Environmental Modelling & Software 61 (2014) 19–38

Contents lists available at ScienceDirect

Environmental Modelling & Software

journal homepage: www.elsevier.com/locate/envsoft

An open framework for agent based modelling of agricultural land use change

Dave Murray-Rust [a,*], Derek T. Robinson [b], Eleonore Guillem [c], Eleni Karali [d], Mark Rounsevell [e]

[a] School of Informatics, University of Edinburgh, Edinburgh, UK
[b] Department of Geography and Environmental Management, University of Waterloo, Canada
[c] Scotland's Rural College, University of Edinburgh, UK
[d] Centro Euro-Mediterraneo sui Cambiamenti Climatici, Italy
[e] School of GeoSciences, University of Edinburgh, UK

ARTICLE INFO

Article history:
Received 22 February 2014
Received in revised form
17 June 2014
Accepted 27 June 2014
Available online 22 July 2014

Keywords:
Agent based modelling
Land use/land cover change
Open source
Modelling frameworks
Ecosystem services

ABSTRACT

There is growing interest in creating empirically grounded agent based models (ABMs) to simulate land use change at a variety of spatio-temporal scales. The development of land use change models is challenging, as there is a need to connect representations of human behavioural processes to simulations of the biophysical environment. This paper presents a new agent-based modelling framework (Aporia) that has the goal of reducing the complexity and difficulty of constructing high-fidelity land use models. Building on earlier conceptual developments for modelling land use change and the provision of ecosystem services, Aporia was designed to be modular, flexible and open, using a declarative, compositional approach to create complex models from subcomponents. The framework can be tightly or loosely coupled with multiple vegetation models, it can be set up to evaluate a range of ecosystem service indicators, and it can be calibrated for a range of different landscape-scale case studies and modelling styles. The framework is released under an Open Source licence, and can be freely re-used and modified to form the basis of new models. We illustrate this with two case studies implemented using Aporia, exploring different socio-economic scenarios and behavioural characteristics on the land use decisions of Swiss and Scottish farmers. We also discuss the benefits of frameworks in terms of their flexibility, expandability, verification and transparency.

Software availability

Name of software: Aporia
Developer: Dave Murray-Rust
First available year: 2011
Software requirements: Java, Eclipse, Repast Simphony
Programming language: Java
Program availability and cost: Free, GPL, http://www.wiki.ed.ac.uk/display/aporia
Contact person: Dave Murray-Rust
E-mail: aporia@mo-seph.com

1. Introduction

Agent based models (ABMs) are increasingly used to model human–environment interactions (Rounsevell et al., 2012a,b). The degree to which these models have been empirically grounded (Janssen and Ostrom, 2006; Robinson et al., 2007) has also steadily increased from simple theoretical models to high fidelity models (e.g., Transims; Toroczkai and Eubank, 2006). In many cases, higher-fidelity ABMs are the result of coupling among models that enable dynamic feedback responses to human decision-making from human, natural, or both systems. For example, when examining the effects of land management decisions, ABMs have been coupled to biophysical models that simulate processes such as plant phenology, crop vegetation growth, and water cycling (Bithell and Brasington, 2009; Luus et al., 2013; Monticino et al., 2007). From design through to implementation, the construction of land-

---

* Corresponding author.
E-mail address: D.Murray-Rust@ed.ac.uk (D. Murray-Rust).

http://dx.doi.org/10.1016/j.envsoft.2014.06.027
1364-8152/

use ABMs often involves developing avenues to link with dynamic vegetation models (DVMs) and individual based models (IBMs) to estimate the impacts of human behaviour on ecological systems (Murray-Rust et al., 2011).

The conceptual and computational challenges associated with developing complex and/or interconnecting multiple models puts a huge burden on researchers and modellers, who are often not trained in computer science or do not have in-depth programming experience. However, a collection of tools exist that can aid researchers "dealing with complexity, re-using modules for different models, and providing support for commonly needed services" (Evert and Holzworth, 2005). Each tool or modelling approach chosen or developed forces the researcher to make a number of trade-offs (e.g., Parker et al., 2003). Ideally our tools should be used to construct models that "maximize generality, realism, and precision toward the overlapping, but not identical goals of understanding, predicting, and modifying nature" (Levins, 1966, p. 422).

Generic toolkits such as Repast (North et al., 2005) and Swarm (Minar et al., 1996) have been widely used and offer the greatest flexibility for researchers to develop ABMs (see Parker et al. (2003) and Nikolai and Madey (2009) for reviews). These toolkits enable the construction of a complete range of models that start with simple toy models (e.g., Boids; Reynolds, 1987). Simple toy models have the benefit that they do not require data, they can be rapidly developed, and they can demonstrate that a set of rules, mechanisms, or processes can produce observed phenomena (i.e., a proof of existence; Waldrop, 1990). Additional data, detail, and complexity can be incorporated systematically at a later stage to increase the degree of realism and precision of a real-world system at the expense of general applicability.

In the study of land-use change, most ABMs are constructed solely from generic toolkits and represent highly specific case-study systems. For example, ABMs representing land-use change have been developed for Altamira, Brazil (Deadman et al., 2004); south-central Indiana, US (Evans and Kelley, 2004); Koper, Slovenia (Robinson et al., 2012); Mayuge District, Uganda (Berger and Schreinemachers, 2006; Schreinemachers and Berger, 2006); San Mariano, Isabela, Philippines (Huigen, 2004); Mae Salaep village, Chiang Rai, Thailand (Barnaud et al., 2005); Washington D.C., US (Irwin and Bockstael, 2002); EU Special Protection Area number 56, central Spain (Millington et al., 2008); and Denton County, Texas, US (Monticino et al., 2007) among many other locations. Few, if any, of these and other ABMs of Land Use and Cover Change (LUCC) have been applied across multiple study areas, which may be attributed to "a lack of funds or labour resources to go beyond core scientific objectives of the research; scientists who are often not software engineers by training and may be hesitant to release models that are inelegantly coded; and/or the extensive time and effort to develop models, which often suggests that individuals optimise their use of a model for specific publications …" (Evans et al., 2013).

To enable others to use and apply a site specific model to other locations requires an investment in time that is likely to have little payoff for small research projects. If resources and time are available to produce a well-constructed model framework, then the trade-offs among generality, realism, and precision can be balanced and a number of benefits ensue, such as re-usability of code, comparability of results and extensibility of models. In this context, a framework is a collection of building blocks (i.e., coded methods) and generic land-use system structure (i.e., abstract classes representing actors in the system, how they can interact and behave, as well as scheduling actions) that enable researchers to focus on conceptual representations of the study system, justification of model parameterisation, and calibration rather than developing a model from scratch. Frameworks are significantly more refined than general ABM toolkits, as they integrate domain knowledge and preassemble building blocks that facilitate domain-specific research questions (e.g., land-use change). Furthermore, a well-defined and designed *framework* can extend the application of a model to multiple case study locations and enable non-programmers to initialise the model for specific contexts, scenarios, or computational experiments (Schweitzer et al., 2011).

This paper presents a new state-of-the-art framework (Aporia) that was developed to aid the creation of agent-based land-use models, at multiple locations with unique socio-economic contexts, by local experts and academics without extensive coding and software engineering experience. Through the configuration of Aporia, researchers can investigate a variety of research questions, such as: How does the relative influence of social factors versus economic and environmental factors guide on-farm crop rotation selection and the provision of ecosystem services? What is the degree of subsidy adoption among farm households and how effective are subsidies in achieving their economic or environmental goals? And, under what conditions is marginal land taken out of (or put into) agricultural production? To answer these types of questions and enable others to also do so, the creation of Aporia focused on the agricultural land-use system.

While other ABM frameworks are available (e.g., MameLuke; Huigen, 2004) that allow for a range of actors to be modelled, and offer a strong theoretical basis on which to build decision-making by creating sets of rules for agents, Aporia, does not impose a specific decision architecture on agents – however, this comes at the cost of requiring increased coding effort if one wants to move beyond the architectures already implemented. Similarly, some frameworks are focused on capturing the dynamics of coupled human–environment systems, in particular human adaptation to changing biophysical factors (LUDAS; Le et al., 2008, 2012). Others (e.g., PALM; Matthews, 2006; SLUCEII; Robinson et al., 2013) have a greater focus on land use change and subsistence agriculture or exurban growth more specifically. Aporia differs from these existing developments in that it provides substantial detail in terms of representing the attitudes of agents, the complex interplay of factors that are prevalent in more intensively farmed areas, and a modern software engineered approach that facilitates the easy integration of geospatial data, a progression from simple to more complex models, and a detailed view of farming practices and their socio-political context, that can all be configured or assembled without computer programming (i.e., through extensible mark-up language [XML]).

We designed Aporia to provide 1) an off-the-shelf framework for researchers to easily create models; 2) comprise a suite of output metrics that when applied across multiple study sites can foster comparison, synthesis, and generalisation of research findings; and 3) a baseline for extension and improvement to support model innovation and to answer novel research questions. To demonstrate these design criteria, the remainder of this paper provides an overview and description of Aporia components. An example of a simple stylised model that is extended by adding Aporia components to increase the complexity of representation of the land system. Then two empirically-grounded case study applications are presented, Aarau, Switzerland (Karali, 2012; Karali et al., 2013) and the Lunan area of Scotland, UK (Guillem and Barnes, 2012; Guillem et al., in review). We also discuss the framework and how it compares to other land-use-modelling frameworks.

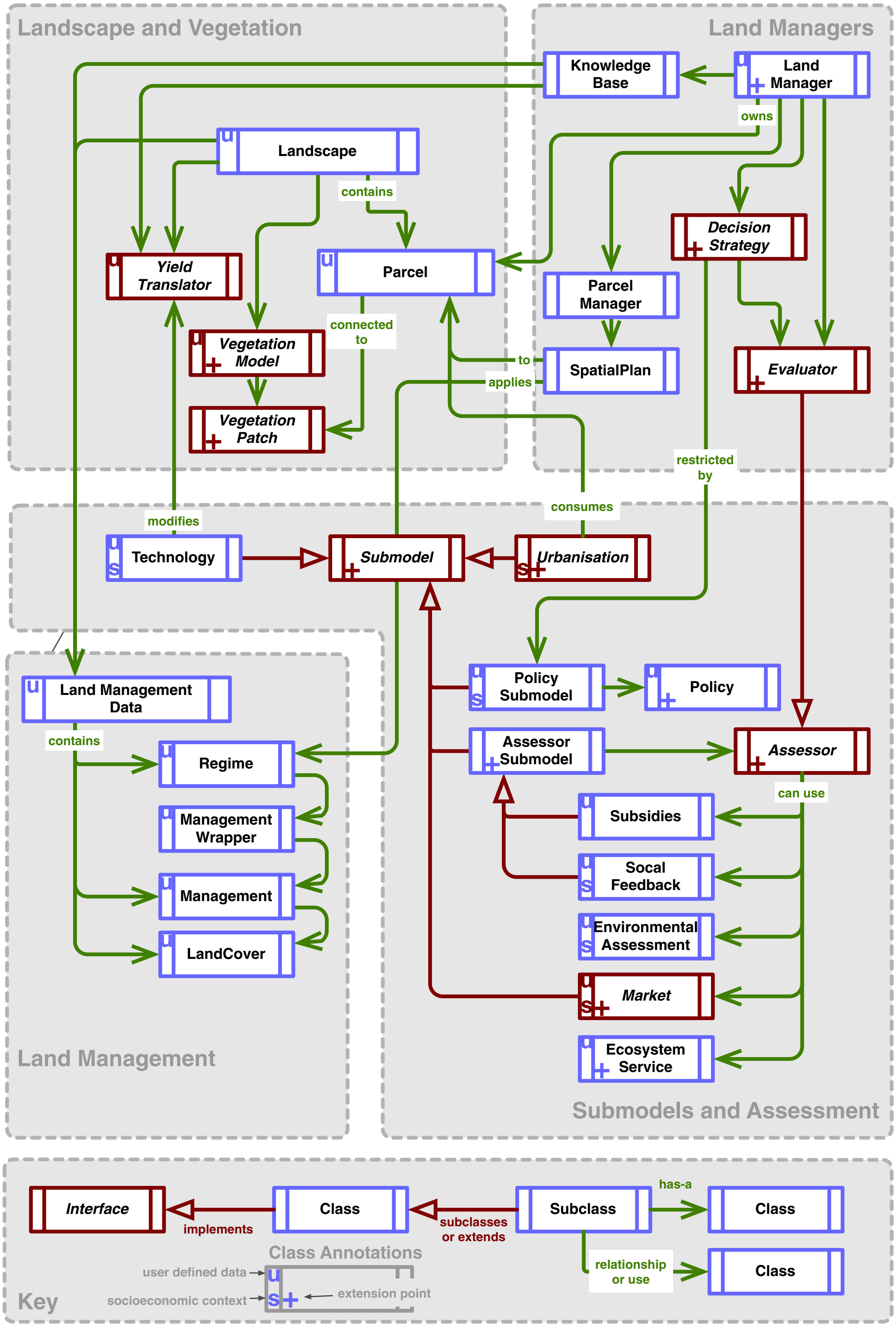


Fig. 1. Simplified class diagram for the major components of Aporia. Red lines indicate “is-a” and green lines indicate “has-a” relationships, which have been defined where appropriate. Intermediate classes have been omitted for clarity. Classes annotated with “+” indicate natural extension points, where user generated implementations can easily be substituted. Classes annotated with “u” indicate objects which are primarily containers for user data, and hence are configured at runtime. “s” denotes classes or interfaces which form part of the socioeconomic context of a model. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2. The framework

Throughout this paper, we distinguish between the conceptual model, which defines how real-world objects and processes are represented; the framework, which is a computational implementation of the conceptual model; and instantiations of the framework to model specific case studies. The Aporia framework was designed and constructed within the context of modelling the agricultural land use system, so the main components represent human land managers, the landscape they manage and management practices, and components to assess and influence human action (Fig. 1). A particular focus is placed on representing farm management decisions as these decisions impact on environmental, economic and social processes, and cannot easily be formalised using other modelling approaches (e.g., statistical and equation-based models; Parunak et al., 1998). While Aporia has the potential to manage land exchange and land-use transitions, this has not been extensively applied, and hence will not be described herein.

Conceptually, we model the landscape as a collection of Parcels, which represent the minimal spatial unit at which decisions are made, e.g., a farmer's field. Parcels are passive objects, which contain data to be modified by human action and natural processes. On each parcel, we represent: i) Land Cover (i.e., what physically resides on the surface of the Earth); ii) Managements; and iii) Regimes (Fig. 1 – Land management box. *Land Covers* represent a coarse-grained categorisation of what physically resides on the surface of the earth whether natural or artificial, for example “Wheat” or “Forest”. Although directed towards vegetation, Land Covers can be used for areas where no vegetation is present, such as artificial surfaces and bodies of water. *Managements* comprise a combination of a single Land Cover with additional management information, such as sowing and harvesting dates, levels of inputs and intended use (Fig. 2). *Regimes* are multi-year sequences of *Managements* applied to individual Parcels. Regimes are established a priori and land-use decisions are made based on the set of available Regimes. The Regime construct allows agents to make decisions over a simplified set of options that are agriculturally plausible and keep the richness of observed crop rotations. Regimes can be developed to represent similar rotations with or without winter cover; Land Covers that require several years of growth to reach maturity or produce outputs (e.g., *Miscanthus*); organic rotations with a very high time-to-return versus intensive rotations where the “head” crop can be grown more often; and implementation of agri-environmental schemes that include grass strips or natural buffers.

The actors currently represented in Aporia are those responsible for choosing how the land is managed on Parcels (LandManagers).

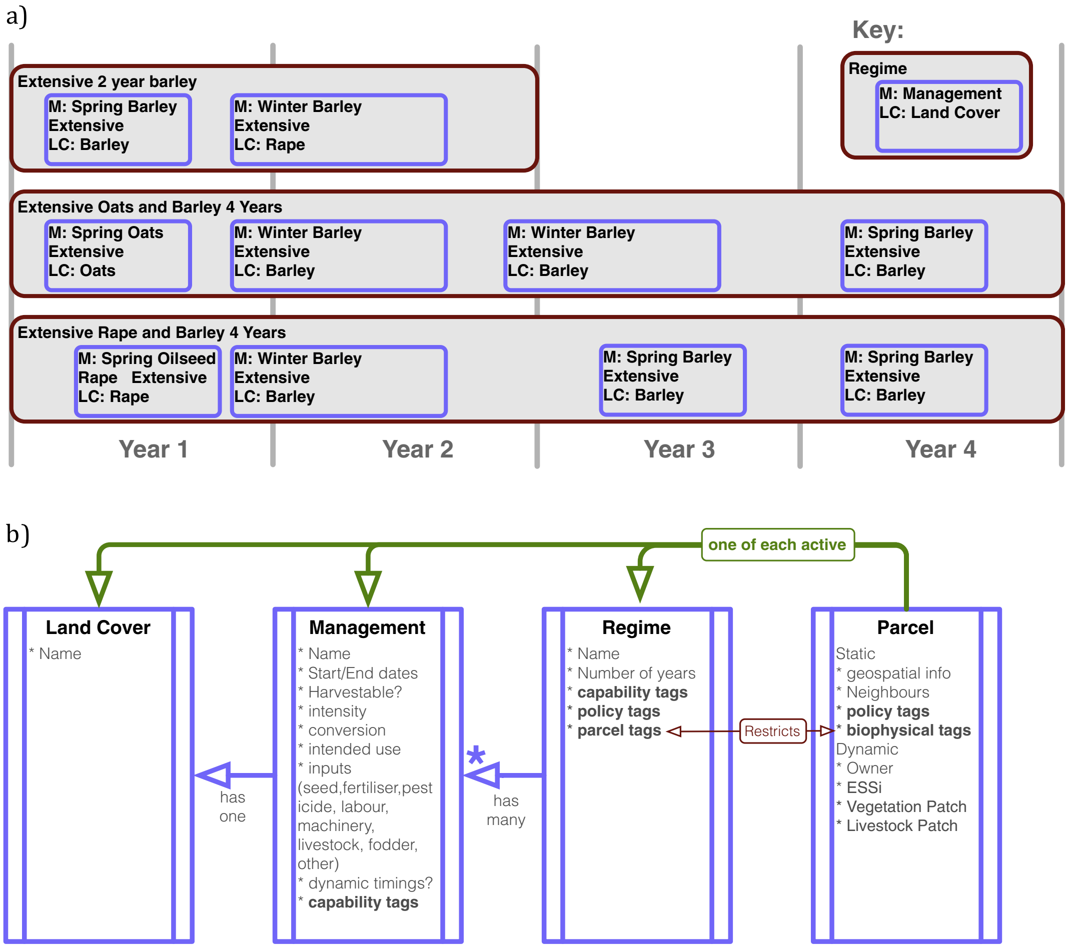


Fig. 2. Land management components and parcel data and temporal organisation. a) Example of three multi-year Regimes, illustrating different sowing and harvesting times. Each Regime (red box) contains several Managements (purple box), each of which has an associated Land Cover (LC). This is a small sample of the Regimes used in the Lunan case study (Section 3.2). b) Data associated with land management components and parcels, all of which is user definable. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

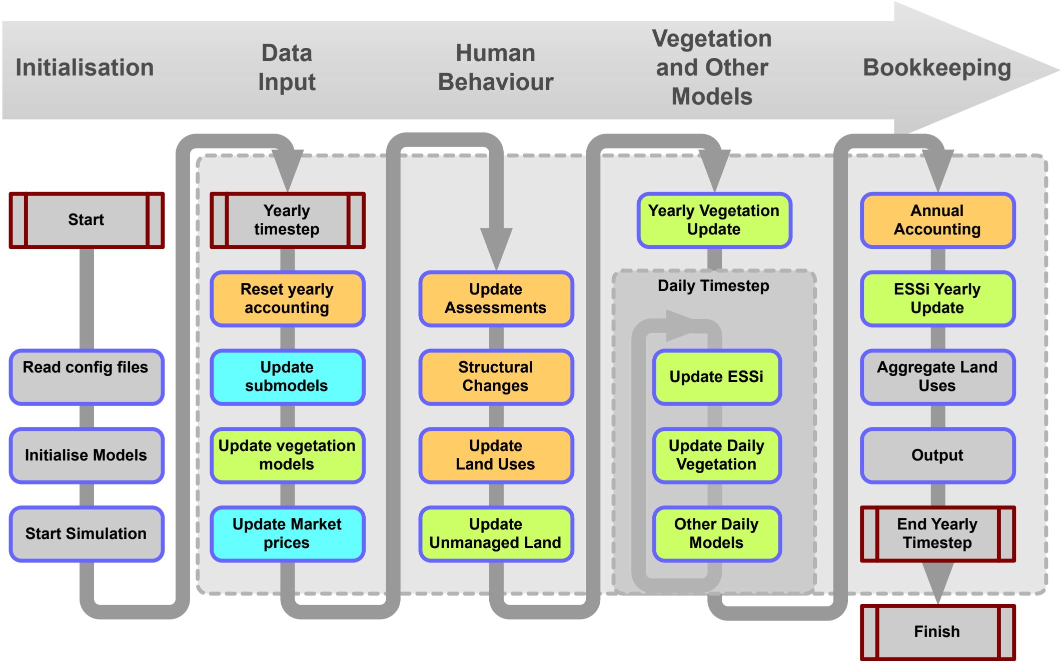


Fig. 3. Flowchart of Aporia in operation demonstrating the sequence of an instantiated model from setup of data inputs through to human and biophysical model actions and outputs.

In the presented applications these generally represent farmers, but they could also be other actors, e.g., foresters or nature conservation managers. Each land manager is responsible for choosing what Regimes and Managements should be applied to each parcel it owns. Since land management decisions are influenced by factors outside the immediate landscape, we include the concept of a *socio-economic context*, which aggregates all larger scale, exogenous processes, such as economics, policies and social attitudes towards land use practices (Section 2.5, and see Fig. 5 for examples of the conceptual linkages between this context and the decision-making of agents).

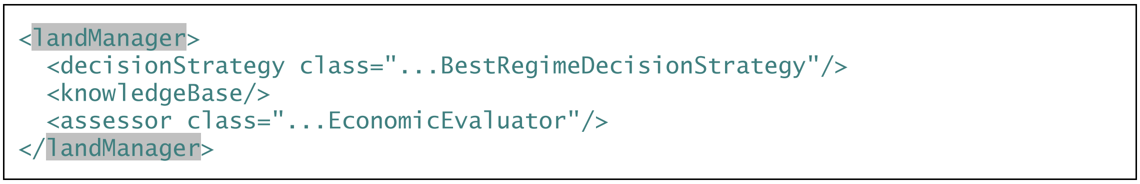
No constraints have been placed on the manner in which land managers make their decisions; however, to relate to human behaviour, we have worked mostly with the idea that a) land managers can assess potential actions according to a range of different evaluation criteria (Section 2.3), and use these assessments to make decisions (Section 2.4). Finally, the landscape is assessed in terms of ecosystem services, representing direct outputs of the land (e.g., crops harvested) as well as ecological benefits and impacts of land use decisions (Section 2.7).

2.1. Software architecture and setup

Aporia is written in Java using Repast Simphony (North et al., 2006, 2005) and SimpleXML, and is designed to be *modular*, *extensible*, *verifiable* and *transparent*, and takes a *declarative*, *compositional* approach to model configuration (e.g., Evert and Holzworth, 2005; Muetzelfeldt and Massheder, 2003). Rather than having a large configuration file which is read in and applied to different sections of a model, or alternatively “do this, then do that” *imperative* instructions, Aporia uses a collection of configuration files to specify a set of model components. The framework then creates objects that match this specification and places them in an environment in which to function (Fig. 3). To give an indication of which components are user extensible, Fig. 1 annotates classes that are natural extension points for additional user functionality (see Appendix 1 for additional expansion options). There is also an absolute separation between code and data, so that most model components are defined in configuration files rather than being hard coded.

In practice, this means that the framework takes a declarative approach to configuration (Section 2.1) and a simulation is defined through XML files specifying Java classes to be used and the parameters they should be given. Aporia uses this specification to instantiate objects that are run autonomously using the declarative annotations provided by Repast Simphony (North et al., 2007). While there are many reasons to adopt a declarative approach, the main driver here is flexibility. The framework embodies the conceptual model, but the actual configuration of the components is user-controlled by: i) parameterising existing components; ii) composing existing components; or iii) creating custom components that implement standard interfaces.

An illustration of the use of XML snippets to define an agent and its behaviour, by specifying Java classes to be used, and their parameters is provided in Listing 1. One instantiation of a Land-Manager agent could simply involve establishing a decision-making strategy that is based on a goal to maximise economic returns (Listing 1a). To create an agent that attempts to balance a set of different criteria, asks its neighbours about expected yields, and selects randomly from several of the best Regimes, the XML text in Listing 1b is used. The differences between these two agent configurations illustrate three key features supported by the declarative approach to configuration. First, the overall configuration of the two agents is of quite different complexity – there is no need to have a single type of specification for all agents, even if they perform the same role. Second, the re-use and nesting of sub-components by defining general approaches (i.e., weighted averaging) that accept any class that provides certain functionality by implementing a common *interface* (i.e., the different assessors used here). Third, a user can create a component that implements the same interface, and asks the model to use their custom component by-creating code to replace sections of model behaviour.



```
<landManager>
  <decisionStrategy class="...BestRegimeDecisionStrategy"/>
  <knowledgeBase/>
  <assessor class="...EconomicEvaluator"/>
</landManager>
```

a) Simple specification of a land manager, who works in purely economic terms

```
<landManager>
   <decisionStrategy class="...BestFromNDecisionStrategy" num=”6”/>
   <knowledgeBase>
      <yieldPredictor class="mycode.CompareNeighboursPredictor"/>
      <pricePredictor class="mycode.RegressionPredictor"/>
   </knowledgeBase>
   <evaluator class="...MulticriteriaEvaluator">
      <assessor class="...AssessorWrapper" weight="0.7">
         <assessor class="...EconomicEvaluator" discountFactor="0.3" />
      </assessor>
      <assessor class="...AssessorWrapper" weight="0.2">
         <assessor class="...SocialEvaluator"/>
      </assessor>
      <assessor class="...AssessorWrapper" weight="0.1">
         <assessor class="...EnvironmentalEvaluator"/>
      </assessor>
   </evaluator>
</landManager>
```

b) Extension of the land manager to include i) custom predictors for yields and prices; ii) weighting of economic gain against social and environmental effects

Listing 1: Alternative declarations of land managers. Class attributes define Java classes to be instantiated. Classes starting "aporia." are part of Aporia, while those starting "mycode." are (fictional) user extensions.

This makes the framework *modular*, as components can be brought in where necessary, and *extensible*, as existing implementations of components of the conceptual model can be replaced with new code. Section 2.9 details some typical modifications, and the time and expertise required to create them, as well as outlining the steps required to create new implementations of certain components. The modularity of the code also supports the use of Unit Testing to improve *verifiability*.

The main components (i.e., the landscape, land management data, agent typology and submodels) are specified in individual XML files, which are linked to from a main scenario file. In addition to the XML setup, standard data formats (CSV files and shapefiles) are used. The system is designed so that most non-XML configuration can be setup in a single spreadsheet, and exported as needed. Fig. 8 in Appendix 6 gives an example of a typical file structure.

2.2. Land management

Land-management decisions are complex, and can become difficult to represent since many management decisions require specification beyond simply which crops are optimal for a location (Chabrier et al., 2007). For example, multi-year constraints define which crops may be grown in sequence (Castellazi et al., 2008; Dogliotti et al., 2003) and affect spatio-temporal patterns of land use; and the style of management (e.g., intensive versus extensive, spring sown versus winter sown) affects the levels of inputs used, sowing and harvest dates, and the yields obtained. These detailed choices have an ecological impact – for example: sowing and harvesting dates and the presence of winter cover affect erosion; the uptake of agri-environmental schemes can reduce ecological impacts of farming (Primdahl et al., 2003); the use of pesticides and machinery in high-intensity farming can affect avian populations (Topping and Odderskær, 2004).

Aporia provides several conceptual components to represent complex programmes of management while remaining tractable from the point of view of agent decision-making. These components include: Land Cover, Managements and Regimes as “land management components”, which are applied to parcels. Each component has unique attributes and stores unique data (Fig. 2); however, components are linked and can share information. Furthermore, all of these attributes can be configured by the user to represent specific land management practices for a given study area.

2.3. Regime evaluation: assessors and predictors

Human and natural processes typically occur at different spatio-temporal scales and connecting their computational representations can be challenging. Frequently, natural systems are represented at high temporal resolutions and coarse spatial resolutions while the anthropogenic component of coupled natural-human systems is often represented at coarse temporal resolutions and relatively fine spatial resolutions (e.g., 30 m; for a spatial-temporal comparison see Evans et al. (2013)). To reconcile the spatio-temporal mismatch between different representations of human and natural processes, we have introduced Assessors as a conceptual component, to abstract the idea of quantitative assessment of a course of action, so that values can be attached to components at whatever level is appropriate, and translated for application at other spatiotemporal scales; for example, a score can be attached to a LandCover, which is then integrated into assessments of any Regimes that use that LandCover, with appropriate scalings for area and time of application.

Assessors are used in several places in the framework, including human decision-making and ecosystem service indicator calculation. Conceptually, the requirement for an Assessor is that it can be used to calculate a score for an action. The score can be a measurement of any quantity of interest that can be measured on an ordinal, interval, or ratio scale, e.g., money, landscape aesthetics, social feedback. The action can be at: i) different temporal scales, i.e., Land Covers, Managements or Regimes; ii) at different spatial scales: entirely abstract, over a single Parcel, or across multiple Parcels. The Assessor mechanism is responsible for converting between the scales in a manner appropriate to the given score.

There are several ways in which users can customise how assessment is carried out (Fig. 4). First, static scores can be set for any land management component (using a *DefaultAssessor* and providing a CSV or XML file to assign scores). Second, multiple *Assessors* can be combined together to carry out multi-criteria evaluation, using either linear sums or more complex functions (see also Equations (5) and (6)). Third, custom Assessors can be written by implementing one or more methods that translate land use actions into scores (see Appendix 3 for more detail). As an indication of what is possible, Appendix 4 describes the Ecosystem Services that have been implemented as Assessors, and the effort required.

A second example is the assessment of expected economic return, which is a complex operation that includes: i) labour, materials, and other factors influencing the cost of implementing a Regime; ii) variable weather/climate affecting yields; and iii) changing socio-economic conditions such as crop prices and demand. Furthermore, heterogeneity in the biophysical conditions of the landscape further complicates the assessment, since the factors of production (i.e., land and its characteristics) are unique for each farmer. The *EconomicAssessor* hence uses a predictor that calculates an expected value based on projected yields and prices, either from personal experience or observation.

The concept of *evaluating* a given action choice using the assessor and predictor structure forms the basis for many calculations in Aporia, including but not limited to economic gain, subsidy levels, ecosystem service indicators (ESSi), and social and environmental feedback. By handling the spatio-temporal mismatch in a single place, the likelihood of coding errors by the modeller is reduced, since the complexity can be contained within a few well-tested components, and researchers are free to concentrate on implementing the algorithms of interest.

2.4. Agents and decision-making

Each agent has a Decision Strategy, which underpins the choice of land use actions each year. Conceptually, this is modelled as a generic decision procedure, whereby at the end of each year, each agent observes the land it is managing, and assigns a new Regime to each parcel whose Regime has finished. Formally, given a set of Managements M, and a set of Regimes R (as defined above), a Regime r is a sequence of Managements:

$$r = (m_1, \ldots, m_n) \quad (m_{1,\ldots,n} \in M) \tag{1}$$

Each agent has a Land Use Plan P, which is an assignment of Regimes (r_i) to Parcels (p_i), each with an index (y_i) representing the year of the Regime that is currently being applied:

$$P = \left\{ \langle p_1, r_1, y_1 \rangle, \ldots, \langle p_j, r_j, y_j \rangle, \langle p_k, -, - \rangle, \ldots, \langle p_n, -, - \rangle \right\} \\ (r_1, \ldots, r_j \in R, r_k, \ldots, r_n = \varnothing) \tag{2}$$

Some parcels will not have a Regime assigned, as the previous Regime has not finished. A decision procedure is a function from a current plan P and a knowledge base K to a new plan P'; it assigns new Regimes to available parcels, while all other Regimes move on to their next year:

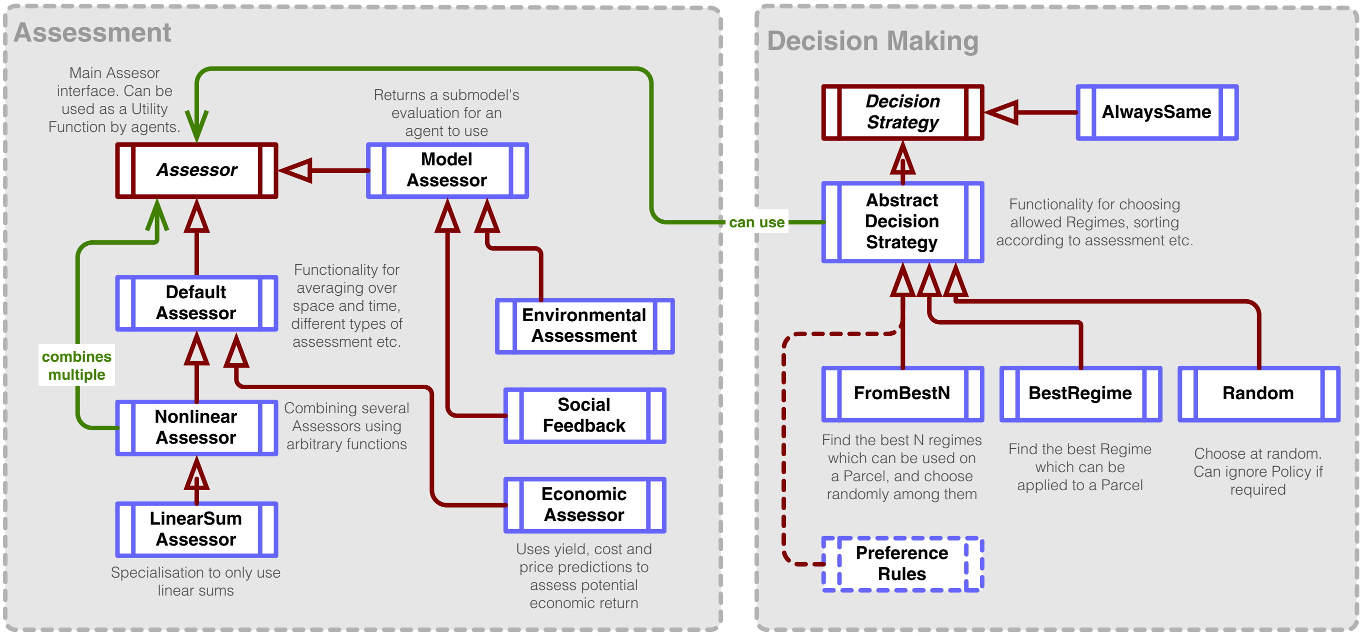


Fig. 4. Class diagram of assessor and decision-making components of Aporia. The *DefaultAssessor* provides basic values for Land Covers, Managements and Regimes. The *(Non) LinearAssessors* can aggregate other Assessors to do multi-criteria evaluation. The *EconomicAssessor* is an example of a special function assessor which evaluates the costs of carrying out a course of action. Various submodels use assessors to evaluate environmental impact or social feedback. The default human decision making process is a combination of an Assessor to define preferences for different actions and a Decision Strategy which uses the evaluation to choose between courses of action.

$$D(P,K) \rightarrow P' = \left\{ \langle p_1, r_1, y_1 + 1 \rangle, \ldots, \langle p_j, r_j, y_j + 1 \rangle, \langle p_k, r_k, 1 \rangle, \ldots, \langle p_n, r_n, 1 \rangle \right\} \tag{3}$$

Again, this is a conceptual component, and the user is free to provide alternative approaches. Any function can be used which takes the agent's knowledge (K in Equation (3)) along with a current assignment of Regimes to Parcels and produces a plan for the next year. An intermediate level of specification is provided where $D(P,K)$ can be specified in terms of: i) a preference or utility function which evaluates Regimes based on their expected utility at a given point in time; ii) a decision procedure which works over evaluated Regimes to choose the next action. Section 2.4.1 outlines a utility based approach for Regime evaluation, but user can create versions of the preference function or the decision procedure as a way of defining new behaviour.

2.4.1. *Multi-criteria preferences*

Many different factors influence the evaluation of a Regime; taking a utility based approach facilitates the inclusion of non-economic factors that incorporate broader concepts such as "well being" (Edwards-Jones, 2006). Hence, much of the modelling with Aporia has used multi-criteria utility functions to take into account a range of factors e.g., social feedback and perception of environmental impact, and go beyond an economic-only conception of utility. Using the Assessment modules (Section 2.3), agents calculate a score for each factor in their utility function. Given a set of n available assessors, the three most commonly used in Aporia are:

$$E(R) = E_{econon}(R) \quad \text{Pure economic evaluation; See Listing 1 : a for example configuration} \tag{4}$$

$$E(R) = \frac{1}{n}\sum_{i=1}^{n} \lambda_i E_r(R) \quad \text{Weighted sum of individual factor scores using a } \textit{LinearSumAssessor}\text{; See Listing 1 : b for example configuration} \tag{5}$$

$$E(R) = \frac{1}{n}\sum_{i=1}^{n} \lambda_i + \delta_+(E_i(R)) + \lambda_i - \delta_-(E_i(R)) \quad \text{Non} - \text{linear sum, with different slopes for positive and negative values, and an offset. } \delta_+(x) = 0 \quad \text{if } x < 0 \text{ and } x \text{ otherwise, and } \delta_-(x) \text{ is the complementary function}^{1} \tag{6}$$

One or more assessors are used to calculate the utility derived from a Regime applied to a parcel of land. Assessors provided by Aporia may be configured and chained together through the XML configuration files without any programming. Additional functions can be added or created by extending those provided or creating new assessors with code.

2.4.2. *Selection of regimes after scoring*

Given a set of Regimes with utility scores, arguably the simplest decision procedure is to apply the Regime with the highest score (i.e., BestRegime decision strategy) to all Parcels that need a new Regime. However, in some cases, the Policy submodel may forbid the best Regime on some or all Parcels, and some Regimes may not be applicable to certain Parcels (for example, high nitrate regimes near watercourses). Hence, the BestRegime decision strategy applies the best Regime that is allowed by both policy constraints and parcel characteristics. As an extension to this, the PickFromBestN strategy selects the N best Regimes that are allowed on the given Parcel and selects one at random. To allow comparative experiments, a Random decision strategy assigns a random Regime to each parcel, and an AlwaysSame strategy maintains the same Regime indefinitely.

2.4.3. *Agent populations and heterogeneity*

To represent Land Manager heterogeneity, populations of land managers can be constructed from typologies—conceptual and multidimensional classifications (Bailey, 1994)—which simplify the diversity present in a real population to allow insight into broad categories of behaviour (Valbuena et al., 2008). By limiting the number of agents to parameterise (from every individual in the population to typically four or five types), the work required to specify behavioural properties becomes tractable. In Aporia, the typology defines all of the behavioural aspects of an agent: the type of decision-making used, and (if a utility based approach is taken) the form of utility function, the assessors used in it, and the values of the weights used (Section 2.4.1).

There are three stages to instantiating a population of land managers: i) the landscape is partitioned into farms, either based on existing spatial ownership data, or on histograms of farm size; ii) next, an agent is constructed for each farm by sampling the agent typology; and iii) individual heterogeneity can be increased by adding historic land use data that can be used to give an agent increased knowledge that affect its decision-making process. In contrast, the historical data can also be used to constrain agent decisions to those historically observed.

2.4.4. *Knowledge and learning*

Each agent has a "Knowledge Base" (represented by K in Equation (3)). This contains all of the information about land uses and other factors that informs an agent's decision. The knowledge base of a Land Manager agent is initiated with a list of Regimes defined by the model user. The knowledge base is supplemented over time by experiential information about Regime success, which is related to predicted (i.e., using historical experiences, neighbours yields, or case-study averages) versus acquired yields, and changes in economic, environmental, and social feedback indicators underlying Regime evaluations. Agent decisions can be affected by the changing socio-economic conditions, the actions of neighbours, biophysical variations and yearly fluctuations in yields and prices, leading to agents being adaptive, and producing spatio-temporal heterogeneity in their decisions and subsequent impacts on ecosystem services.

2.5. *Socio-economic context*

Scenario analysis plays an increasingly important role in the application of land use change models, especially where the

---

[1] This is a custom equation designed for working with specific data from conjoint analysis.

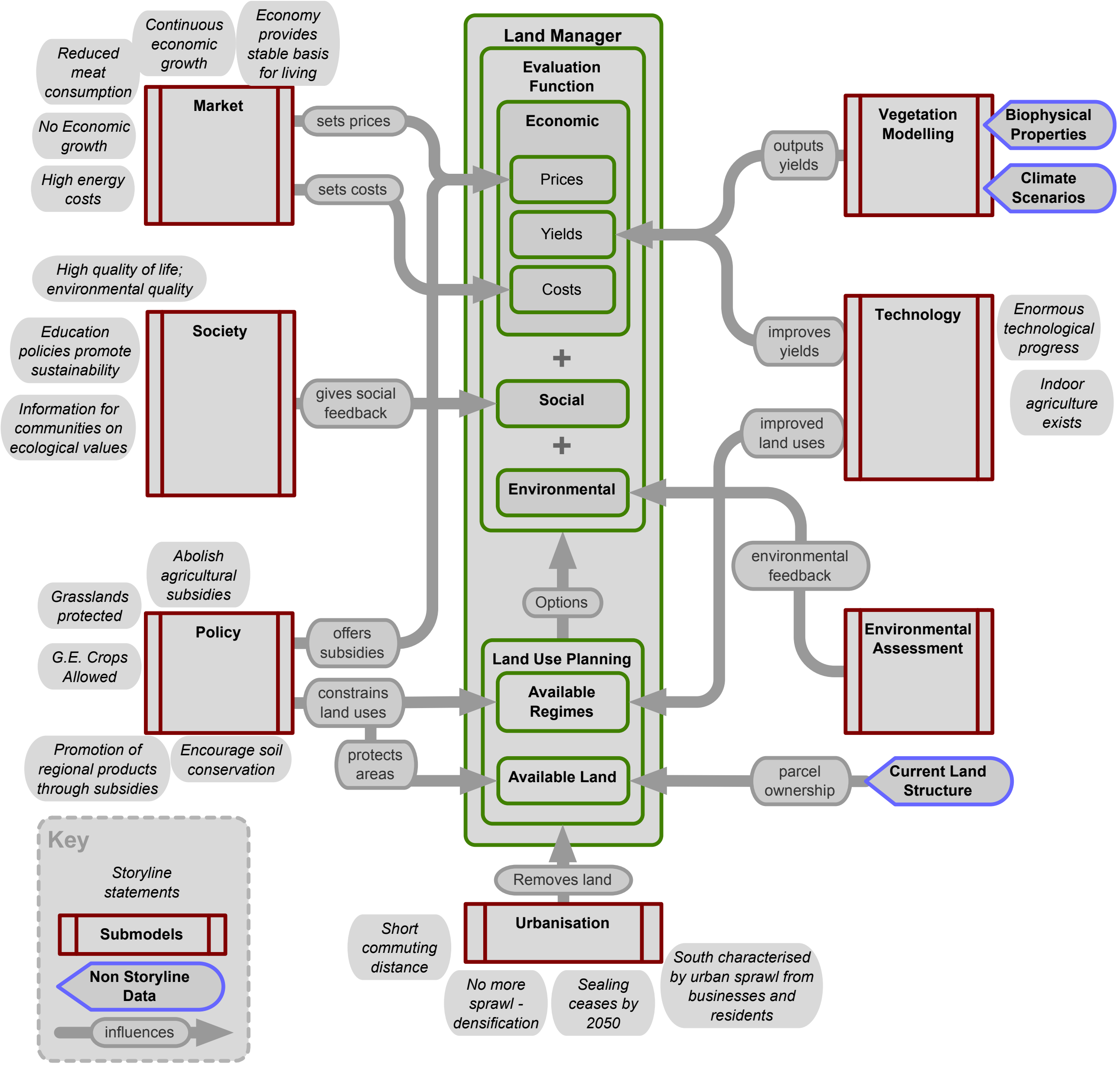


Fig. 5. Socio-economic context for decision making; mechanisms of effect on agent behaviour. This shows the conceptual linkages from storyline statement, through parameterisation of submodels, and then the effects they have on a (typical) agent's decision-making process. The final mechanisms of action take the form of restricting the options available, or altering the agent's evaluation of a given course of action.

impacts of that change are considered (Abildtrup et al., 2006; Bohunovsky et al., 2010). Often, these scenarios take the form of qualitative narratives describing the trajectories of large scale drivers (Rounsevell and Metzger, 2010; Spangenberg, 2007). Conceptually, in order to enable the use of narrative scenarios, a socio-economic context is implemented for a given simulation; this is done using a selection of submodels.[2] Fig. 5 shows a conceptual overview of how this contextualisation occurs: starting at the outside of the diagram, narrative statements and quantitative outputs are used to define parameters for each submodel. The diagram shows the mechanisms of action by which the submodels alter both: i) the range of possibilities available to each agent when making decisions; and ii) the attractiveness of any given possibility.

The *Policy* submodel runs a set of user provided policies, which are either *subsidies* or *restrictions*, each of which can be switched on or off over time to simulate policy change. *Subsidies* use the Assessor framework from Section 2.3 to give a monetary amount for any given course of action; users can construct their own subsidies, and spatial effects can be taken into account. *Restrictions* are functional controls placed on an Agent, Parcel or Regime that constrain decisions and acceptable land managements, e.g., forbidding high nitrate Regimes close to rivers.

The *Market* provides time varying prices and costs for agricultural inputs and outputs (seed, fertiliser etc.), and supports all economic calculations, as well as reporting quantities bought and

---

[2] While this may seem like an overly grand term for what are mostly simple components, the implication is that they can be replaced with more developed versions. For example several different urbanisation models have been built, with different methods of allocating new urban land, and the simple market can be replaced by one which simulates market fluctuations using autoregressive noise.

sold. As well as setting basic price curves, users can replace the price mechanism with others (e.g., to add noise, or using an algorithmic process to generate prices).

Using the *Technology* module, users can specify: i) changes in yield over time for various Managements; and ii) the timings for introduction of novel Regimes in the course of a run.

The *Social* and *Environmental* modules are typical examples of Assessment Submodels, which produce a numeric score to be used in agent decision-making. Here, they are used to translate storyline statements into quantitative values, to give agents feedback about the social and environmental aspects of their behaviour.

Urbanisation models convert land to artificial surfaces. Several mechanisms exist to do this and the following two are provided with the Aporia framework: specifying an area of land to be annually converted and (optionally) a ranking for conversion, and specifying a year for conversion for a specific parcel.

When setting up simulations, the socio-economic context provides an organising principle to combine both qualitative and quantitative information from scenario storylines and model output. Although a full discussion is outside the scope of this paper, in general: i) statements from storylines which *should* be modelled are extracted, and attached to the relevant submodels; ii) any statements which cannot be modelled are noted; iii) scenario statements are converted into qualitative and then quantitative parameter settings; iv) any other qualitative information which can be included is brought in; v) any undefined submodel parameters are filled in with reference to the scenario storylines. Taken together, these submodels give a rich, time-varying context that can be used to model a range of real-world situations.

2.6. Vegetation modelling

Vegetation modelling is used in Aporia to simulate changing crop yields for farmers, and additionally to underpin the modelling of ESS (Section 2.7). Due to the importance and complexity of vegetation modelling, three different vegetation models have been implemented for user adoption depending on available data and intended use. In each case, the vegetation model creates a Vegetation Patch for each Parcel on which vegetation growth and potentially other biophysical processes are simulated.

The *DefaultVegetationModel* is a lookup table for user-supplied data on the height and biomass of Land Covers or Managements over time. This *Default* model does not include any response to climate change or natural variation, and is typically setup based on case study yield data. The *VegetationMetaModel* extends the default model with yearly yield data to represent yield change over time. This has been used to carry out loose coupling with dynamic vegetation models (DVMs; i.e., LPJ-GUESS (Sitch et al., 2003), CARAIB (Warnant et al., 1994)) by translating DVM output into yearly yields for the case study. To add a new DVM, the only configuration the user must supply is describing the format of the DVM output and mapping from the DVM vegetation classes to Aporia Land Covers or Managements. Finally, an *LPJVegetationModel* tightly integrates a version of LPJ-GUESS that simulates managed land as well as natural vegetation (Lindeskog et al., 2013). While the details of ABM and DVM coupling are beyond the scope of this paper,[3] tight coupling takes considerable technical expertise, but it enables dynamic representations of ESSi over time and increased representation of heterogeneity across the landscape.

2.7. Ecosystem services

Implementing the ecosystem service concept (Carpenter et al., 2009; Hancock, 2010)—analysing the benefits which humans derive from the landscape—means Aporia can be used to evaluate the impact of changes in land management and land use through changes in the (simulated) provision of ecosystem services (Rounsevell et al., 2012a,b, 2010). Typically, ecosystem services are modelled using indicators, each of which relates to a single benefit, such as the provision of fresh water, food provision, flood prevention, and carbon storage. Aporia follows this approach, and has a user defined collection of indicators for ESSi of interest. Appendix 4 gives a list of the indicators which have been used, and the work necessary to create them. In increasing order of complexity, ESSi simulation is done by: i) assigning a simple score for a given ESSi using the Assessor framework, which is a quick and easy way to setup a lookup table, where values for each Regime or Management are known and do not change over time, or are difficult to compute; ii) providing a conversion rate from harvests into ESSi levels, i.e., tonnes to calories for food production; iii) extracting data from coupled models (e.g., carbon sequestration from a DVM or populations of charismatic species from an IBM); and iv) setting up custom calculations, e.g., for landscape aesthetics, where a set of parcel-based statistics is computed. ESSi values are maintained per parcel and recorded over time, so that maps and trajectories of service provision may be produced.

2.8. Outputs

At runtime, a range of visual displays provides transparency into model behaviour, e.g., tables showing market prices and costs, maps of spatial variables such as current management practices or ESSi. Each Land Manager can have its own display, which shows the current state of its knowledge and decision-making.

A highly configurable output section can record almost all aspects of internal state as CSV files, with the option of outputting Parcel level data as shapefiles. To give some examples, each year Aporia can output: i) Parcel data containing information on ownership, current land management practices (Land Cover, Regime and Management, intensity, and a configurable system of tags), ecosystem service indicator values etc.; ii) Land Manager data with financial status, utility, average ESSi levels, subsidies received and so on; iii) Market data with prices and quantities of all inputs and outputs bought and sold; iv) Subsidies data with the total amount of each subsidy paid out etc.; v) land management data with the area of each Management started or harvested each year, how much it cost, average yield, sowing and planting times etc.

Landscape features can also be serialised either to a database or over a network connection, to allow alternative visualisations and the playback of previous runs.

2.9. Example model development

To illustrate how the Aporia framework may be used, we present several initial stages of development for a simple stylised model.[4] The initial setup uses a hypothetical agricultural landscape consisting of 100 parcels with 60 land owners. The land cover options

---

[3] SWIG (www.swig.org) is used to wrap the compiled C++ code into a library that can be called from Java. Multiple copies of LPJ-ml are created inside Aporia, with yearly applications of land use and daily updates on vegetation state. In addition to crop yield, ecosystem service indicators are extracted from the DVM (e.g. carbon storage and water flows). This allows full biophysical heterogeneity among parcels where data supports it.

[4] These files are available as a project at bitbucket.org/mo_seph/aporia-tutorial-data and all XML files described in this paper are created from Aporia XML templates available with the framework download.

are confined to three crops and grassland and four land management regimes. Two single year regimes are created, one for each of Crop 1 and Crop 2. A two-year regime is created for Crop 3 which alternates with grassland. A fourth regime extends over seven years and combines Crops 2 and 3 with periods of grassland. All crops produce the same yield, have a market value of C1 > C2 == C3, and all costs are ignored. Through the aggregation of individual landowners making heterogeneous decisions across the landscape, the model simulates aggregate changes in land use, land cover, crop yields, farm livelihoods, and ecological function.

The construction of this stylised model involves creating 8 XML files adapted from templates provided with Aporia, a shapefile for the landscape, and 4 CSV files supplying values for vegetation and market parameters. This model implemented with Aporia represents a simple starting point for representing the agricultural land system and performing computational experiments. To illustrate the types of experiments that can be performed with this simple configuration we outline four scenarios in which framework components are utilised to systematically increase the complexity of the land system being represented. We use the term scenario for each model configuration because these components are configured in our XML scenario file and typically once a specific model is configured then the external drivers and model parameters are manipulated to create scenarios.

StaticEconomic (SE) Scenario. This scenario comprises a single type of land manager who is only concerned with economic gain and chooses the Regime with the highest economic return at every decision point.

StaticMulticriteria (SM) Scenario. The SE scenario is augmented by introducing two additional types of land managers, who make select a regime based on economic return, environmental sustainability, and social. One of these new land manager agents weighs social feedback more heavily than environmental sustainability.

To construct the SM scenario we i) add an XML and CSV file to create a “Social” assessment of the different land uses, with better scores for Crop2 and Crop3 than Crop1; ii) repeat the same process as (i) for environmental assessment; iii) create two new XML files to represent the new agent types with appropriate preference weights; iv) copy the initial typology file and add in the two new agent types along with their proportions in the population; v) copy the original scenario file and link it to the new agent typology.

Global Prices (GP) Scenario. Building on the previous two scenarios (SE and SM), we introduce global crop prices. These prices affect regime selection and the economic return of land manager agent decision making. In this scenario Crop1 slightly decreases in value, while Crop2 and Crop3 increase substantially in value over the duration of a model run. The incorporation of external drivers affecting land manager decisions (like prices) is achieved by creating a new CSV file which defines prices for each crops at different time points. Again, the scenario file is duplicated and pointed at the new global prices CSV file.

Global Subsidies (GS) Scenario. In the final scenario, a set of subsidies are introduced for the long running Regime 4, starting in 2010. This involves adding i) a CSV file to define subsidy levels for each regime; ii) a CSV file giving relative subsidy levels over time and iii) an XML file which used these files to create an Assessor which links to the Market to give subsidy levels.

Our systematic incorporation of model components and external drivers demonstrate the utility of the framework and how they may affect the land use dynamics within a simulation (Fig. 6). In the SE scenario, the economic land managers quickly settle on growing Crop1 everywhere and verify that the model is performing as we would expect. The different types of land manager agents in the SM scenario produce a mixture of crops that is still dominated by Crop1. In the GP scenario the decreasing price of Crop1 causes land managers to shift to other land use options, and finally in the GS scenario, the increased subsidy for Regime 4 leads to an increase in the amount of grassland.

At all stages of the model development, it was possible to see which Regimes each land manager was carrying out; how they were calculating their preference scores; the levels of subsidy available; and the amount of each crop sold on the market and its price. This development process is similar to the process for creating a more detailed, empirically grounded model – the difference being the amount of data that is incorporated and the addition of more components to manage the new details. For reference. Fig. 8 in Appendix 5 gives an overview of the configuration files used in an empirically grounded case-study application.

2.10. Extension to new situations

The previous section described how the existing components can be combined to create increasingly detailed models. A relevant avenue to explore is what happens when the existing components are not sufficient. Aporia has been designed to be extensible through the use of interfaces and declarative configuration as mentioned in Section 2.1. When developing a new simulation, four general techniques may be used: *composition*, *substitution*, *addition* and *modification*. These are as follows:

- In many situations, complex behaviour can be specified by parameterising and *composing* existing components (as in the previous section). This covers situations such as adding new

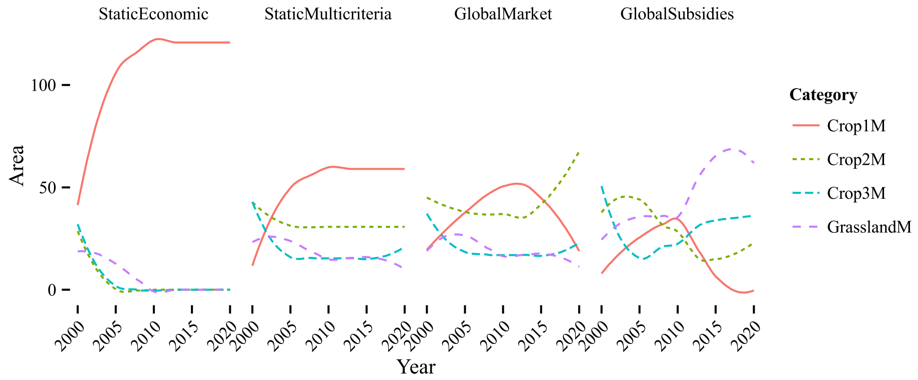


Fig. 6. Land use areas under different stylised scenarios. StaticEconomic only uses economic agents, and fixed prices. StaticMulticriteria has three agent types each with an emphasis on social, economic or environmental values. DynamicMarket extends this to include price changes, while DynamicSubsidies adds in a subsidy for the longer, mixed Regime.

Table 1
Example modifications, depth of change and required effort/skill levels.

Modification	Type	Baseline time and skill
Expanding land use types in a model	Configuration only	1/2 day
Support for simple spatial policies	Configuration and core code modification	1 day, expert knowledge
Creating an alternative Urban model	Configuration and additional code	1 day (or more depending on complexity of model), programming background
Adding an ecosystem service covered by current equations	Configuration only	1/2 day (assuming input data is available)
Adding a new ecosystem service indicator	Configuration and additional code	1 day or more (depending on complexity of ESSi code)
New preference function	Additional code	1 day, depending on complexity
New decision procedure	Additional code	1 day or more, depending on complexity

Land Covers, Managements and Regimes; adding new types of agents and describing their preferences and decision procedures; setting up spatial data; defining policies and biophysical constraints; creating some ESSi and so on.

- Sometimes, a new version of a component needs to be created and *substituted* for the existing implementation. In this case, no modification to the core code is needed, and the new functionality can be made available as an option to others. New versions of many components can be created, such as: preference functions, decision procedures, market pricing models, ecosystem service indicators, and vegetation models.
- Occasionally, a new kind of component needs to be created and *added* to the existing structures. Examples of this approach include adding an urbanisation component and an individual based skylark model (both discussed below). Connectivity between components here is supported by Parcels, which can contain extra information produced by submodels, and are used to provide a “public-facing”, always valid view onto data. Facilities exist to assist in creating companion objects for each Parcel so that submodels can store model specific data and carry out calculations.
- Finally, in rare cases, *modifications* need to be made to the existing framework. This requires someone with expert programming knowledge. In most cases, these types of changes have consisted of implementing planned functionality, such as support for spatial policy in decision-making procedures, or parcel neighbourhoods for modelling landscape connectivity.

Table 1 gives some example modifications, and the expertise required to carry them out. In general, these represent the time taken to implement a simple version of the required feature, and hence most of the time taken is to integrate the new code with the existing model, including configuration and testing. They can hence be taken as a baseline, and additional time would be required for implementing complex algorithms.

Three illustrative cases are outlined in more detail here:

- As part of a case study, an urban model was required. This model would remove Parcels from Land Managers, and convert them to artificial surfaces. Due to the declarative configuration, no changes to the core code were required: it was sufficient to create a new class, which was loaded alongside the existing models, and interacted with the land exchange components already present. An initial urban submodel took approximately one day's coding effort, and did not require changes to the core code. It was also possible to abstract much of the functionality required, into a stub implementation. Given this stub, a scientist unfamiliar with the codebase created a more advanced urban model in around a day, and about two screens of code.
- As part of the same case study, a more flexible Policy model was required, which could apply spatial limits on the application of Regimes to Parcels – for example protecting existing grassland, or limiting high nitrate Regimes around watercourses. This required an update of the Decision Processes used by Land Managers to pay proper attention to spatial policies, and a modification of the Policy model to load and update the new policies. Again, this took less than a day to implement, and now provides base functionality for the easy implementation of new types of spatially explicit policy measures.
- One aspect of the Lunan case study (Section 3.2) involved coupling an individual based model (IBM) of skylarks (*Alauda arvensis*) with the human management components presented here (Guillem, 2012). In this case, no internal code modifications were required; the skylark model set up a corresponding object for each parcel, which contained a population of skylark “agents” (nests, eggs, various stages of development); the existing parcel information was used by the skylark model to calculate habitat suitability and carrying capacity; and the skylark population was added to the parcel's attributes so that it could be used in other calculations-in this case, as the basis of an ESSi.

3. Case study applications

We present two example applications of Aporia to illustrate the types of data used, models that can be instantiated, and outputs that can be produced. Each model implementation was led and configured by case-study experts to assess the impacts of land-management decisions on agricultural and economic outputs as well as the provision of ecosystem services. The two case studies have differing sets of data used for parameterisation and establish policy and socio-economic contexts, hence a different configuration of framework components and outputs were created for each application (Table 2).

3.1. Aarau valley

The Aarau valley comprised an exemplar study area in the Ecochange project[5] for the assessment of ecosystem changes in Europe. It is a peri-urban area of 99 km^2, located in the northern part of Switzerland, in proximity to the urban centres of Zurich and Basel. The mosaic of land uses and land covers that is present in the area creates a unique landscape of high aesthetic and ecological value. The dominance of production-led subsidies and the subsequent intensification of farming practices had led to public concerns about the environmental impact of agriculture, as well as food surpluses and high production costs (Curry and Stucki, 1997). As a result, the Swiss agricultural sector was restructured around a multi-functional model (Cretegny, 2001) that placed emphasis on the importance of the environmentally-friendly management of farmland. Despite the establishment of strict agricultural regulatory mechanisms, land use decisions, such as the choice between intensive, extensive or organic farming systems, remain dependent on the individual farmer. Knowledge about farmer decisions is important for understanding

---

[5] http://www.ecochange-project.eu/ecochange-project.

Table 2
Comparison of data sources and configuration options used in the Lunan and Aarau case studies.

Component	Aarau	Lunan
Landscape and econ.		
Parcel Structure	Aargauisches Geographisches Informationssystem (AGIS)	SIACS[b]
Available regimes	Social survey and expert knowledge	Expert knowledge
Farm sizes/locations	Random allocation using a histogram of observed sizes	SIACS[b]
Current prices and costs	Farm management handbooks[a]	farm management handbooks[c]
Prices projection	Abildtrup et al., 2006	Abildtrup et al., 2006
Land managers		
Preferences	Choice-based conjoint analysis	Choice-based conjoint analysis
Proportions	Typology based on attitudes and objectives (Karali, 2012)	Attitudinal typology (Guillem et al., 2012)
Capabilities	Based on current regimes	Based on historical data 2000–2008 (SIACS)
Utility function	Weighted sum	NonLinear sum
Decision process	PickFromBest	PickFromBest
Feedback and ESSi		
Social	Access to greenspace, nuisance (expert judgement), tradition (years land cover present 1996–2009)	Tradition and recreation (expert judgement)
Environmental	Land cover diversity of regime (Simpson's Index), nitrogen input per land cover (kg of N/ha) (farm management handbooks)	Land cover, nitrogen need, diversity of rotations (based on DEFRA scoring system)
Ecosystem services	Food (crop, meat) and bioenergy crop production	Food and bioenergy production; charismatic species population (skylark)
Vegetation modelling		
Development	LPJ, loosely coupled	LPJ, loosely coupled

[a] Agridea 1996–2009, FiBL.
[b] Spatially-explicit Integrated Administration and Control System. They are digital GIS maps of about 1600 parcels with unique ownership ID (about 115 farms).
[c] Economic and management data for the UK arable and livestock sector, edited every year by SAC, Edinburgh, UK.

landscape patterns and changes in rural areas, and informing policy-making. In this context, Aporia was used to explore the ways that farmer responses to different socio-economic conditions might affect rural landscapes in the study area.

In 2010, a survey was used to collect data on farmers' behaviour and decision-making. Analysis of farmer attitudes and objectives acquired through the survey identified four types of farming behaviour: business-oriented, lifestylers, multifunctionalist and traditionalist (Karali et al., 2013). Each of these types was implemented as agent types using Aporia. Farmer-agent preferences for environmental, economic and social incentives were estimated using a set of indicators as proxies (Table 2). Due to the absence of spatially explicit parcel boundary data, an artificial (stylised) landscape was constructed. Three hypothetical scenarios from the ALARM project (Spangenberg, 2007)were used to set alternative contexts for the simulations. The ALARM scenarios follow the trajectories of the IPCC Special Report on Emissions Scenarios (Nakićenović and Swart, 2000), and were locally downscaled for the study area using a participatory approach (Bohunovsky et al., 2010). The scenarios were: i.) Business-As-Might-Be-Usual (BAMBU) following A_2; ii.) GRowth Applied Strategy (GRAS), a neoliberal, growth oriented scenario following A_1F_I; and iii.) Sustainable European Development Goal (SEDG), a more environmentally sustainable pathway following B_2. The submodels of *Policy*, *Market*, *Technology* and the *Social* and *Environmental* feedbacks (Section 2.4.4, Fig. 5) were informed through different sources such as social surveys, official sources (e.g., Farm management handbooks, FiBL), and expert judgement (Table 2).

Simulations were run from 2000 to 2050, with an annual timestep. Output on the proportion of farmland surface managed with different farming systems (intensive, extensive, organic) showed shifts from the baseline conditions in all three scenarios (Fig. 7a). Differences in the magnitude and extent of these changes suggest the strong effect that socio-economic context has on farmer decisions.

In the GRAS scenario, for example, the higher output prices for organically produced food resulted in an increase in organically managed farmland, even in the absence of subsidies encouraging its adoption. A similar increase in organic farming was also shown in the BAMBU scenario. The proportion of the different farming systems, however, remained stable during the simulation. This reflected the small changes in the parameter values for this scenario in 2020 and 2050.

In the SEDG scenario, a high proportion of farmland was managed intensively, which contradicted our expectations based on the storyline narrative. In this scenario societal feedback played less of a role since a farming nuisance[6] was considered not to exist in an environmentally-oriented world. By relaxing the importance of this indicator, farmer agents result in choosing management actions with higher nuisance scores. The overall area used for organic Regimes in the SEDG scenario was higher than for the other two farming systems, as one would expect. Although organic farming is associated with lower yields, the higher market prices and the subsidies that farmers received for organic farming in this scenario offset the income losses due to low productivity. The effect of subsidies was demonstrated clearly in 2050 for the SEDG scenario, when the relative difference between the subsidies farmers received for organic and other management subsidies becomes as high as 37% (see Table 6, Appendix 6).

3.2. Lunan catchment

The Lunan catchment project was initiated as part of a RERAD[7] research fund for assessing and improving the sustainability of intensive arable farming systems in Scotland. The catchment covers an area of 132 km^2 on the East coast of Scotland. Intensive management has created environmental and social issues in the area,

---

[6] Nuisance is concept that reflects society's perceptions on the potential harm that might be caused by farming practices.

[7] Rural and Environment Research and Analysis Directorate (Scottish Government).

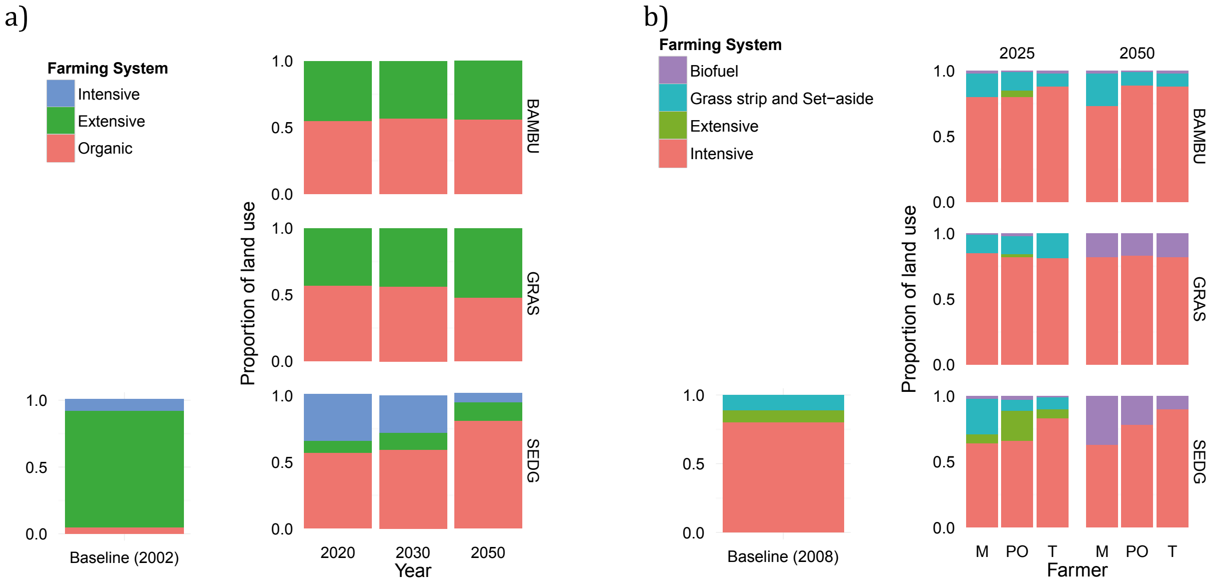


Fig. 7. Changes in the proportion of farmland in the two case studies. a) Aarau, intensive, extensive and organic management per scenario. b) Lunan, proportions of intensive, extensive, grass strips/setaside and biofuel farming per scenario, broken down per farmer type: M = multifunctional, PO = profit oriented, T = traditional.

e.g., diffuse water pollution and changes to traditional landscapes. Improving our understanding of farmer decision-making and its heterogeneity in the region was considered a critical step in the design of successful policy instruments to improve the sustainability of farming practices in the area. In this sense, Aporia was used to evaluate heterogeneous farmer reactions to different socio-economic and policy contexts, and assess their impacts on land-use change and ESSis.

An initial social survey was conducted to assess the different attitudes and behaviour of Lunan farmers. Results were used to generate an attitudinal typology of farmer types containing profit-oriented, multi-functionalist and traditionalist approaches (Guillem et al., 2012). Choice-based conjoint experiments were then used to estimate preferences about the economic, environmental and social factors driving land-management decisions for each group. This information was used to inform the proportions of each agent type and the weights in their utility function within the model.

Land Manager (i.e., farmer) agents were assigned to farm parcels representing real patterns of farm ownership (Table 2). Based on current management practices, the farmer agents evaluate more than 150 Regimes that are possible in arable areas of Scotland, which vary in terms of length, crops, level of inputs (i.e., intensive or extensive) and uses (e.g., food, agri-environmental action, bioenergy).

Economics in the model were set up using data from Farm Management Handbooks to parameterise the *Market* submodel (Section 2.4.4). A simple scoring system was used to set up the Environmental and Social feedbacks (see Table 2). The environmental feedback score is an average of three indicators selected after focus groups with farmers, with scores adapted from DEFRA (2002). These are: i) *Land cover*, which represents the proportion of time within a Regime that vegetation cover is present on a parcel; this is important for the preservation of soil; ii) *Diversity* of Regimes, which characterises the length of a crop rotation and the variety of plant groups within the rotation; this contributes to soil aeration and soil invertebrate diversity; and iii) *Nitrogen Need*, which represents the balance of nitrogen added to the soil against the amount absorbed by crops. The social feedback indicators used are: i) *Recreation,* which is estimated through a binary score representing the accessibility of each Regime[8]; and ii) *Tradition*, which is scored according to the length of time each crop has been produced in the area (information is sourced from agricultural census and FAO data).

For comparative purposes, the submodels were parameterised using the same ALARM scenarios as in the Aarau case study. Prices and costs, as well as the improvement of crop yields (technological factor) were projected based on Abildtrup et al. (2006). Locally relevant *Policy* changes were introduced, such as decoupled payments for a variety of agricultural activities, voluntary agri-environmental schemes (i.e., set-aside and grass strips), and support for bioenergy cropping.

Simulations were performed for each of the scenarios from 2000 to 2050. Fig. 7b shows the proportion of different broad categories of Regimes broken down by farmer type and scenario. The difference in allocation is especially evident in the SEDG scenario, where the non-economic aspects of farming are encouraged. The multi-functionalist farmers apply grass margins and bioenergy on a larger area than the two other types to increase their environmental utility. The profit-oriented farmers adapt to market signals rather than other socio-environmental incentives. The traditionalists maintain intensive management while increasing the social and environmental attributes by selecting, for instance, traditional Regimes that are longer and more diverse. This type of output shows both the effects of the socio-economic context and the importance of internal values on decision-making (Guillem et al., 2012, in review).

4. Discussion

4.1. Evaluating a framework

The evaluation of a framework is a different process than the evaluation of a particular model's application. In this case,

---

[8] For instance, public access through the parcel is assumed to be possible in crops with grass margins.

we concern ourselves with questions of *flexibility*, *extensibility*, *verification* and *transparency*, however, we also draw inspiration from Argent et al. (2006) and Estrada and Rebollar (2006).

Flexibility is the ability of the framework to be used in different situations to answer different questions without the need for additional coding. To date, three case studies have been implemented using the model – the Lunan catchment, the Aarau valley (both detailed above), and the Brabant-Walloon region in Belgium. These case studies address a range of questions: in Aarau, the changes expected in rural landscapes due to a range of socio-economic drivers; in Lunan, the effect of policy on ESS provision; and in Belgium the potential for transitions to sustainable land use practices. In each case, the framework has been used with different data and components to instantiate models that answer case-study specific research questions. Only in the Brabant-Walloon case study was it necessary to extend the framework to add in an urban submodel and support for spatial policies. This demonstrates that the framework has wide applicability within the agricultural land-use domain, and potentially beyond this sector.

To a large extent, flexibility is ensured due to the declarative configuration approach (Section 2.1). In particular, mapping configuration files onto a set of semi-autonomous objects allows for deep specification of behaviour, and provides many entry points for coding bespoke functionality where necessary. For example, agent behaviours can be constructed without writing Java – simply by composing specifications for behavioural components along with their parameters.

Refinement is a related notion (Estrada and Rebollar, 2006), concerned with the framework's ability to adapt to increasingly complex representations as model development progresses. A prime example of this is the range of vegetation modelling techniques, which allows for a smooth transition from simple toy models, using artificial yield values, through empirically informing the model with case study yields and climate responses up to a technically complex and computationally demanding fully integrated model (Luus et al., 2013). Altogether, a model can be changed from a simple landscape with a few farmers and simple economics, into a complex simulation involving changes in policy, productivity and market fluctuations with several different visualisations by only editing XML and CSV files.

Extensibility refers to how easily the framework can be extended to deal with new processes or improved models of existing processes, and takes over from the *flexibility* discussed previously. While we have not made a formal analysis, Table 1 illustrated typical modifications, and the time and competence required to carry them out. We identified four methods of extending or modifying a simulation: *composition*, *substitution*, *addition* and *modification*, in increasing order of difficulty. In general, most work so far has been carried out through composition, in large part by non-programmers combining existing components to set up the behaviour they require. Some instances of substitution have occurred, with computationally-oriented modellers taking on the task of writing new versions of components. Where addition has been necessary this has generally been handled by (or in close association with) the original framework designers, although it is quite conceivable that others could carry out this process as well. Finally, most of the *modification* carried out (exclusively by the original designer) has been the implementation of planned-for functionality. Taking a modelling framework approach means that modification extensions are more difficult to implement than they would be in a standalone model, as a broader range of possible uses must be taken into account, and care must be taken not to break existing functionality. However, the trade-off is that the increased capabilities can then be used by all current and future modellers. A concrete example is that adding spatially explicit support as required in one case study (Section 2.9) then allowed researchers working on other case studies to consider spatial policies in their work.

The examples given should serve to illustrate the generally extendable nature of the core codebase and the power of the declarative approach, as a step towards a "Modelling Playground" approach (Arnold, 2012). At the same time, it makes clear the benefit of using a framework to construct ABMs—centralised improvements become available to all the modellers involved in the project for seamless integration into their existing simulations.

Verification, *validation* and *calibration* are always an issue with models of complex systems. Validation of spatially-explicit models into the future is still a difficult challenge (Pontius et al., 2008, 2004). Validation also typically relies on the stationarity of a situation for modelling. Hence, the case studies presented here take the approach of *calibration* with case study data, followed by scenario analysis (e.g., Murray-Rust et al., 2013; Rounsevell et al., 2006). However, *verification* is an appropriate question to ask of a framework: one of the key benefits of an open, extensible framework is that core functionality can be extensively tested and verified. Aporia is developed under a Test Driven Development methodology: routines to test the correctness of functionality are created alongside the functionality itself. Unit Tests are used to test the functionality of individual methods, and integration tests test the behaviour of small example scenarios. Currently, 171 tests are used, which cover 87% of the lines in the code (excluding visualisation classes).

Related to *verification* is the issue of *transparency* (or traceability - Estrada and Rebollar (2006)). In this case, "how easy is it for modellers using the system to investigate the behaviour of their models and understand what is happening?" Aporia provides several tools to achieve this:

- Configurable output files can be used to give post-hoc analysis of most model components, including the status of individual Land Managers and Parcels; subsidies paid, quantities and prices of various crops etc. Spatially explicit data such as land use and land cover can also be provided as shapefiles.
- Live maps, both using the default RepastS functionality and higher performance custom displays give a real-time insight into ESSi levels, vegetation height and patterns of land cover.
- Dedicated displays can be used to drill down into submodels and individual agents. This allows modellers to observe directly the effect of different decision strategies and preference structures.
- Finally, as part of an experiment into the communication of ABM results to stakeholders, a communication protocol has been created, which allows for real-time communication of model state with a high quality 3D visualisation engine.

All of these points together contribute to making a complex model understandable to its users in an interdisciplinary context (Ravetz, 2006). Additionally, Aporia has been designed around the use of version control for both code and data to support understanding of the evolution of a particular case study implementation. The declarative nature of the configuration is an aid to understanding certain model assumptions, as they are effectively stated in the input files. However, more work could be carried out to elucidate the full range of model assumptions in a clear and concise manner. Similarly, more immediate descriptions and visualisations of model structure could help the end user to understand the operation of the framework and the connections between components.

A framework is never complete; there is always a laundry list of functionality desired by users and designers, waiting for time, resources or relevant data before it is implemented, and enumerating these is pointless. Instead, we highlight some areas of future expansion in the spirit of better explaining the capabilities of Aporia that are either present, but not yet used, or have infrastructural support, but have not yet been developed.

Firstly, land exchange is a major part of many land-use models; the Urban submodel demonstrates the infrastructure support for exchange of ownership and management, but the current cohort of agents have not been set up to make that kind of decision, and no land market or similar structure has been created. Similarly, land abandonment, and the subsequent succession of vegetation has support, but is not a feature of the case studies examined so far. Finally, the learning, communication and planning carried out by agents is currently quite limited; however, there is the possibility for agents to look at the variance and trends in yields over time and space; to look at the choices made and results obtained by their neighbours or social networks. With land manager agents in particular, we feel that Aporia now provides a stable foundation on which to explore more detailed models of decision-making and social structures.

4.2. Conclusions

We have presented the Aporia framework, and discussed it in terms of *flexibility*, *extensibility*, *verification*, and *transparency*, both in terms of building simple models out of existing components and the effort required to add additional components. The declarative and compositional approach taken increases the flexibility of the framework[9] and allows for deep configuration by non-programmers as well as integration with a range of other models and applications. It can be used additionally by starting with a simple representation that is gradually *refined* to a more complex, high-fidelity model. The case studies in Scotland and Aarau have demonstrated how a detailed model can be set up using a combination of existing agro-economic and land use data, a social survey and future storylines, and further how these storylines lead to simulated changes of land management and the adoption of alternative farming techniques. The *extensibility* of the system has been shown by developing formulations for spatially explicit policies and urbanisation. The model is highly *transparent* as almost all model states can be investigated at runtime, or output for later perusal.

In summary, the Aporia framework allows for a range of land use models, of varying complexity to be built and calibrated with real world data. It combines several advances in ABM of land use, including declarative composition; a detailed multi-level handling of temporality, with distinctions between land cover and land management; a varied socio-economic context supporting multi-criteria decision-making; ecosystem service modelling within the framework; with a collection of displays and output support to allow easy analysis and testing of models that are created.

Acknowledgements

This framework has been produced under the EU ECOCHANGE project EU GOCE-036866. Additional thanks go to Lilibeth Acosta-Michlik and Corentin Fontaine for testing, debugging and using the framework, plus Nicolas Dendoncker, Lisa Bohunovsky, Ines Omann, Julia Wesely and Anna Hersberger for their input into the development of the framework. We would also like to thank the reviewers for their support in improving this manuscript.

1. Tags – annotations for parcels and regimes

Tags have been used throughout Aporia to provide a simple general mechanism for expandability. Each “tags” entry is an unordered set of user defined strings. These tags are used in several places in the framework:

- Each Regime and Management has a set of Capability tags, representing abstract clusters of skills and equipment required to carry out the given activity. This can be used restrict the ability of agents to execute Regimes – if a manager does not have all the capabilities specified in the Regime, it cannot be used. This can be used to e.g., separate livestock from arable farmers, or farmers with access to capital (and hence ability to grow intensive crops) from those without.
- Each Regime and Parcel has a set of Policy tags. These tags can be used to give a spatial aspect to Policy behaviour. For example to implement a policy which restricts the use of high nitrate Regimes near riverbanks, some Regimes could be tagged with a “high-nitrates” tag, and some Parcels with a “near-river” tag. A Spatial Policy could then be created to forbid the application of those Regimes on those Parcels.
- Each Parcel and Regime can have a set of Biophysical Tags which indicates the biophysical properties required by the Regime and provided by the Parcel. Together these delimit the range of possible Regimes applied to a given Parcel, representing constraints such as slope, soil type, and e.g., disallowing highly mechanised agriculture on hillsides.

2. Default assessor equations

The Default Assessor can have scores set for Land Covers, Managements and Regimes, notated s_l, s_m, s_r respectively. There is also an assumption that the score does not change based on the parcel. Basic scores are calculated as follows:

$$S(l) = s_l \text{ or } 0$$

$$S(m) = s_m \text{ or } S(l)$$

Scores for Regimes are more complex. There are several configurations that can be used, which are appropriate for different kinds of assessment (Table 3). For example, when comparing economic returns, one would tend to calculate the sum of all of the Management scores (i.e., total income), taken at the end of each Management, with future gains discounted and averaged over the number of years to get an average gross margin; alternatively, ESSi indicators might be calculated as an average over the number of days a Management is present.

When analysing applications of Regimes to Parcels, it is possible to return an area sum, an area weighted average, or an unweighted sum of the values for each Parcel.

3. Creating new assessors

If the existing assessors do not provide enough functionality, then new Assessors can be created. The assessor interface requires several methods for assessing different kinds of action, from single

---

[9] Arguably, it is the ability to create models by composing a variety of components which makes this a framework rather than a model.

Table 3
Evaluator parameters and their effects.

Parameter	Effect
averageByTime	If true, then the score is averaged over the time the management is present. Otherwise, it is computed once for each management.
averageByManagements	If true, then the score for the regime is averaged by the number of managements, otherwise it is by the number of years or not at all.
averageByHorizon	If true, then a yearly average score is returned, otherwise the total score is returned.
regimeValuesPerYear	If true, then when specific values are given for regimes, they are taken as yearly (i.e. for a 5 year regime, they will be multiplied by 5). Otherwise, values are treated as scores for the entire regime, and if part of the regime is assessed, then a proportional amount will be returned.
regimeValuesConstant	If true, then the any regime scores are returned directly, i.e. without any time averaging. This is good for when the score represents an average value already.
countUnmanagedTime	If true, then time within a regime which doesn't have a management applied will be counted. Otherwise, it won't be taken into account when calculating values. For example, if a management is applied for 100 days in a year, if countUnmanagedTime is false, then the basic score is returned. Otherwise, it is the basic score*100/365.
countAtEnd	If true, then managements are counted when the finish (e.g., for harvesting them) otherwise when they start (e.g., for costs).
discountFactor	If this is more than 0, it is used to discount future rewards, i.e. value = base*(1 − discount)^n for values n years in the future.

Land Covers up to a 5 year plan for several Parcels. The easiest way to manage this is to subclass DefaultAssessor, which has sensible default implementations of the more complex methods, as shown in Table 4.

In addition, it might be desired to combine the scores from several assessors in different ways. The NonLinearSumAssessor takes a set of standard assessors, applies a transform to each one, and returns the sum. New transforms can be defined with a single method class.

4. Ecosystem services implemented

A number of ESSi have been implemented. Table 5 gives an overview. Most of the indicators were configured using the standard assessor framework; using either the default assessor for indicators based on Land Cover/Management, or the harvest assessor for services which are calculated based on yields of some kind. Additional code was necessary to calculate landscape aesthetics from a variety of patch based metrics, and to integrate input from a custom IBM and the LPJ guess vegetation model.

5. Example configuration overview

Table 4
Assessor methods to override.

Object to assess	Default implementation
Land cover	0
Management	Score for the management's land cover.
Management applied to a parcel	The management's score, optionally scaled by Parcel area.
Management in X years time	The management's score discounted by (1 − df)^x where df is a configurable discount factor defaulting to 0.
Management on a parcel in X years time	As above.
Application of several managements to parcels	The score for each management applied to each parcel, optionally averaged by area or by the number of parcels.
Regime (optionally the first n years, with an X year offset, applied to a parcel)	The average of the scores for all the managements in the Regime, optionally weighted by time or the amount of time they are applied.
Assignment of regimes to parcels	The score for each regime, optionally averaged by area.

Table 5
Ecosystem service indicators currently implemented.

Ecosystem service indicator	Technique	Typical data/calibration
Landscape aesthetics	Custom equation	Sum of: proportion of natural vegetation, Simpson's evenness index, number of patches and patch contagion
Biofuel harvest	Harvest ESSi	Vegetation production and conversion rates
Food production (crops and meat)	Harvest ESSi	Vegetation and meat production, and conversion rates
Charismatic species	Individual based model of skylark population	Field data and literature
Recreation provision	Standard assessor	Expert judgement
Carbon storage	Coupling with LPJ	
Traditional practices	Standard assessor	Expert judgement and historic data
Protection against soil erosion	Standard assessor	DEFRA scores
Diversity of rotations	Standard assessor	DEFRA scores
Nitrogen cycle	Standard assessor	DEFRA scores

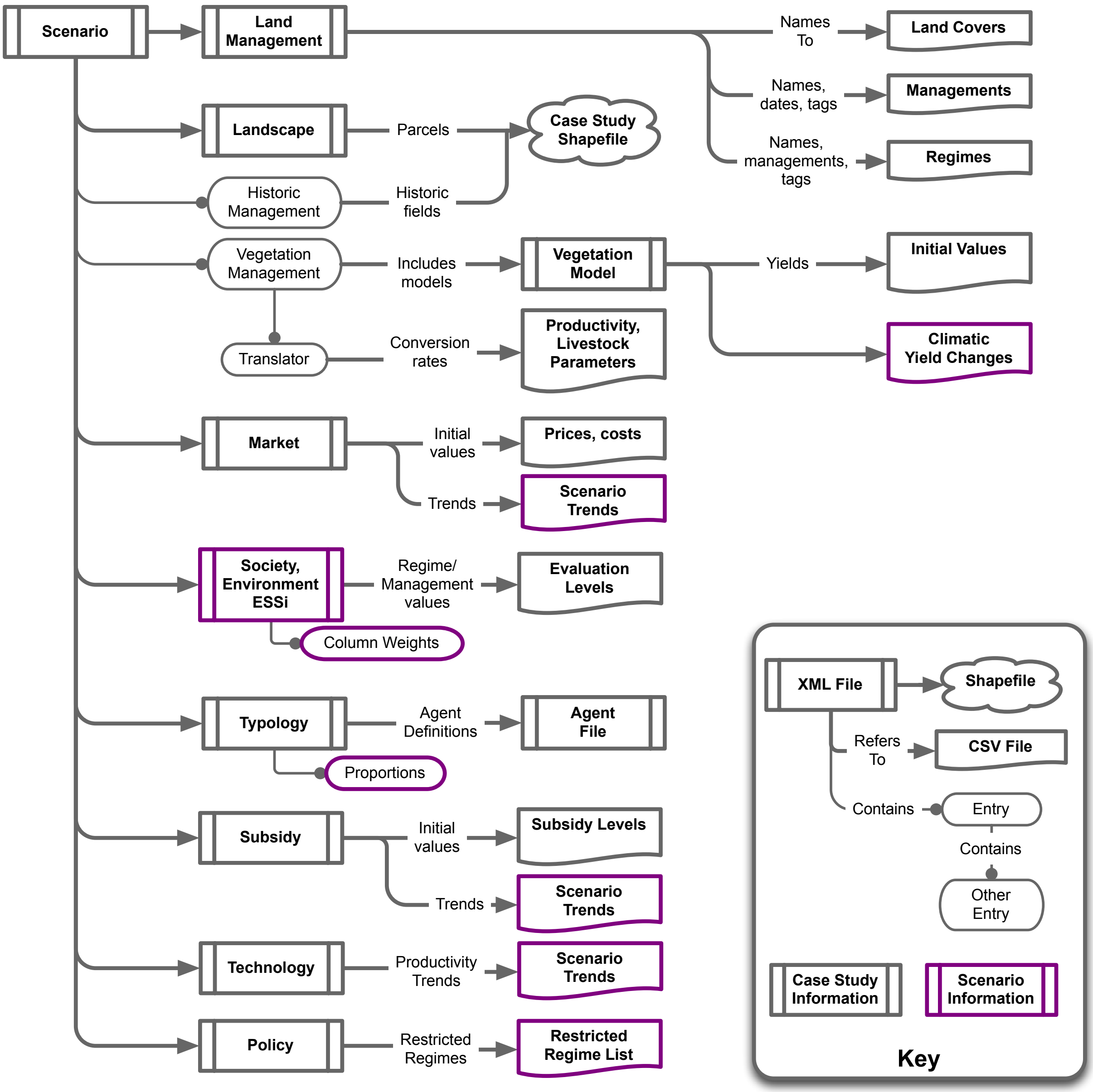


Fig. 8. Example file structure, showing XML, CSV and shapefiles, and highlighting scenario dependent files.

6. Example parameterisation

Table 6
Model parameterisation for the Aarau case study, per scenario, relative to baseline (2000) for two future timeslices (2020 and 2050).

Variable/driver	Description	Baseline year	BAMBU		GRAS		SEDG	
		2000	2020	2050	2020	2050	2020	2050
		1.00	1.31	1.77	1.67	2.68	1.04	1.09
Area subsidies	Changes in subsidy values relative to year 2000							
Intensive managements			0.9	0.8	0	0	0.9	0.8
Extensive managements		1.00	0.9	0.8	0	0	0.9	0.8
Organic managements		1.00	0.9	0.8	0	0	1.07	1.17
Prices	Changes in market prices relative to year 2000							
Cereal prices		1.00	0.92	0.80	0.83	0.59	1.12	1.31
Meat prices		1.00	0.98	0.95	0.96	0.89	1.10	1.24
Costs	Changes in costs relative to baseline year							

Table 6 (*continued*)

Variable/driver	Description	Baseline year	BAMBU		GRAS		SEDG	
		2000	2020	2050	2020	2050	2020	2050
		1.00	1.31	1.77	1.67	2.68	1.04	1.09
Seed		1.00	1.08	1.21	1.20	1.49	0.94	0.85
Machinery		1.00	1.06	1.16	0.88	0.70	1.32	1.79
Pesticide		1.00	1.02	1.04	0.88	0.69	1.47	2.18
Fertiliser		1.00	1.07	1.18	0.94	0.85	1.61	2.59
Labour		1.00	1.14	1.35	1.51	2.28	1.10	1.25
Social feedback								
Nuisance	Regime contributions to smell, noise and health		45%		45%		10%	
Access to greenspace	The potential use of the location covered by a certain regime for recreational purposes		20%		45%		45%	
Tradition	Number of years that regime is present in the area		35%		10%		45%	

References

Abildtrup, J., Audsley, E., Fekete-Farkas, M., Giupponi, C., Gylling, M., Rosato, P., Rounsevell, M., 2006. Socio-economic scenario development for the assessment of climate change impacts on agricultural land use: a pairwise comparison approach. Environ. Sci. Policy 9, 101–115.

Argent, R.M., Voinov, A., Maxwell, T., Cuddy, S.M., Rahman, J.M., Seaton, S., Vertessy, R.A., Braddock, R.D., 2006. Comparing modelling frameworks – a workshop approach. Environ. Model. Softw. 21, 895–910.

Arnold, T.R., 2012. Procedural knowledge for integrated modelling: towards the modelling playground. Environ. Model. Softw., 1–14.

Bailey, K.D., 1994. Typologies and Taxonomies: an Introduction to Classification Techniques. In: Sage University Papers: Quantitative Applications in the Social Sciences, No. 07-102. Sage Publications, Inc., Thousand Oaks, CA, US.

Barnaud, C., Promburom, P., Trébuil, G., Bousquet, F., 2005. Companion modelling to support collective land management in the highlands of Northern Thailand. In: Proceedings of the International Conference on Natural Resources Related Conflicts Management in Southeast Asia. Institute of Dispute Resolution, Khon Kaen University, Khon Kaen, Thailand.

Berger, T., Schreinemachers, P., 2006. Creating agents and landscapes for multiagent systems from random samples. Ecol. Soc. 11, 19.

Bithell, M., Brasington, J., 2009. Coupling agent-based models of subsistence farming with individual-based forest models and dynamic models of water distribution. Environ. Model. Softw. 24, 173–190.

Bohunovsky, L., Jäger, J., Omann, I., 2010. Participatory scenario development for integrated sustainability assessment. Reg. Environ. Change 11, 271–284.

Carpenter, S.R., Mooney, H.A., Agard, J., Capistrano, D., DeFries, R.S., Díaz, S., Dietz, T., Duraiappah, A.K., Oteng-Yeboah, A., Pereira, H.M., others, 2009. Science for managing ecosystem services: beyond the millennium ecosystem assessment. Proc. Natl. Acad. Sci. U. S. A. 106, 1305–1312.

Castellazi, M., Wood, G., Burgess, P., Morrise, J., Conrad, K., Perry, J., 2008. A systematic representation of crop rotations. Agric. Syst. 97, 26–33.

Chabrier, P., Garcia, F., Martin-Clouaire, R., Quesnel, G., Raynal, H., 2007. Toward a simulation modeling platform for studying cropping systems management: the record project. In: International Congress on Modelling and Simulation. International Society for Computer Simulation, pp. 10–13.

Cretegny, L., 2001. The agricultural policy reform in Switzerland: an assessment of the agriculture multifunctionality. In: 4th Annual Conference on Global Economic Analysis. Purdue University.

Curry, N., Stucki, E., 1997. Swiss agricultural policy and the environment: an example for the rest of Europe to follow? J. Environ. Plan. Manag. 40, 465–482.

Deadman, P.J., Robinson, D.T., Moran, E., Brondizio, E., 2004. Colonist household decision-making and land-use change in the Amazon Rainforest: an agent-based simulation. Environ. Plan. B: Plan. Des. 31, 693–709.

DEFRA, 2002. Arable Cropping and the Environment – a Guide.

Dogliotti, S., Rossing, W., van Ittersum, M., 2003. ROTAT, a tool for systematically generating crop rotations. Eur. J. Agron. 19, 239–250.

Edwards-Jones, G., 2006. Modelling farmer decision-making: concepts, progress and challenges. Anim. Sci. 82, 783–790.

Estrada, H., Rebollar, A., 2006. An empirical evaluation of the i* framework in a model-based software generation environment. Adv. Inf. Syst. Eng., 513–527.

Evans, T., Robinson, D.T., Schmitt-Harsh, M., 2013. Limitations, challenges, and solutions to integrating carbon dynamics with land-use models. In: Brown, D.G., Robinson, D.T., French, N.H.F., Reed, B.C. (Eds.), Land Use and the Carbon Cycle: Advances in Integrated Science, Management, and Policy. Cambridge University Press.

Evans, T.P., Kelley, H., 2004. Multi-scale analysis of a household level agent-based model of landcover change. J. Environ. Manag. 72, 57–72.

Evert, F., Holzworth, D., 2005. Convergence in integrated modelling frameworks. In: MODSIM 2005 International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand, pp. 745–750.

Guillem, E., Barnes, A., 2012. Modelling CAP Reform Measures on an Intensive Cropping Area. In: Rural Policy Centre Research Briefing. SAC, Edinburgh, UK.

Guillem, E.E., 2012. Human Behaviour and Ecosystem Services in Sustainable Farming Landscapes – an Agent-Based Model of Socio-Ecological System. University of Edinburgh.

Guillem, E.E., Barnes, A.P., Rounsevell, M.D.A., Renwick, A., 2012. Refining perception-based farmer typologies with the analysis of past census data. J. Environ. Manag. 110, 226–235.

Guillem, E.E., Murray-Rust, D., Robinson, D.D.T., Barnes, A., Acosta-Michlik, L., Rounsevell, M.D.A. Observation of heterogeneous farm strategies and sustainable trade-offs using an empirically-grounded agent-based model. Environ. Model. Softw. (in review).

Hancock, J., 2010. The case for an ecosystem service approach to decision-making: an overview. Biosci. Horiz. 3, 188.

Huigen, M.G.A., 2004. First principles of the MameLuke multi-actor modelling framework for land use change, illustrated with a Philippine case study. J. Environ. Manag. 72, 5–21.

Janssen, M.A., Ostrom, E., 2006. Empirically based, agent-based models. Ecol. Soc. 11, 37.

Karali, E., 2012. Investigating the Effect of Farmer Land-Use Decisions on Rural Landscapes Using an Agent-Based Model Approach. University of Edinburgh.

Karali, E., Brunner, B., Doherty, R., Hersperger, A.M., Rounsevell, M.D.A., 2013. The effect of farmer attitudes and objectives on the heterogeneity of farm attributes and management in Switzerland. Hum. Ecol. 45, 915–926.

Le, Q.B., Park, S.J., Vlek, P.L.G., Cremers, A.B., 2008. Land use dynamic simulator (LUDAS): a multi-agent system model for simulating spatio-temporal dynamics of coupled human-landscape system: 2. Scenario-based. Ecol. Inform. 3, 135–153.

Le, Q.B., Seidl, R., Scholz, R.W., 2012. Feedback loops and types of adaptation in the modelling of land-use decisions in an agent-based simulation. Environ. Model. Softw. 27-28, 83–96.

Levins, R., 1966. The strategy of model building in population biology. Am. Sci. 54, 412–431.

Lindeskog, M., Arneth, A., Bondeau, A., Waha, K., Seaquist, J., Olin, S., Smith, B., 2013. Implications of accounting for land use in simulations of ecosystem services and carbon cycling in Africa. Earth Syst. Dyn. Discuss. 4, 235–278.

Luus, K.A., Robinson, D.T., Deadman, P.J., 2013. Representing ecological processes in agent-based models of land use and cover change. J. Land Use Sci. 8, 175–198.

Matthews, R., 2006. The people and landscape model (PALM): towards full integration of human decision-making and biophysical simulation models. Ecol. Model. 194, 329–343.

Millington, J., Romero-Calcerrada, R., Wainwright, J., Perry, G., 2008. An agent-based model of Mediterranean agricultural land-use/cover change for examining wildfire risk. J. Artif. Soc. Soc. Simul. 11, 4.

Minar, N., Burkhart, R., Langton, C., Askenazi, M., 1996. The Swarm Simulation System: a Toolkit for Building Multi-Agent Simulations.

Monticino, M., Acevedo, M., Callicott, B., Cogdill, T., Lindquist, C., 2007. Coupled human and natural systems: a multi-agent-based approach. Environ. Model. Softw. 22, 656–663.

Muetzelfeldt, R., Massheder, J., 2003. The Simile visual modelling environment. Eur. J. Agron. 18, 345–358.

Murray-Rust, D., Dendoncker, N., Dawson, T.P., Acosta-Michlik, L., Karali, E., Guillem, E., Rounsevell, M., 2011. Conceptualising the analysis of socio-ecological systems through ecosystem services and agent-based modelling. J. Land Use Sci. 6, 83–99.

Murray-Rust, D., Rieser, V., Robinson, D.T., Milicic, V., Rounsevell, M., 2013. Agent-based modelling of land use dynamics and residential quality of life for future scenarios. Environ. Model. Softw. 46, 75–89.

Nakićenović, N., Swart, R. (Eds.), 2000. IPCC Special Report: Emissions Scenarios, Working Group III of the Intergovernmental Panel on Climate Change IPCC. Cambridge University Press.

Nikolai, C., Madey, G., 2009. Tools of the trade: a survey of various agent based modeling platforms. J. Artif. Soc. Soc. Simul. 12, 2.

North, M., Howe, T., Collier, N., Vos, J., 2007. A declarative model assembly infrastructure for verification and validation. In: Advancing Social Simulation: the First World Congress. Springer, pp. 129–140.

North, M.J., Collier, N.T., Vos, J.R., 2006. Experiences creating three implementations of the repast agent modeling toolkit. ACM Trans. Model. Comput. Simul. 16, 1–25.

North, M.J., Howe, T.R., Collier, N.T., Vos, J.R., 2005. The repast simphony runtime system. In: Proceedings of the Agent 2005 Conference on Generative Social Processes Models and Mechanisms, pp. 151–158.

Parker, D.C., Manson, S.M., Janssen, M.A., Hoffmann, M.J., Deadman, P., 2003. Multi-agent systems for the simulation of land-use and land-cover change: a review. Ann. Assoc. Am. Geogr. 93, 314–337.

Parunak, H. van D., Savit, R., Riolo, R.L., 1998. Agent-Based Modeling vs. Equation-Based Modeling: a Case Study and Users' Guide. In: Lecture Notes in Computer Science, vol. 1534, pp. 277–283.

Pontius, R.G., Boersma, W., Castella, J.C., Clarke, K., de Nijs, T., Dietzel, C., Duan, Z., Fotsing, E., Goldstein, N., Kok, K., 2008. Comparing the input, output, and validation maps for several models of land change. Ann. Reg. Sci. 42, 11–37.

Pontius, R.G., Huffaker, D., Denman, K., 2004. Useful techniques of validation for spatially explicit land-change models. Ecol. Model. 179, 445–461.

Primdahl, J., Peco, B., Schramek, J., Andersen, E., Onate, J.J., 2003. Environmental effects of agri-environmental schemes in Western Europe. J. Environ. Manag. 67, 129–138.

Ravetz, J., 2006. Post-normal science and the complexity of transitions towards sustainability. Ecol. Complex. 3, 275–284.

Reynolds, C., 1987. Flocks, herds and schools: a distributed behavioral model. ACM SIGGRAPH Comput. Graph..

Robinson, D.T., Brown, D.G., Parker, D.C., Schreinemachers, P., Janssen, M.A., Huigen, M., Wittmer, H., Gotts, N., Promburom, P., Irwin, E., 2007. Comparison of empirical methods for building agent-based models in land use science. J. Land Use Sci. 2, 31–55.

Robinson, D.T., Murray-Rust, D., Rieser, V., Milicic, V., Rounsevell, M., 2012. Modelling the impacts of land system dynamics on human well-being: using an agent-based approach to cope with data limitations in Koper, Slovenia. Comput. Environ. Urban Syst..

Robinson, D.T., Sun, S., Hutchins, M., Riolo, R.L., Brown, D.G., Parker, D.C., Filatova, T., Currie, W.S., Kiger, S., 2013. Effects of land markets and land management on ecosystem function: a framework for modelling exurban land-change. Environ. Model. Softw. 45, 129–140.

Rounsevell, M.D.A., Dawson, T.P., Harrison, P.A., 2010. A conceptual framework to assess the effects of environmental change on ecosystem services. Biodivers. Conserv. 19, 2823–2842.

Rounsevell, M.D.A., Metzger, M.J., 2010. Developing qualitative scenario storylines for environmental change assessment. Wiley Interdiscip. Rev. Clim. Change.

Rounsevell, M.D.A., Pedroli, B., Erb, K.-H., Gramberger, M., Busck, A.G., Haberl, H., Kristensen, S., Kuemmerle, T., Lavorel, S., Lindner, M., Lotze-Campen, H., Metzger, M.J., Murray-Rust, D., Popp, A., Pérez-Soba, M., Reenberg, A., Vadineanu, A., Verburg, P.H., Wolfslehner, B., 2012a. Challenges for land system science. Land Use Policy 29, 899–910.

Rounsevell, M.D.A., Reginster, I., Araújo, M.B., Carter, T.R., Dendoncker, N., Ewert, F., House, J.I., Kankaanpää, S., Leemans, R., Metzger, M.J., 2006. A coherent set of future land use change scenarios for Europe. Agric. Ecosyst. Environ. 114, 57–68.

Rounsevell, M.D.A., Robinson, D.T., Murray-Rust, D., 2012b. From actors to agents in socio-ecological systems models. Philos. Trans. R. Soc. B: Biol. Sci. 367, 259–269.

Schreinemachers, P., Berger, T., 2006. Land use decisions in developing countries and their representation in multi-agent systems. J. Land Use Sci. 1, 29–44.

Schweitzer, C., Priess, J.A., Das, S., 2011. A generic framework for land-use modelling. Environ. Model. Softw. 26, 1052–1055.

Sitch, S., Smith, B., Prentice, I.C., Arneth, A., Bondeau, A., Cramer, W., Kaplan, J.O., Levis, S., Lucht, W., Sykes, M.T., others, 2003. Evaluation of ecosystem dynamics, plant geography and terrestrial carbon cycling in the LPJ dynamic global vegetation model. Glob. Change Biol. 9, 161–185.

Spangenberg, J.H., 2007. Integrated scenarios for assessing biodiversity risks. Sustain. Dev. 15, 343–356.

Topping, C.J., Odderskær, P., 2004. Modeling the influence of temporal and spatial factors on the assessment of impacts of pesticides on skylarks. Environ. Toxicol. Chem. 23, 509–520.

Toroczkai, Z., Eubank, S., 2006. Agent-based modeling as a decision-making tool. In: Frontiers of Engineering. National Academy of Engineering, pp. 22–27.

Valbuena, D., Verburg, P.H., Bregt, A.K., 2008. A method to define a typology for agent-based analysis in regional land-use research. Agric. Ecosyst. Environ. 128, 27–36.

Waldrop, M., 1990. Asking for the Moon: the moon-Mars initiative may or may not fly on Capitol Hill, but NASA wants the scientists on its side. Science (New York, NY) 247, 637–638.

Warnant, P., François, L., Strivay, D., Gérard, J.C., 1994. CARAIB: a global model of terrestrial biological productivity. Glob. Biogeochem. Cycles 8, 255–270.