in cognitive agent architecture. In M. Wooldridge and N. R. Jennings, editors, *Intelligent Agents: Theories, Architectures, and Languages*, LNIA 890, pages 56–70. Springer-Verlag, 1995

Conte, R. Emergent (Info)Institutions, *Cognitive Systems Research*, 2001, Volume 2, Issue 2

Dennett D. *Consciousness Explained*, Boston: Little, Brown, 1991

Falcone R., Pezzulo G., Castelfranchi C., Calvi G.. Why a cognitive trustier performs better: Simulating trust-based Contract Nets. *Proceedings of AAMAS 2004*.

Franklin S., *Artificial Minds*. Cambridge, MA: The MIT Press, 1995

Gangemi A, Mika P. Understanding the Semantic Web through Descriptions and Situation. *LNCS 2519*, Springer Verlag, 2003.

Hofstadter D. R. and FARG, *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought*, Basic Books, New York, (1995).

Jackson J. V., Idea for a Mind. *Siggart Newsletter*, 181, (1987).

Kirsh, D. & Maglio, P. (1994). On distinguishing epistemic from pragmatic action. *Cognitive Science*, 18, 513–549.

Kokinov B. N., The context-sensitive cognitive architecture DUAL, in *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, Lawrence Erlbaum Associates, (1994).

Kosko B. (1986). Fuzzy Cognitive Maps. *International Journal Man-Machine Studies*, vol. 24, pp.65-75.

Maes P., Situated Agents Can Have Goals. *Robotics and Autonomous Systems*, 6 (1990).

Minsky M. *The Society of Mind*. Simon and Schuster, NY, 1986.

Pezzulo G., Lorini E.. Calvi G. How do I Know how much I don’t Know? A cognitive approach about Uncertainty and Ignorance. *Proceedings of COGSCI 2004*.

Pezzulo, G., Calvi G.. AKIRA: a Framework for MABS. *Proceedings of MAMABS 2004*

Piaget J.. *L'équilibration des structures cognitives: problème central du développement*. Pr. Univ. de France, Paris (1975).

Rao A., Georgeff M., BDI Agents from Theory to Practice, *Technical Note 56, AAI*, April 1995.

Sloman A. *What sort of architecture is required for a human-like agent?* In: M. Wooldridge. A. Rao (eds). *Foundations of Rational Agency*, Kluwer Academic Publishers, 1999.