

Ethos: A MAS Framework for Modeling Human Social Behavior and Culture

Jorge P. F. Simão
{jsimao@di.fct.unl.pt}

Luís Moniz Pereira
{lmp@di.fct.unl.pt}

Centro de Inteligência Artificial – CENTRIA
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
2829 - 516 Caparica, Portugal

Abstract: We present a framework for a Multi-Agent System (MAS) devised to support the modeling and simulation of agent-based models of human social behavior and culture change. We set forth its main abstractions, and test the usefulness and generality of the framework by describing how two previously published models from the literature have been re-implemented in it. We argue that our framework provides features that simplify the modeling process of a wide range of models of human social behavior, beyond what current MAS accomplish.

Talk Outline

- **Ethos** Simulation Framework Design Goals
- Current ABM Simulation Frameworks
- Building Blocks for Computational Social Theory
- **Ethos** Simulation Framework Overview
- Example: A Gene-Culture Co-Evolutionary Model
- Human Mate Choice: Case Study I
- The Cultural Evolution of Preferences: Case Study II
- Conclusions and Future Work

Ethos Simulation Framework Design

Goals (1)

Ethos is an Object-Oriented Simulation Framework (implemented in **Java**)

- Computational Support for Social Theory Building
 - Reify in Software Useful Theoretical Constructs (shared and/or plausible)
 - Experiment with Variations of Theoretical Constructs
 - Reuse Theoretical Constructs
 - Easy (Re)Implementation and Extension of Large Array of Models
 - Easy to explore Model and Theory Space

Ethos Simulation Framework Design Goals (2)

- General Computational Requirements
 - Expressiveness and Flexibility
 - Extensibility and Modificability
 - Transparency
 - Performance, Scalability, and Robustness
 - Portability and Ease of Use

Current ABM Simulation Frameworks

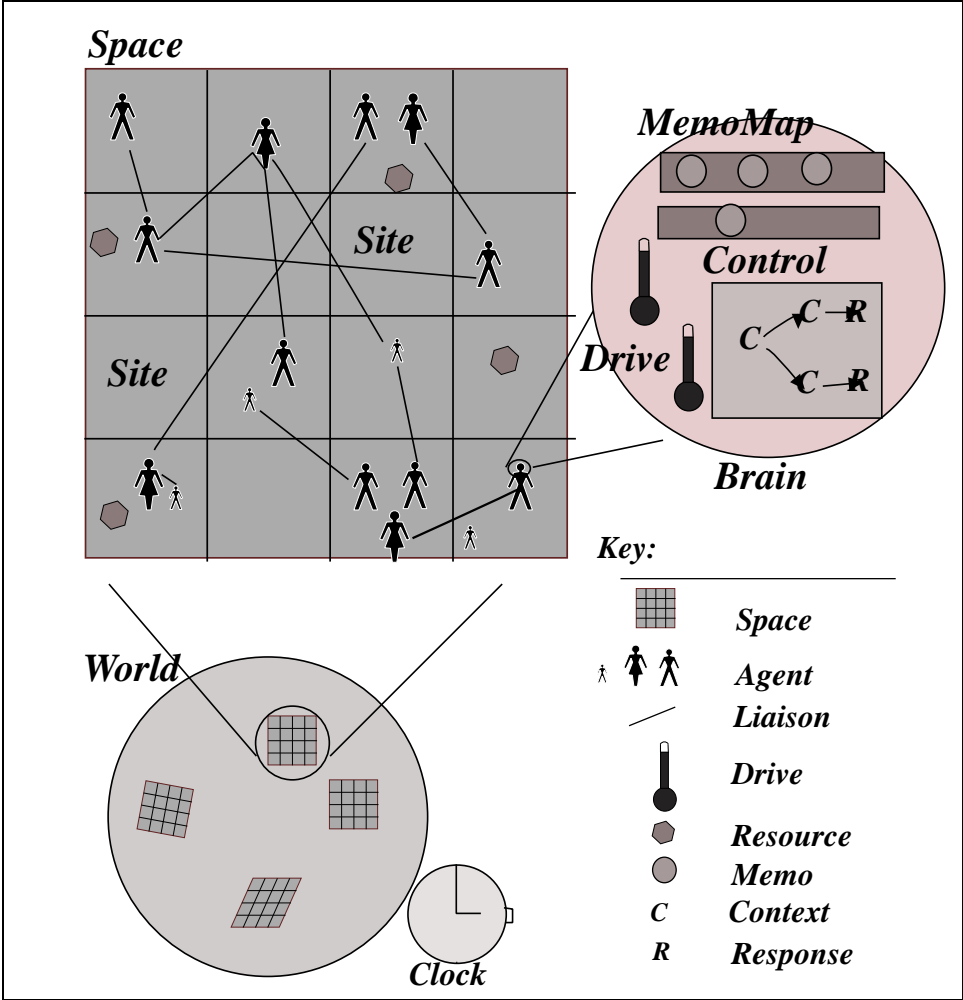
- **Swarm, RePast, Ascape**
 - + Good Support for General Computational Service
 - Lacks Specific Support for Social Theory Building
- **PS-I**
 - + Support for Social Theory
 - Targeted only to a Specific Set of Mid-Range Theories: Constructivist Identity Theories
- **Evo**
 - + Support for Evolutionary Discovery of Behavior Strategies
 - Limited Plausibility of Mechanisms (Evolutionary Programming)
- **Starlog, AgentSheets**
 - + Easy to Use
 - Mostly Limited to “Toy” Models
- **Sugarscape, Conumat, . . .** (other highly parameterized models)
 - + Interesting Case Studies
 - Not a Generic Simulation Framework

Building Blocks for Computational Social Theory

- (Meta-)Models of Human Cognition and Behavior
 - Abstract Models of Brain Structure and Function
 - Abstract Recurrent Modes of Human Behavior and Social Interaction
 - Insights (and Lessons) from Traditional and Situated Cognitive Science
- Mechanisms of Social Transmission and Influence and Cultural Dissemination
 - Animal and Human Social Learning Theory
 - Support for multiple mechanisms: contagium vs. transmission; direct influence vs. indirect through shared environment; . . .
- Models of Social Structure and Organization
 - Complexity Theory
 - Multi-level Selection and Co-evolution
 - Developmental Systems Theories

Ethos Simulation Framework Overview (1)

- Physical Environment Structure
 - **Space** is the unit of spatial layout; topological arrangement of **Site**
 - **Site** have any number of **Body**
 - **Body** represents a physical entity: (Human) Agent, Resource, Organization
 - **World** as aggregation of **Space**: (possibly) coordinating network distribution



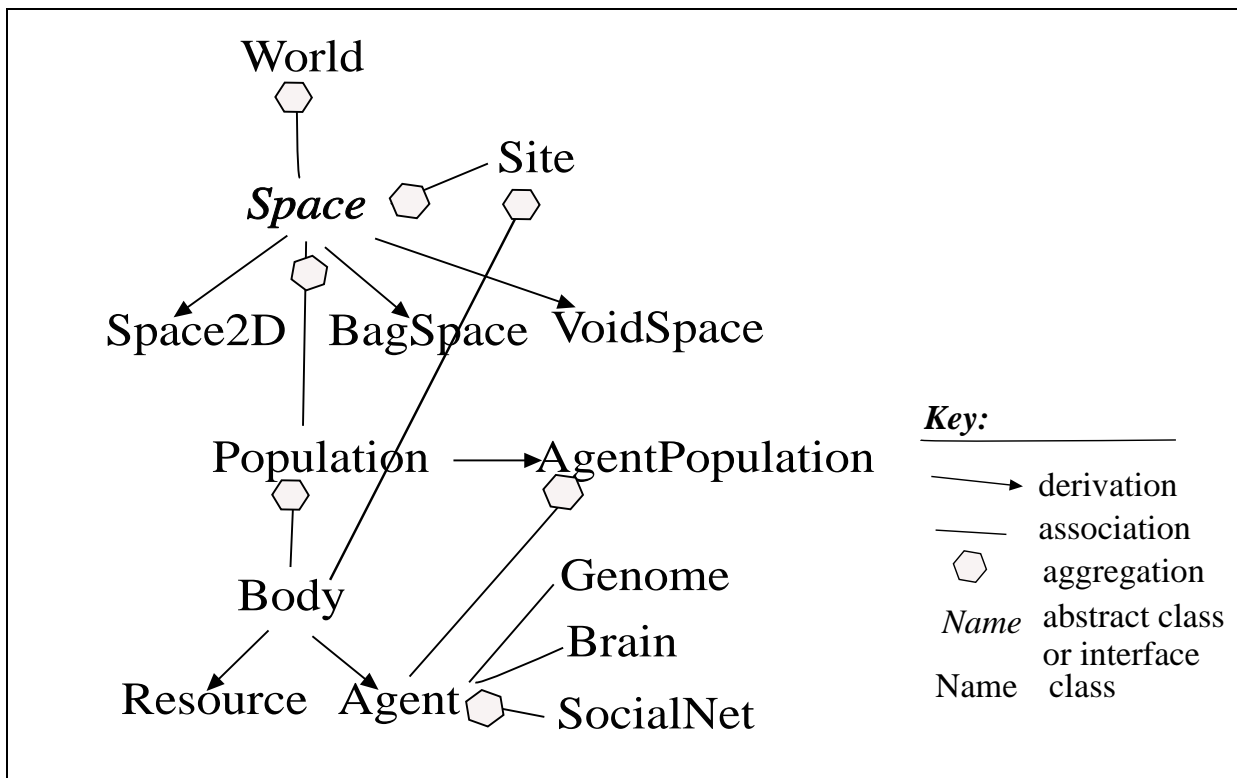
Ethos Simulation Framework Overview

(2)

- (Human) Agent Structure

Agent = Genome + Visible Attributes + Social Networks + Brain

- **Genome** is a set of inherited traits
- **Attr** is a visible agent attribute (e.g. physical); can be distorted through a **Liaison**
- **Liaison** is a connection between agents in a **SocialNet**
- **Selector** objects used as reusable selection criteria mechanism: **SocialNet**, ...
- **Brain** is the memory storage and behavior control mechanism

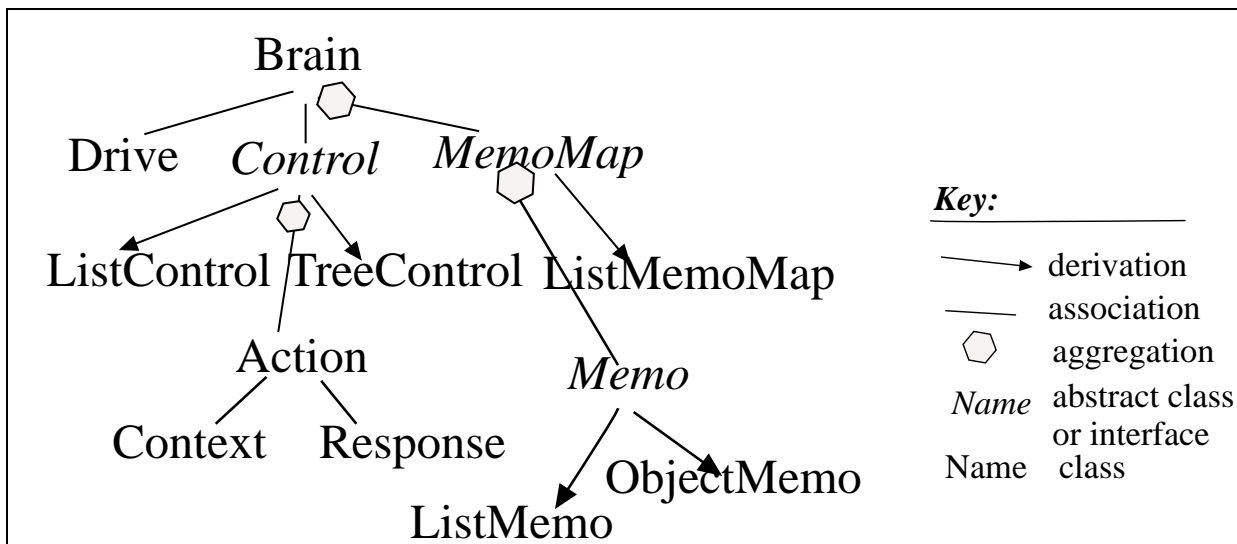


Ethos Simulation Framework Overview (3)

- Brain Structure

Brain = Memory Maps + Internal Drives + Behavior Control

- **MemoMap** is a set o memory items; (possibly) with interaction and *holistic* interpretation
- **Memo** is an assembly of memory state value competing for *storage*, *activation*, and *valuation*
- **Drive** is a dynamic internal variable which (can) influence action selection
- **Control** is the action selection mechanism
- **Action = Context + Response**



Ethos Simulation Framework Overview

(4)

- Event Management and Behavior Scheduling
(Hetra-)Hierarchical Discrete time step events
 - **Population** are aggregations of **Body**; coordinate their activities
 - **Population** also place-holder for operations at aggregate level
 - **Population** set associated with a **Space**
 - **Agent** follows a multi-phase behavior act:
 1. *inspect*: **Stimulus** objects change **Brain** state
 2. *feedback*: direct social feedback from other agents delivered
 3. *respond*: get responses generated by brain; internal responses performed
 4. *perform*: external action performed on environment; show for observation to other agents

Example: A Gene-Culture Co-Evolutionary Model (1)

```
// Higgs's Mimetic Transition Model.

package ethos.demo.mimetic;

import ethos.model.*; ...

class MimeticWorld extends World {
    final int K = 2; ... //model parameters

    class MyAgent extends Agent {
        double biofit , cultfit;

        MyAgent() {
            setBrain(new MyBrain());
            setGenome(new Genome(new Gene[] { ... }));
            setMaxAge(3);
        }

        class MyBrain extends Brain {
            class Meme extends ObjectMemo {
                double fitbio , fitcult; int id;
                Meme(int id) { ... }
            }

            MyBrain() {
                MemoMap memes = new ListMemoMap() {
                    //probablistic storage of each meme
                    //override default method
                    public detect(Stimulus s) {
                        List ms = (List) s.getObject();
                        for (Iterator i=ms.iterator();
                            i.hasNext();){
                            Memo m = (Memo) i.next();
                            if (Rnd.urandom()<learningAbility()) {
                                super.detect(s);
                            }
                        }
                    }
                }
            }
        }
    }
}

addMemoMap(memes);
```

Example: A Gene-Culture Co-Evolutionary Model (2)

```
// continuation

// Typical Control set up
ctrl.addAction(new Context() {
    public double getActivation() { ... }}
, new Response() {
    public void perform() { ... }});
}

public void inspectSetup() {
    if (getParent1() == null //no parent
        || getAge() >= 2) { //too old
        return;
    }
    Population pop =
        getParent1().getPopulation(0);
    List teachers = (new Selector(K, 1)).
        select(pop.getMembers());
    toInspect(teachers);
}

public Object responseSetup() {
    if (Rnd.urandom() <= PINV) { //invent meme
        memes.add(new Meme(++memeid));
    }
    getStimulus().setObject(
        memes.getAllStimulus()); //show all memes
    return null;
}

double getFitness(int fitid) { ... }

double learningAbility() { ... }
}
```

Example: A Gene-Culture Co-evolutionary model (3)

```
// continuation

class MyAgentCreator extends
  AgentCreatorImpl { ... }

MimeticWorld() {
  addSpace(new VoidSpace());
  Population pop = new AgentPopulation();
  AgentCreator agc = new MyAgentCreator();
  pop.addMembers(agc.createN(N));
  pop.setAutoRemove(true);
  getSpace(0).addPopulation(pop);
}

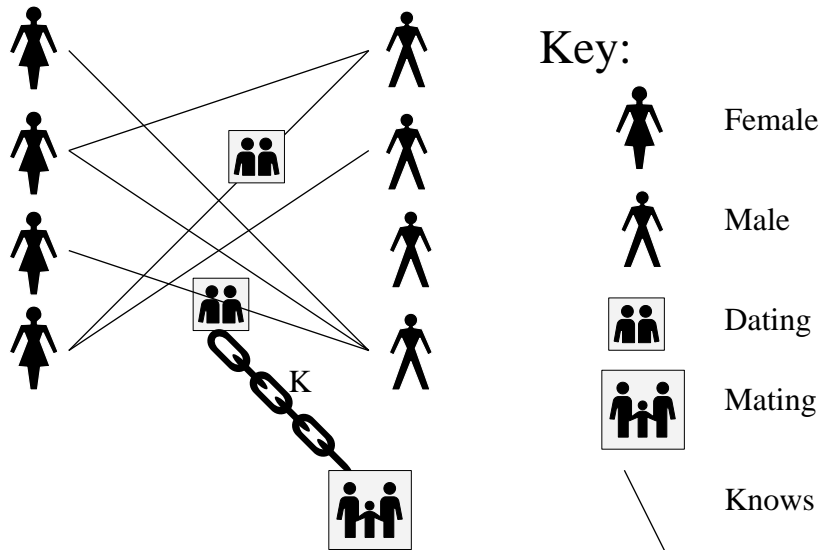
void step() {
  super.step();
  List pairs = getSpace(0).getPopulation(0).
    mateUniSexual(new Selector(N, 0));

  List off = AgentPopulation.
    generateOffspring(pairs, null);

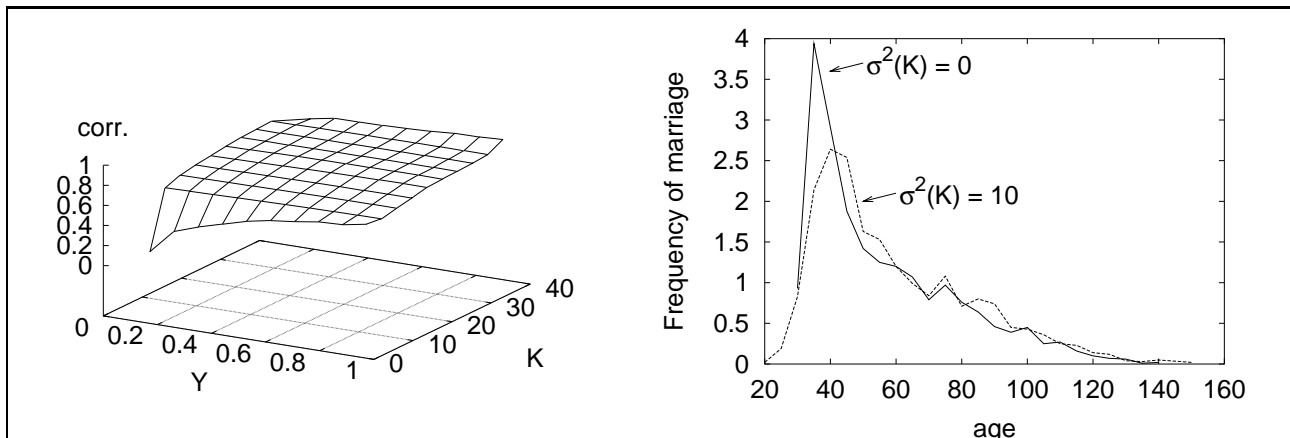
  Population pop = new AgentPopulation(off);
  pop.setAutoRemove(true);
  getSpace(0).addPopulation(pop2);
}

public static void main(String[] args)
throws IOException {
  MimeticWorld model = new MimeticWorld();
  model.run(NRUNS, NGENS);
}
}
```

Human Mate Choice: Case Study I



$$F(q_m, t) = q_m \times \frac{L - t}{L}$$



a) Correlation of qualities in each mated pair; b) Distribution of age at mating (marriage)

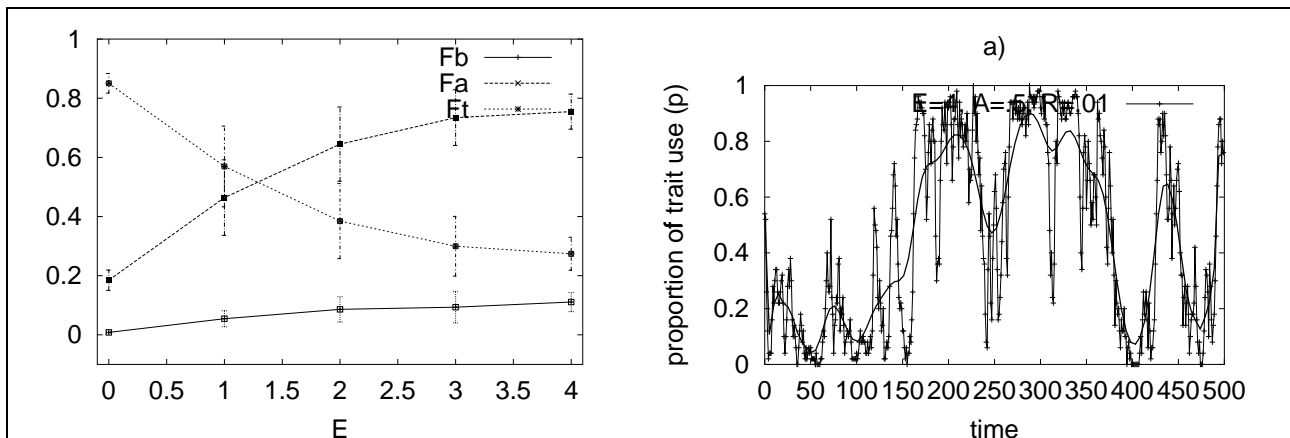
The Cultural Evolution of Preferences: Case Study II

Population of P Agents: $a_i = \langle q_i, t_i, v_i^0, v_i^1 \rangle$.

Preference changes:

$$v_i^1(t) = v_i^1(t-1) \cdot \alpha + \sum_{a_j: a_j \in M_i \wedge t_j=1} q_j \cdot (1-\alpha)$$

$$v_i^0(t) = v_i^0(t-1) \cdot \alpha + \sum_{a_j: a_j \in M_i \wedge t_j=0} q_j \cdot (1-\alpha)$$



a) Measures of fashion like behavior as a function of assortment (social stratification); b) Proportion of trait use in an example simulation run

Conclusions and Future Work

- ABM Software support for Social Theory Building
 - Is Feasible: Identify Key Foundational Abstractions
 - Is Useful: Simplifies Theory Building, Comparison, and Testing
 - (Possibly) Required: Contributes to the Unification of the Social Sciences
- Further Developments in **Ethos**
 - (Re)Implement Additional Models
 - Refine and Add Abstractions (if and as needed)
 - Make Software Publicly Available