This is the final draft of the contribution published as:

Langhammer, M., **Thober, J.**, **Lange, M.**, **Frank, K.**, **Grimm, V.** (2019):
Agricultural landscape generators for simulation models: A review of existing solutions and an outline of future directions
Ecol. Model. **393** , 135 – 151

The publisher's version is available at:

http://dx.doi.org/10.1016/j.ecolmodel.2018.12.010

Agricultural landscape generators for simulation models: a review of existing solutions and an outline of future directions

[a]Maria Langhammer*, [a,b]Jule Thober, [a]Martin Lange, [a,c,d]Karin Frank, [a,c,e]Volker Grimm

[a]Helmholtz Centre for Environmental Research – UFZ, Department of Ecological Modelling, Permoserstr. 15, 04318 Leipzig, Germany

[b]Helmholtz Centre for Environmental Research – UFZ, Department of Computational Landscape Ecology, Permoserstr. 15, 04318 Leipzig, Germany

[c]German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Deutscher Platz 5e, 04103 Leipzig, Germany

[d]University of Osnabrück, Institute for Environmental Systems Research, Barbarastr. 12, 49076 Osnabrück, Germany

[e]University of Potsdam, Institute of Biochemistry and Biology, Department of Plant Ecology and Nature Conservation, Am Mühlenberg 3, 14476 Golm (Potsdam), Germany

*Corresponding author

Keywords: agricultural landscape, field pattern, agricultural landscape generator, landscape simulator, neutral landscape model, process-based model

Abstract

There is an increasing need for an assessment of the impacts of land use and land use change (LUCC). In this context, simulation models are valuable tools for investigating the impacts of stakeholder actions or policy decisions. Agricultural landscape generators (ALGs), which systematically and automatically generate realistic but simplified representations of land cover in agricultural landscapes, can provide the input for LUCC models. We reviewed existing ALGs in terms of their objectives, design and scope. We found eight ALGs that met our definition. They were based either on generic mathematical algorithms (pattern-based) or on representations of ecological or land use processes (process-based). Most ALGs integrate only a few landscape metrics, which limits the design of the landscape pattern and thus the range of applications. For example, only a few specific farming systems have been implemented. We conclude that existing ALGs contain useful approaches that can be used for specific purposes, but ideally generic modular ALGs are developed that can be used for a wide range of scenarios, regions and model types. We have compiled features of such generic ALGs and propose a possible software architecture. Considerable joint efforts are required to develop such generic ALGs, but the benefits in terms of a better understanding and development of more efficient agricultural policies would be high.

1 Introduction

In response to climate change, a growing human population, and globalisation, land use and land cover are changing at unprecedented rates. Understanding and predicting these changes and their consequences for biodiversity and ecosystem services are among the grand challenges of ecological and environmental research. For intensively used agricultural landscapes, for example, in Europe, key questions include how spatio-temporal patterns in land cover are affected by national and EU policies or by the global market and how, in turn, they are affecting ecosystem services and biodiversity. Empirical approaches to meeting these challenges are limited because of the scale dependency and the multitude of factors involved. Simulation models on land use and land cover change (LUCC) are therefore widely and increasingly developed to back- and forecast landscape changes (Lambin *et al.*, 2000, Agarwal *et al.*, 2002, Parker *et al.*, 2003, Heistermann *et al.*, 2006). Such simulation models, if they are sufficiently realistic, allow us to generate scenarios and to rigorously explore, by using mathematics and computer logics, the consequences of stakeholder actions or political decisions (e.g., Johst *et al.*, 2015).

However, when using spatially explicit simulation models for this purpose, there is a dilemma when it comes to representing land cover in agricultural landscapes. On the one hand, such land cover maps can be taken from maps from geographical information systems (GIS). While such input implies high realism and significance for regional case studies (e.g., Wätzold *et al.*, 2016), general insights and transferability to other regions or questions are limited. A single map does not allow us to systematically vary the features of a landscape. Even if we contrast different real landscapes, we only obtain snapshots of possible relationships, which are likely to be nonlinear. Moreover, the creation of such detailed maps is time-consuming, as it often requires manual data processing, parameterisation and calibration.

On the other hand, an unlimited number of virtual landscapes can be generated using algorithms that systematically and automatically vary landscape metrics such as percentage cover, fragmentation, or spatial autocorrelation (Gardner *et al.*, 1987, Gardner, 1999, Saura & Martínez-Millán, 2000, Hiebeler, 2007, Cambui *et al.*, 2014). Using these virtual landscapes as input, models can provide insights into the consequences of changing landscape features and help to formulate, test and validate hypotheses (Gardner & Urban, 2007). However, such landscapes are usually difficult to relate to real landscapes such that it remains unclear what we have learned about the real world.

The alternative to these abstract landscapes is landscape generators, which generate virtual, but structurally realistic, maps of land cover by trying to combine both realism and the option to vary landscape features. The generators create variations of artificial agricultural land cover mosaics at the spatial resolution of individual fields for a given set of parameters. To be classified as a “generator”, we here require that they allow for varying features of the generated landscapes in a systematic and automated way.

Changes in agricultural landscapes and their ecological consequences occur at small scales (Houet *et al.*, 2010), e.g., the variation of the field mosaic and the implementation of crop rotations or policy measures at the farm level. There is thus a need for high-resolution spatial simulation models and corresponding tools that are capable of generating artificial land cover maps at high resolution under a predefined parameter set. Still, limited research has been done in the field and few such landscape generators have been developed so far. These generators define agricultural landscapes as a mosaic of land-use patches (fields) and landscape elements (e.g., hedges). Landscape features that are typically varied are composition (type and proportion of land cover) and landscape configuration (spatial arrangement of the land covers). The distributions of these features can be taken from distributions observed in real landscapes, such that insights gained from the model are relevant for, e.g., analysing ecological processes or exploring the consequences of EU policy instruments such as the "greening" of farming (e.g., Langhammer *et al.*, 2017) or agri-environment schemes (e.g., Sturm *et al.*, 2018).

So far, no common term has been established for this type of landscape generator. Therefore, we here suggest referring to them as "agricultural landscape generators" (ALGs) and use it in the following as a generic term. ALGs are computer programs that generate structurally realistic but simplified artificial representations of agricultural landscapes, i.e. maps of land cover. Both the landscape configuration (field mosaic) and the landscape composition (land cover) are variable. We nevertheless refer to them as generators to emphasise that they are not used for simulations by themselves and do not display temporal and spatial dynamics. The output of ALGs is a map that can be used as input for LUCC models or other model types. ALGs can still represent change over time by producing a series of consecutive maps. ALGs are either implemented as stand-alone programs or as sub models within LUCC models.

The approaches to generating landscapes can be distinguished into two main categories: pattern-based and process-based. Pattern-based generators, also known as neutral landscape models, are based on generic algorithms and produce virtual landscapes regardless of the underlying ecological or social processes (Gardner *et al.*, 1987, With & King, 1997). They work with one or more characteristics of composition and configuration of a landscape. Regarding complexity, they range from pure neutral models to more-realistic models in terms of landscape structure (Johnson *et al.*, 1999, Gaucherel *et al.*, 2014). The resulting landscapes are mostly pixel matrices, with each pixel representing a spatial unit assigned to a certain land cover class. So far, neutral landscape models have been used primarily in the research field of forest and landscape ecology, but rarely for agricultural landscapes. The coupling of neutral landscapes with population models allows species' perceptions of landscape configuration, e.g., habitat fragmentation and landscape connectivity, to be addressed (With, 1997).

Process-based generators, also known as mechanistic models, produce landscape patterns as a result of ecological or socio-economic processes that are explicitly integrated into the model (Jackson *et al.*, 2000, Cuddington *et al.*, 2013). The result of these generators is also a static map to be used as input for dynamic simulation models. "Process-based" in this context means that the mechanisms leading to a certain landscape pattern can be explicitly addressed. An example is *Dinamica* (Filho *et al.*, 2002). Such generators are based on a theoretical understanding or hypotheses of the relevant processes that cause landscape patterns, and they are helpful in determining how real agricultural landscapes and their dynamics emerge. The resulting landscapes allow more explicitly stated hypotheses and can be used as the basis for addressing specific questions, e.g., pattern-process interaction (Schröder & Seppelt, 2006), land-use change prediction, the fate of specific species or of biodiversity in general, and effects on ecosystem functioning and resilience.

In this review, we examine existing ALGs that are able to automatically generate an agricultural landscape with given features. Our two main motivations are as follows: (1) There is an increasing need to evaluate the driving forces behind and the extent and consequences of land use and land cover change. Recent EU policies, for example, aim at increasing biodiversity by requiring environmentally friendly farming practices; the so-called "greening" of farming (European Commission Regulation (EU) No 1307/2013). Whether or not such policies will in fact increase biodiversity is an open question (Pe'er et al., 2014). The usefulness of such policies requieres two things: a realistic simulation model and realistic landscape, where both different eco-regions and policies can be represented. In the case of honeybees, the model BEEHAVE (Becher et al., 2014) is such a model, and so far only the software tool NePoFarm (Horn, 2017) exists, that takes the structure of a given landscape, important from GIS, implements different crop diversity and rotation scenarios, and then explores how different greening measures, such as flower strips, affect the resilience and persistence of honeybee colonies. With a kind of generic ALG, one could also vary the structure of the landscape, for example regions with mostly small or large fields, or landscapes with and without semi-natural habitat or hedges. The same landscapes could then also be used for exploring the performance of bumblebees (Becher et al., 2018), wild bees (Everaars et al., 2018), or completely different taxa, where other landscape features might be important. (2) Our second interest is in finding the main features and criteria of generic software tools that could generate agricultural landscapes with more comprehensive and variable configuration and composition by using a modular design. The landscapes generated by such generic ALG could be used as input for any specific LUCC model addressing specific questions. This would generate coherence and synergies across individual studies that currently do not exist.

2 Methods

Because there is no consistent term for agricultural landscape generators so far, in our literature search, we used the following search terms: “ecological model*” AND crop* AND landscape*, “landscape model*” AND crop*, “landscape model*” AND neutral, “landscape generat*” AND crop*, “landscape generat*” AND neutral, “landscape simulat*” AND crop*, and finally, “landscape simulat*” AND neutral. Using these terms in Web of Science (Clarivate Analytics) led to 186 publications that cover a broad range of approaches, aspects of agricultural landscape simulations, and fields of application.

The most important selection criterion for including an ALG approach in our review was the user-defined automatic generation of agricultural land cover patterns, which can serve as a input for dynamic simulation models. We excluded landscape generators that do not encompass any characteristics of agricultural landscapes or did not include the land-use type agriculture. We also did not include geographical information system (GIS) models, remote sensing land use and land cover change (LUCC) models and agent-based land use models (ABMs) (Lambin *et al.*, 2000, Agarwal *et al.*, 2002, Schulze *et al.*, 2017), although they too allow landscape maps to be altered. Firstly, these models usually work with real landscapes and data, and secondly, landscape changes in composition and configuration are mostly based on the outcome of decision models and not on automated procedures, which were mandatory according to our definition of a landscape generator.

We examined the following features of the ALGs: the specific aims, the method of landscape generation in terms of configuration and composition, the validation, the application regarding policy measures, and the software availability. Regarding landscape composition, we analysed which method for crop generation and allocation of crops to the fields has been applied. The more complex an ALG is, the more compositional details can be varied, such as crop types, fringe structures, and other land-use types. We explored which of them have been implemented so far in the context of specific case studies. We also compiled a brief overview of existing additional software tools that allow crop rotation to be implemented for a given landscape configuration.

Moreover, because it is important for allowing the exploration of ecological questions in agricultural landscapes, we examined whether natural and semi-natural habitats were included. The same applies to the coupling with ecological population models to allow the analysis of landscape effects on species. Because enormous potential of agricultural landscape simulations lies in the evaluation of management and policy measures, we investigated whether and how such scenario analyses were carried out. Finally, programming language, software application, documentation, and availability of the models were determined.

Based on these findings, we finally derived and outlined the requirements for generic ALGs. In particular, we wanted to identify which landscape features are essential or optional for user-friendly agricultural landscape generation and which conclusions can be drawn for the model and software architecture.

3 Results

3.1 Existing ALGs

The list of publications containing solutions for generating such agricultural landscapes was short (Fig. 1). We identified ten relevant peer-reviewed articles published between 2006 and 2017. These ten publications relate to a total of seven ALG approaches, as several landscape generators were described or applied in more than one publication. We added four more relevant publications describing one of the selected ALG approaches more precisely. To display even more possibilities, we added the non-ISI-listed Scottish Natural Heritage Commissioned Report No. 692 (Begg & Dye, 2015) describing another ALG approach. Altogether, we found eight ALG approaches.

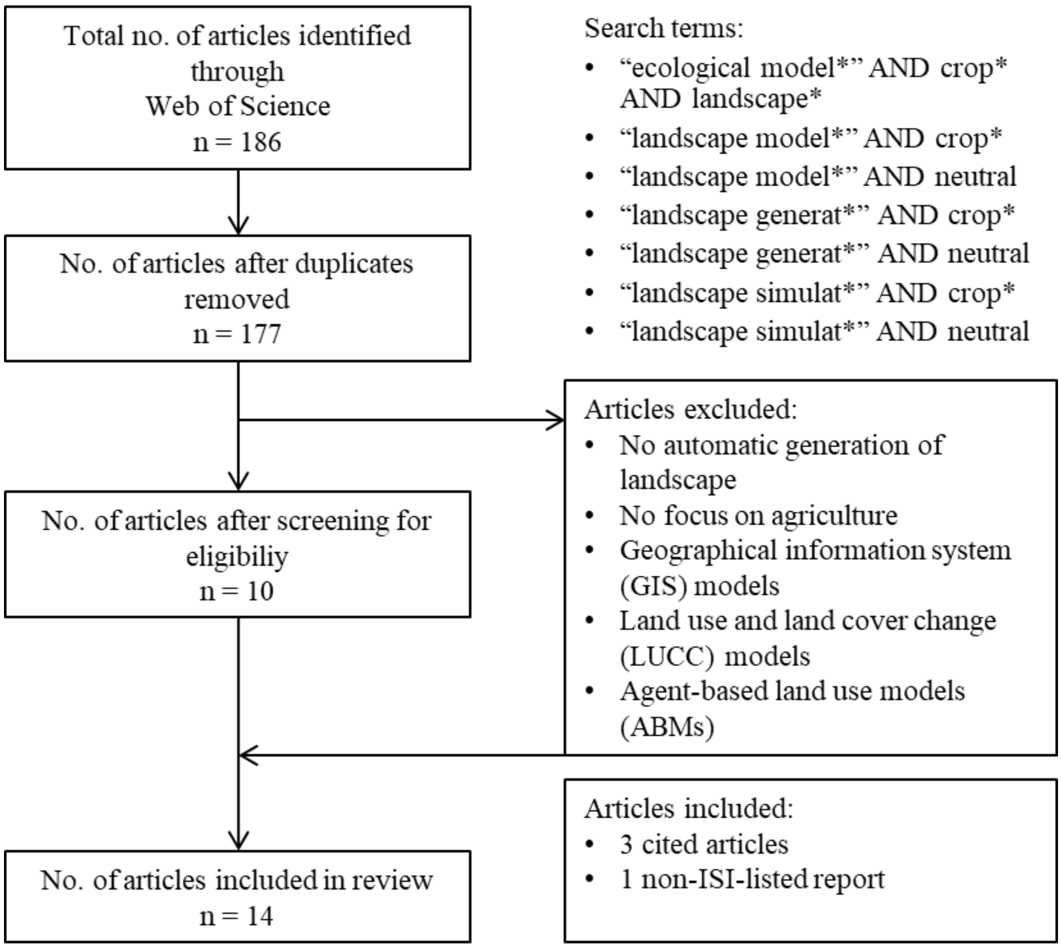

Figure 1: Flowchart of literature search and selection process

An overview of the reviewed ALG approaches is given in Tab. 1 and Fig. 2. The approaches differ widely in their method of landscape generation and agricultural details; therefore it was difficult to give an integrated and structured overview and to derive general concepts. To emphasise and better compare the range of existing approaches for each feature, we present our results feature by feature, not model by model. Nevertheless, Table 2 provides an overview of the reviewed ALGs. The ALGs presented here have mainly been published within the last 10 years (Tab. A1). This makes it a young field of research with few approaches so far, and most of them are described and/or applied in a small number of publications only. For each ALG we show one example output map to get an impression of the application possibilities, even if there may be many more output options available.

Table 1. General information on existing agricultural landscape generators (ALGs), i.e., models or programs that have the option to automatically and systematically generate virtual agricultural landscapes with given features in terms of configuration and composition.

Publication	Name	Nr. of publications*	Time span	Language	Validation
Begg & Dye (2015)	AgBioscape	1	2015	C#	n.i.
Engel *et al.* (2012)	Landscape generator	2	2012-2014	C++	n.i.
Gaucherel *et al.* (2006)	DYPAL	8	2006-2017	C++	yes
Inkoom *et al.* (2017)	SG4GISCAME	2	2017	n.i.	yes
Le Ber *et al.* (2009)	GENEXP-LANDSITES	4	2008-2013	Java	yes
Papaïx *et al.* (2014)	*Ddal* Landscape simulator	1	2014	n.i.	n.i.
Pe'er *et al.* (2013)	*G-RaFFe*	3	2011-2018	n.i.	yes
van Strien *et al.* (2016)	Landscape Generator (LG)	4	2011-2016	Java	yes

* Listed in Table 4.
n.i. No information available.

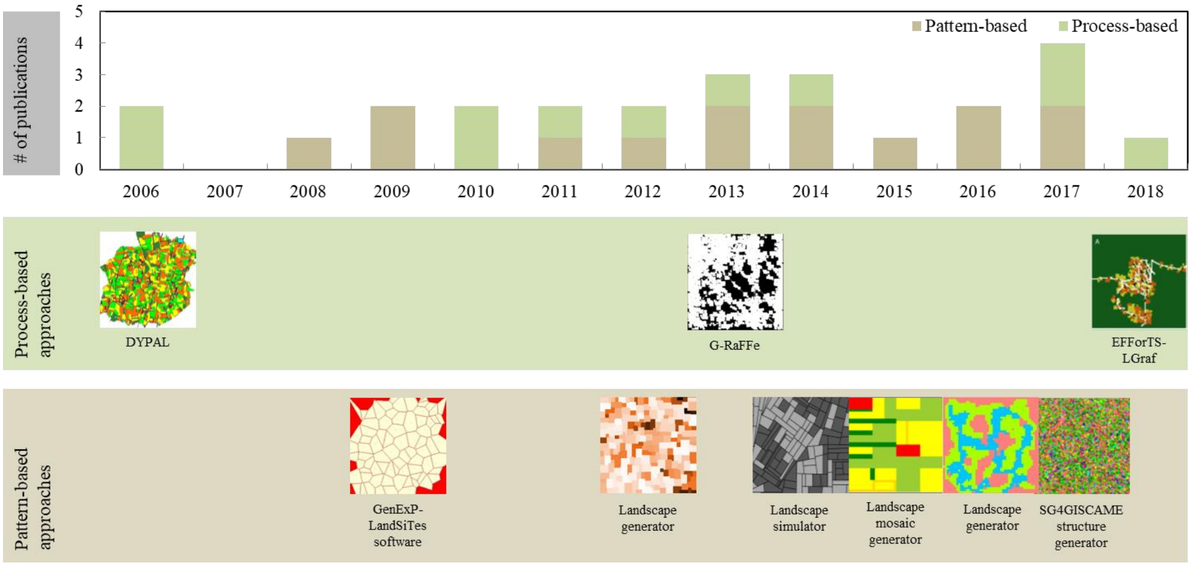

Figure 2. Evolution of the reviewed ALG approaches sorted by date of the first publication. DYPAL (Gaucherel *et al.*, 2006), G-RaFFe (Pe'er *et al.*, 2013), EFForTS-Lgraf (Dislich *et al.*, 2018), GenExP-LandSiTes software (Le Ber *et al.*, 2009, Le Ber & Mari, 2013), Landscape generator (Engel *et al.*, 2012, Everaars *et al.*, 2014), Landscape simulator (Papaïx *et al.*, 2014), Landscape mosaic generator (Begg & Dye, 2015), Landscape generator (van Strien *et al.*, 2016), SG4GISCAME structure generator (Inkoom *et al.*, 2017).

3.2 Aims

"Aim" here refers to the purpose of an ALG. There are various motivations for developing and applying computer-generated agricultural landscapes. However, there are two general objectives, which are not exclusive but lead to different foci and, hence, different designs of an ALG (Fig. 3).

Purpose of the ALG

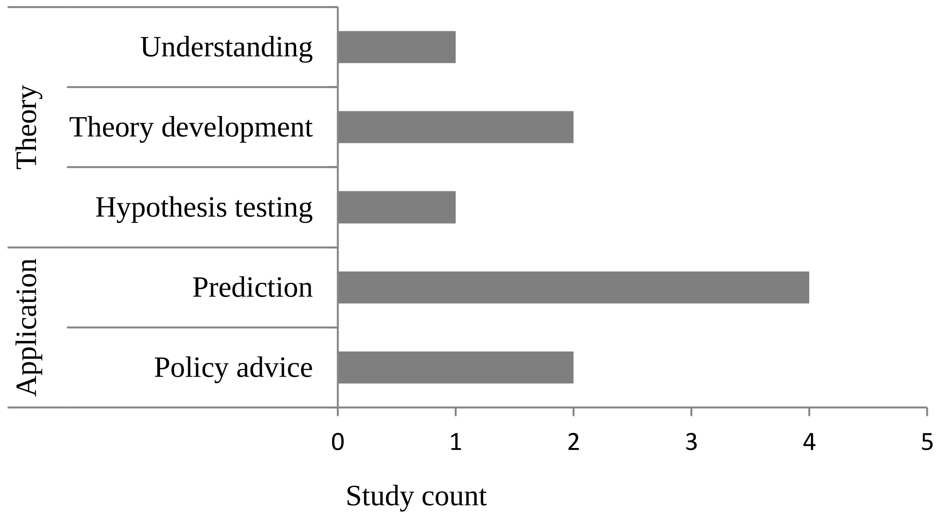

Figure 3. Overview of purposes of the reviewed ALGs. Numbers do not have to add up to the total number of reviewed papers, because multiple entries (e.g., multiple model purposes) are possible.

A first general objective in ALGs is to gather theoretical understanding and foster theory development. For example the software platform *DYPAL* (*Dynamic PAtchy Landscape*) by Gaucherel *et al.* (2006), was designed to analyse processes that drive changes in landscape patterns and ecological functioning, e.g., field aggregation and land use allocation. This not only allows investigating the relationships between landscape-shaping processes and the resultant landscape pattern but also how agricultural landscapes affect ecological processes. The generated landscapes of *G-RaFFe* (Pe'er *et al.*, 2013), which mimics the processes in which roads penetrate into natural habitats, serve as templates for theoretical analyses and the testing of hypotheses. In the same context, neutral models of agricultural landscapes, for example the vector-based model *GenExP-LandSiTes* (Le Ber *et al.*, 2009), are used for studying agro-ecological processes within a certain variability of landscape patterns.

A second objective of ALGs is the application of computer-generated agricultural landscapes for spatial simulation models in order to predict the ecological consequences of landscape dynamics or to give policy advice. The landscape generator by Slager & de Vries (2013) and it's updated version (van Strien *et al.*, 2016), for example, was developed to investigate the influence of landscape patterns on spatial ecological processes. The landscapes generated are used to analyse the effects of changes in the landscape configuration and composition on biodiversity and conservation issues at the landscape scale. Therefore, they are usually coupled with models of specific species to investigate habitat suitability and population dynamics (Tab. 2). The aim of the AgBioscape modelling framework (Begg & Dye, 2015) is to explore options of managing arable landscapes and their ability to enhance functional biodiversity and devise conservation recommendations. The landscape generator of Engel *et al.* (2012) and its updated version (Everaars *et al.*, 2014), were developed to simulate changes in land use mosaics caused by bioenergy scenarios such as the thinning and spatial agglomeration of crops and to analyse their effects on the abundance of different farmland bird species. The modelling framework *Ddal* (disease dynamics in agricultural landscapes) aims at exploring the effects of landscape configuration and composition on the development of an epidemic (Papaïx *et al.*, 2014). The structure generator SG4GISCAME (Inkoom *et al.*, 2017) aims at giving inputs for spatial ecosystem service assessment in data-scarce areas.

The range of aims shows that the purpose of the approaches presented here is different. Thus, some ALGs are a simulation model by themselves that can be used as a stand-alone tool to investigate certain questions or processes, e.g. on the drivers producing a landscape pattern. The result, however, is a computer-generated landscape pattern that can be used further for analysing respective consequences.

3.3. Configuration

“Configuration” refers to the size, shape, and spatial arrangement of structural elements of agricultural landscapes such as fields, semi-natural habitats, or hedgerows. ALGs generate landscape configurations that are either pattern-based (i.e., provide heuristic design rules and algorithms for reproducing typical structural characteristics of agricultural landscapes without addressing the pattern-forming driving forces) or are process-based by explicitly including underlying ecological or land use processes. Overall, there are more pattern-based approaches (6) than process-based approaches (2). In the remainder of this section, we first give a general overview of the different features of the ALGs and then present each ALG in detail (Tab. 2).

Pattern-based approaches

The landscape mosaic generator of the AgBioScape model platform generates a pattern of rectangular fields and narrow fringe structures such as hedges, whereby the size, shape and clustering of the fields can be controlled (Fig. 4). The generator from (Engel *et al.*, 2012, Everaars *et al.*, 2014) also generates rectangular fields of variable size, which are distributed irregularly and randomly in the landscape (Fig. 5). Other landscape structures cannot be displayed. The landscape generator described by van Strien *et al.* (2016) is more complex than the two previous approaches, as the user can set target values of landscape metrics that quantify landscape configurations, e.g., the maximum perimeter or shape of the bounding box of a patch (Fig. 6). Therewith, the linear shape of hedgerows can be depicted, and the configuration in general becomes more realistic.

In contrast to these grid-based approaches, there are also pattern-based models that work on the basis of vectors. Raster (grid-based) and vector landscapes differ fundamentally in terms of their spatial composition. Raster landscapes with one grid cell being the smallest unit are particularly suitable for gradual landscape dynamics and continuous processes. In vectorial landscapes, patches, typically polygons of varying sizes, are described by the exact coordinates of their bounding vertices. Since landscapes, especially agricultural landscapes, are strongly characterised by patches and corridors (Forman & Godron, 1981, Turner, 1989), the vector-based approach is very well suited for this type of landscape. Nevertheless, it is much less used because the geometry and algorithms are more complex.

The vectorial approaches of Le Ber *et al.* (2009), Inkoom *et al.* (2017) and Papaïx *et al.* (2014) are based on tessellation methods that are used to manage sets of polygons. Tessellation starts from a point pattern and determines polygons based on distances to the closest neighbour points without overlapping or holes. The spatial distribution is determined by the distribution of the tessellation seeds (point pattern). If a landscape mosaic is based on the seed distribution of a real landscape, the spatial pattern will be similar to the real landscape (Le Ber *et al.*, 2009). By controlling the size, shape and clustering of the polygons, a landscape mosaic develops and different land uses can be assigned to each patch or field.

Landscape mosaic generator (Begg & Dye, 2015)

The AgBioscape modelling framework (Begg & Dye, 2015) integrates a landscape mosaic generator and a population module to simulate interactions between a range of species and cropping systems, management and landscape characteristics. The landscapes consist of fields and the boundaries between them (Fig. 4). Fringe structures such as grass margins and hedgerows can be depicted. By specifying the height, width, and total number of fields, the size, shape and clustering of the fields can be controlled. The generator continually subdivides a two-dimensional space to produce a mosaic of rectangular fields. The algorithms for generating the landscape structure contain stochastic elements, leading to a spatial variation of landscape patterns.

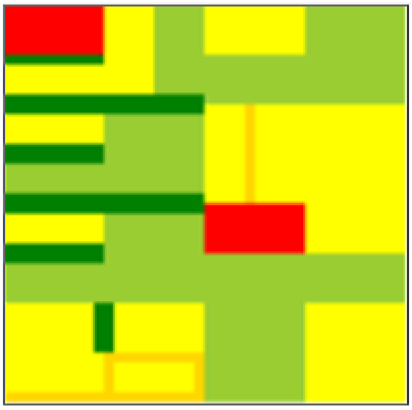

Figure 4. Example of a 4 km² landscape generated by the software platform AgBioscape (Begg & Dye, 2015) with wheat crops (yellow), grass ley (light green), woodland (red), grass margins (orange), and hedgerow (dark green).

Landscape generator (Engel et al., 2012, Everaars et al., 2014)

A given mean field size is used by this generator to create landscapes that consist of an irregular and randomly distributed mosaic of agricultural fields with varying shapes, sizes and edge lengths (Fig. 5). The landscapes are 9 km² in size and are divided into grid cells with a 4 m grain size (750 by 750 grid cells). In a first step, fields of a predefined size are placed randomly on the main grid until the whole space is occupied. Afterwards, a correction algorithm replaces all fields that are too small by merging them with neighbouring fields. Field margins and in-field strips can be implemented.

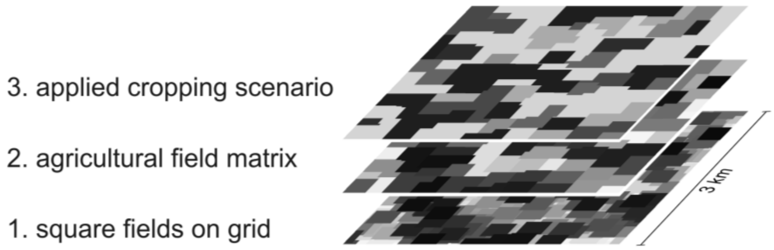

Figure 5. Visualization of different calculation steps in the model of Everaars *et al.* (2014). A given mean field size is used by a landscape generator to produce a natural-looking mosaic of agricultural fields (Layers 1 and 2; grey values chosen arbitrarily). A given cropping scenario determines which crops are present in the landscape and at what proportions (Layer 3; each crop represented by a shade of grey).

Landscape generator (van Strien et al., 2016)

This generator integrates different landscape metrics, quantifying the landscape configuration and composition. The distribution and clustering of land-use classes as well as the proportion of adjacent landscape components can be determined (e.g., hedgerows around fields). The required input of the generator is configuration files with target values for the landscape metrics and an initial input raster in the ASCII format that can either be a random percolation map or an existing landscape. In random percolation maps, each raster cell is randomly assigned to a predefined land-use class with a certain probability (Gardner *et al.*, 1987). It can be created with several GIS and programming languages (e.g., R, Python). The following landscape metrics can be varied on the class level or patch level: number of patches for a certain land-use class, area of a certain patch, maximum perimeter, contact with another land-use class, shape of the bounding box of a patch, total area within a bounding box occupied by a certain patch and the rectangular criterion. The generator uses an optimisation algorithm to "find" landscapes of which the composition and configuration correspond to the target values of the landscape metrics. It does so by iteratively swapping raster cells and determining whether the new landscape is an improvement with regard to the target values. The algorithm contains stochastic components, so the generation of two identical landscapes with the same input settings is not possible. In return, it is feasible to generate landscape series in which single landscape metrics are varied (Fig. 6).

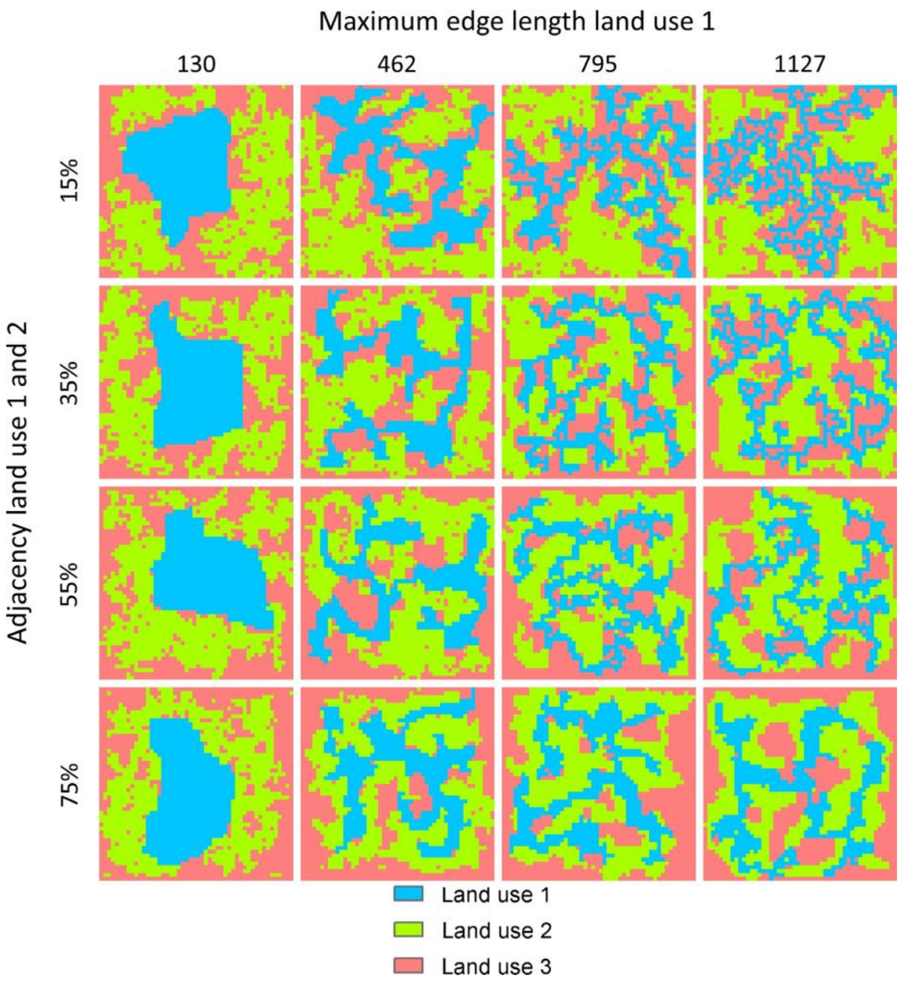

Figure 6. Landscape series generated by the landscape generator of van Strien *et al.* (2016). Landscapes in this series have three land-use categories, various maximum edge lengths of land use 1 and different proportions of the adjacency between land uses 1 and 2. Note that for agricultural landscapes, "land use" would actually refer to "land cover".

GenExP-LandSiTes software (Le Ber et al., 2009, Le Ber & Mari, 2013)

This software simulates neutral agricultural landscapes to explore the variability of landscape characteristics and the variation in the geometry of fields (Fig. 7). An irregular patchy landscape mosaic without fringe structures develops. The field patterns are obtained by using two different tessellation methods, the Voronoi tessellation and a rectangular tessellation, which make it possible to control the size, number and the shape of fields. The user can choose the kind of seed distribution (original, simulated or random), the tessellation type (Voronoi or rectangular) and the cropping pattern distribution (random or stochastic). The model provides a library to calculate basic landscape descriptors (field area, perimeter, number of vertices, centroid and shape). File export is possible in raster or vector format and with shape files.

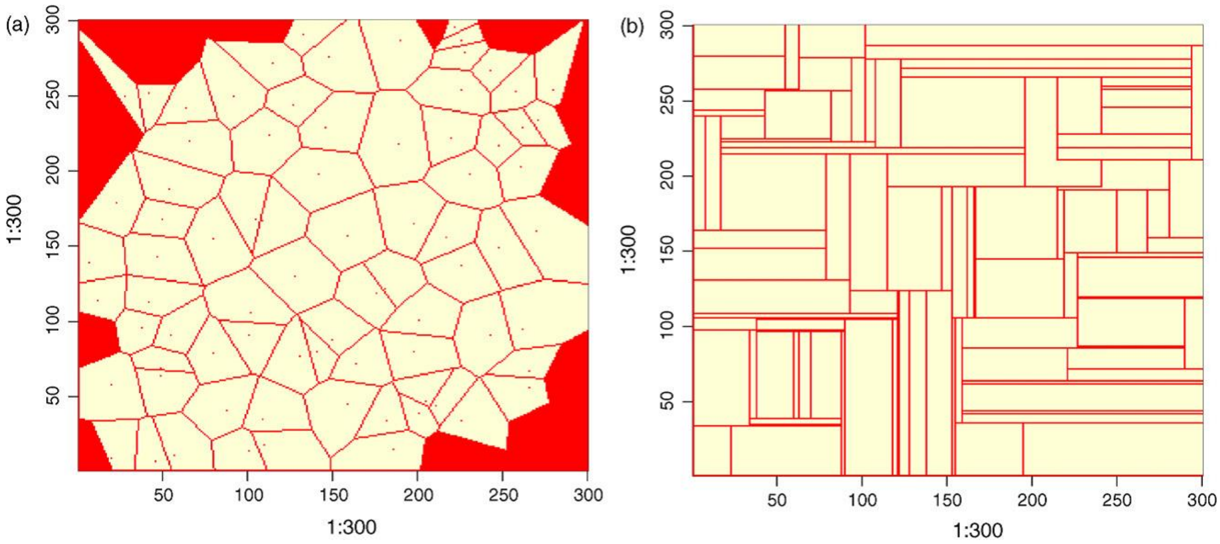

Figure 7. Two different tessellations based on sets of seeds generated by the GenExP-LandSiTes software (Le Ber *et al.*, 2009): the Voronoi tessellation (a) and the rectangular tessellation with only T-vertices (b). Infinite Voronoi cells were eliminated.

SG4GISCAME structure generator (Inkoom *et al.*, 2017)

The SG4GISCAME structure generator, which is a module within the GISCAME software, uses Voronoi tessellation and different algorithms to produce realistic landscape patterns (Fig. 8). First, the landscape is separated into a number of triangles from a regular midpoint (triangulation). The edge, shape and size of the polygons can be altered through initial split and tolerance levels and spatial resolution. As a second step, the irregular triangles are merged to form more-realistic polygons on the basis of a user-designed or random process (merging). Finally, users can alter or refine the output geometry of the landscapes using either a manual distribution option or a cellular automaton algorithm (refinement). The resulting vector data are transformed into raster data, and the output is an ASCII text file. A set of landscape pattern metrics is implemented to assess variations in landscape configuration and composition: patch cohesion, average patch shape, contagion, area weighted mean shape index, and landscape patch index.

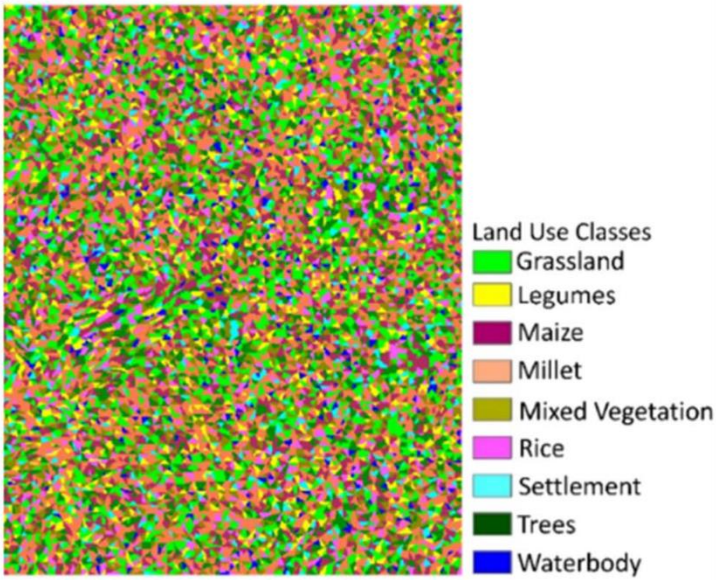

Figure 8. Application of a cellular automaton algorithm on a simulated landscape with 25 m resolution in SG4GISCAME (Inkoom *et al.*, 2017).

Landscape simulator (Papaïx et al., 2014)

The simple landscape simulator within the *Ddal* framework generates a landscape mosaic based on the T-tessellation simulation algorithm developed by Kiêu *et al.* (2013) (Fig. 9). The input parameters number of fields, field surface average, field surface variability, and the square-like form of fields can be determined. As a result, it is possible to influence the degree of fragmentation of the landscape and to prevent triangular fields. The simulator is coupled with a pathogen population dynamics model.

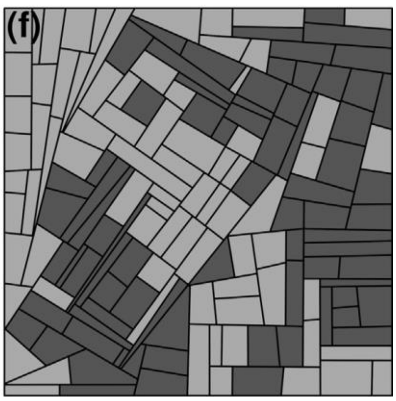

Figure. 9. Example of simulated landscape structures generated by the *Ddal* landscape simulator (Papaïx *et al.*, 2014) with two host types (light (50%) and dark (50%) grey) dispatched among a 155-field landscape.

Process-based approaches

The following two approaches work with explicit spatial locations determining the process being modelled. By using the modelling platform *DYPAL*, many different processes can be investigated at the landscape level, e.g., the spatial aggregation and distribution of fields and hedges, land use allocation, and land use rotations. The landscape processes are applied to landscape units by using one or more algorithms. Even though most of the previous applications of *DYPAL* are based on real landscape patterns, Gaucherel *et al.* (2006) have implemented neutral models (Patchy Landscape Neutral Models) in the software platform. In contrast to the complex modelling platform *DYPAL*, the *G-RaFFe* model (Pe'er *et al.*, 2013) generates landscapes based on a single process: forest fragmentation by roads and the generation of agricultural fields.

DYPAL (Gaucherel et al., 2006)

DYPAL, formerly known as *L1* (Gaucherel *et al.*, 2006), is a modelling platform for generic landscape modelling on various scales and landscape types (field, farm, and region). Landscape processes such as hedgerow planting and removal, when applied to landscape units, lead to evolution in the composition and configuration of the landscape. The ALG can simulate the patch dynamics of fields as well as dynamic fringe structures such as hedgerows (Fig. 10). The patches are defined as polygons, but a pixel definition of the polygons was kept to be able to simulate continuous processes. The required input parameters depend on the process being modelled. The platform was designed around a kernel that provides an organisational data structure and a generic landscape structure. Several libraries provide specific algorithms permitting the handling of sets of points, linear networks or a mosaic of adjacent polygons. The user can freely choose the data structures, the driving decisions and processes, the simulation steps, the level of detail of the description, and the chosen scales. Several further developments of the software platform and a number of applications have been made so far. Gaucherel *et al.* (2010) and Gaucherel *et al.* (2012) implemented additional mathematical models (formal grammar equations) into the platform to mechanistically simulate landscape dynamics. Bonhomme *et al.* (2017) implemented further configurationally changes of patches that enable all possible operations and combine them into a coherent mathematical framework.

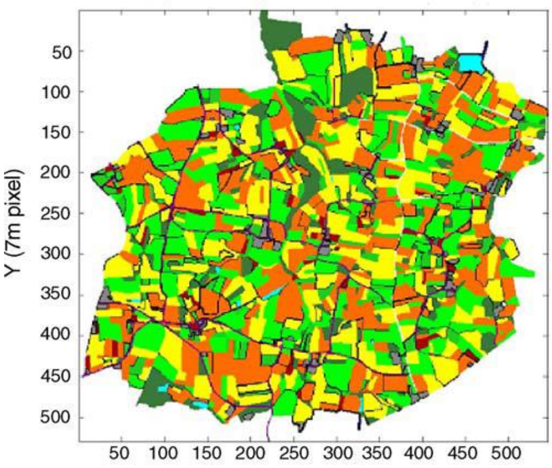

Figure 10. Example of an agricultural landscape mosaic generated by the modeling platform DYPAL (Gaucherel *et al.*, 2006)

G-RaFFe (Pe'er et al., 2013)

The simple process-based simulator *G-RaFFe* is able to generate the spatial patterns of agricultural fields embedded in a natural habitat (esp. forest) that emerge from forest fragmentation by roads (Fig. 11). Three main parameters determine the generated landscapes: the habitat cover, the number of roads crossing the landscape, and the field size. An additional parameter, maximum field disconnection, specifies whether and to what distance agricultural fields can be detached from roads or other fields. The model starts with a 100% forest landscape and then creates roads that lead straight through the landscape, converting the forest into "non-forest". Roads are generated until the number of roads reaches the desired number, unless the forest cover reaches the target value specified by the user. Once all roads have been generated, agricultural fields are separated from them by a random movement of simulated "farmers". All fields have a square shape (same length and height), the size of which is derived from a uniform distribution between one and the maximum length specified by the user. Field expansion is a per-step process that can stop when the potentially converted cells are beyond the map extent or when the desired forest cover is reached. EFForTS-LGraf (Dislich *et al.*, 2018) is an extended version of *G-RaFFe*, which additionally includes two different land uses and households (Fig. 11). Households can own several fields of different sizes with different land uses.

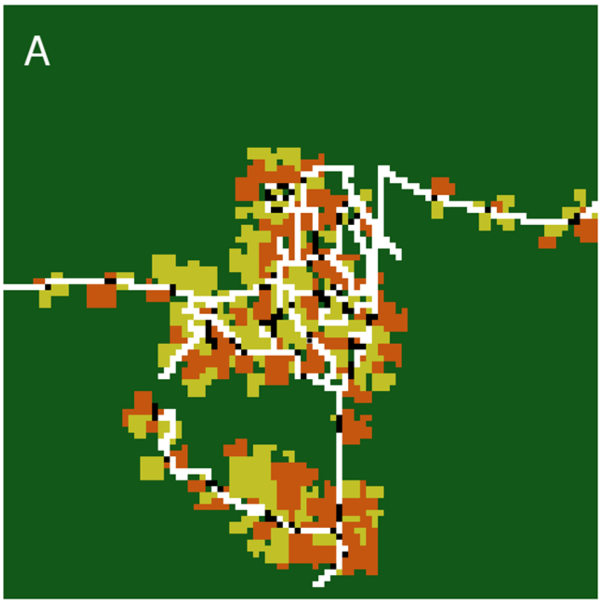

Figure 11. Virtual landscape generated by EFForTS-Lgraf (Dislich *et al.*, 2018), an extended version of *G-RaFFe* (Pe'er *et al.*, 2013). Roads are marked in white, household home bases in black, oil palm plantations in orange, rubber plantations in dark yellow. Dark green is the area which is not used for agriculture.

Table 2. Configurational and compositional details of existing agricultural landscape generators (ALGs). The scope of application of an ALG is demonstrated with specific case studies. Thus, the table shows only a selection of compositional details; usually, many more application types are feasible.

Publication	Configurational Details			Compositional Details						Population module avail.
	Type of landscape generation	Map type	Input parameter	Case study	Study region	Crop types	Crop generation	Fringe structures	Semi-natural habitats	Taxa
Begg & Dye (2015)	Pattern-based	Grid-based	Field height Field width Number of fields	Begg & Dye (2015)	GB	Cereal crop Broad-leaf crop Wild bird seed mix Grass ley	Stochastic	Grass margin Hedgerow	Conservation headland Forest	*Perdix perdix*, Carabid, Parasitoid, Aphid
Engel *et al.* (2012)	Pattern-based	Grid-based	Mean field size Number of crops Crop types	Everaars *et al.* (2014)	DE	Alfalfa Barley Beets Grassland Maize Oat Oilseed rape Potato Summer Rye Winter Rye Ryegrass Set-aside Sunflower Triticale Wheat	Stochastic	Field margin In-field strip	Integrated Biodiversity Area	*Alauda arvensis, Motacilla flava, Miliaria calandra, Vanellus vanellus*
Gaucherel *et al.* (2006)	Process-based / Pattern-based	Vector-based Grid-based	Depends on the process being modeled	Gaucherel *et al.* (2006)a, Gaucherel *et al.* (2006)b	FR	Maize Fallow land Permanent grassland Temporary grassland Cereal field Vegetable field	Stochastic, Gibbs process	Hedgerow River Road	Forest	-
Inkoom *et al.* (2017)	Pattern-based	Vector-based Grid-based	Cell size Initial split Initial split tolerance Split algorithm Split algorithm	Inkoom *et al.* (2017)	GH	Grassland Legumes Maize Millet Rice	Statistic, Stochastic	-	Mixed Vegetation Trees	-

			tolerance Area size Centre point Median line							
Le Ber *et al.* (2009)	Pattern-based	Vector-based	Seed choice (original, simulated, random) Tessellation choice Cropping pattern choice (random, stochastic)	Le Ber *et al.* (2009)	FR	Coupling with crop model	Stochastic	-	-	-
Papaïx *et al.* (2014)	Pattern-based	Vector-based	Number of fields Surface of field Field surface variability Square-like field shapes	Papaïx et al. (2014)	-	Coupling with crop model	Stochastic	-	-	Plant pathogen
Pe'er *et al.* (2013)	Process-based	Grid-based	Map extent Desired habitat cover Number of roads Field size Max. field disconnection Density of farming households Household size Fraction of land uses (Road width)	Pe'er *et al.* (2011), Pe'er *et al.* (2013), Dislich *et al.* (2018)	AR, PY, BR, ID	Oil palm, Rubber	n.i.	Road	Forest	Hypothetical bird species
van Strien *et al.* (2016)	Pattern-based	Grid-based	Proportion of landscape component Number of patches Maximum edge length Patch size Patch maximum perimeter Patch edge contrast Shape of the bounding box of a patch	Slager & de Vries (2013)	NL	Maize	-	Hedgerow	-	-

3.4 Composition

"Composition" here refers to the assignment of landscape elements (configuration) to different land cover types or structures. However, not all of the generators reviewed allow a finer differentiation in this respect (Tab. 2). The generated landscapes of *G-RaFFe* consist only of forest and agricultural fields, in which 'agricultural field' is not further specified. However, agricultural fields can be covered with oil palm or rubber plantations in the extended version EFForTS-LGraf. The composition of the generated landscape described by van Strien *et al.* (2016) is defined by various land use categories. Regarding agricultural landscapes, one case study on maize farmland with hedgerows has been carried out so far (Slager & de Vries, 2013).

GenExP-LandSiTes and presumably *Ddal* can be coupled with external crop rotation models such as *CarrotAge* (Le Ber *et al.*, 2006), whereby a wide range of crops can be taken into account. The data mining software *CarrotAge* is based on high-order hidden Markov models for analysing spatio-temporal cropping patterns. Many crop rotation approaches work with Markov models to achieve a change in landscape configuration or composition. Markov models provide matrices with transition probabilities between different crop types; similar matrices have been widely used for representing succession in forests (Horn *et al.*, 1975). These models are descriptive, not mechanistic, but they can be a starting point for realistic depictions of existing composition dynamics. Different policies can be implemented by manipulating the transition probabilities. In *Ddal* (Papaïx *et al.*, 2014), no explicit crop types have been implemented so far. The number of land use types (host types), the proportions they cover and their level of spatial aggregation can be varied by applying a stochastic algorithm.

The other four generators work with pre-defined crop types or land use classes, which are distributed to the fields mostly according to stochastic rules of crop successions. The landscape mosaic generator described by Begg & Dye (2015) describes crop rotations as first-order Markov chains (Usher, 1992). Only four crop types have been implemented so far, but different land-use types such as hedgerows, flowering margins and conservation headlands can be chosen. The number and type of 15 different crops can be varied in the landscape generator of Everaars *et al.* (2014) to build different cropping scenarios. Crops are allocated randomly or with a probability that equals the relative proportion to the agricultural fields for multiple subsequent scenarios. In addition, fields can be allocated with integrated biodiversity area (IBA), defined as semi-natural habitat. *DYPAL* simulates deterministic or stochastic crop successions within farms, while other landscape elements such as hedgerows, forests and rivers can be depicted. In Gaucherel *et al.* (2006), land-use types are assigned to patches using the Gibbs process, which is derived from statistical physics (Caldiera & Presutti, 1974) and describes the local interactions between landscape units. It is also possible to apply the Gibbs process to landscape configurations (Gaucherel, 2008). This can be achieved by editing the shape or size of the landscape patches or by selecting a Gibbs pair function to constrain the relative positions of the seeds. In SG4GISCAME (Inkoom *et al.*, 2017), a large number of land use classes can be specified as well as their relative amount. The distribution is based on transition probabilities and tolerable or intolerable neighbourhood settings. In addition to crop types, five of the ALGs can depict semi-natural habitats such conservation headlands (Begg & Dye, 2015), integrated biodiversity areas (Everaars *et al.*, 2014) and forest (Gaucherel *et al.*, 2006, Pe'er *et al.*, 2013, Begg & Dye, 2015, Inkoom *et al.*, 2017). Four of the ALGs were coupled with population modules (Pe'er *et al.*, 2011, Everaars *et al.*, 2014, Papaïx *et al.*, 2014, Begg & Dye, 2015).

Additional crop rotation tools

As alternatives to *CarrotAge*, a range of other crop rotation tools exist that can be used as a supplement to a landscape generator. Some useful examples are presented as follows.

LandSFACTS (Landscape Scale Functional Allocation of Crops Temporally and Spatially) allocates a crop to each field for each simulation year in a GIS shape file (Castellazzi *et al.*, 2008, Castellazzi *et al.*, 2010). Fields are represented as polygons in vector format with fixed boundaries. The model is based on a stochastic process using the probabilities of crop-to-crop transitions (Markov chains) and rule-based constraints.

The linear optimisation model *CropRota* (Schönhart *et al.*, 2009, Schönhart *et al.*, 2011) integrates agronomic criteria and observed land use data at field, farm, or regional scales in order to generate typical crop rotations for the particular scale.

ROTAT (Dogliotti *et al.*, 2003) is a computer program that combines crops from a predefined list to generate all possible different rotations. A number of filters or rules based on explicit agronomic criteria and expert knowledge can be controlled by the user.

ROTOR (Bachinger & Zander, 2007) is a static rule-based model for generating and evaluating site-specific and agronomically sustainable crop rotations for organic farming systems in central Europe. ROTOR requires as input data field specific soil data, mean annual precipitation and mean precipitation during the winter half year.

NePoFarm (Horn, 2017) is a landscape generator that creates farmland scenarios as input files for the honeybee model BEEHAVE (Becher *et al.*, 2014). Field patches are allocated randomly in two British template landscapes and are characterised by crop type identity and crop diversity (crop type number and relative abundance). NePoFarm is based on GIS maps and is implemented in the freely available programming language R.

3.5 Validation

"Validation" here refers to testing whether landscapes generated by an ALG reproduce realistic landscape features. To validate the generated landscapes, five of the reviewed approaches compared certain landscape metrics with real landscapes (Gaucherel *et al.*, 2006, Le Ber *et al.*, 2009, Pe'er *et al.*, 2013, van Strien *et al.*, 2016, Inkoom *et al.*, 2017). Three of them (Pe'er *et al.*, 2013, van Strien *et al.*, 2016, Inkoom *et al.*, 2017) calculated the relevant landscape metrics of the real and computer-generated landscapes by using *FRAGSTATS* (McGarigal *et al.*, 2012). The calculated landscape metrics can also be used as input parameters for the ALG to approximate the computer-generated landscape to the real one. As a result, the generated landscapes preserve the main characteristics of the original landscape while being configurationally different. Inkoom *et al.* (2017) used a Turing Test to explore expert visual judgement in comparing neutral landscapes to real landscapes. The publications of the other approaches did not contain any evidence of validation.

3.6 Evaluation of policy measures

Here, we summarise to what extent and how ALGs were used to evaluate existing or prospective policy measures addressing agricultural landscapes. Everaars *et al.* (2014) simulated future scenarios of different land-use changes due to policy-making and economic aspects of bioenergy production. The effectiveness of different mitigation strategies on farmland birds was analysed on the basis of the generated landscapes. *DYPAL* has been developed, inter alia, to assess the environmental consequences of agricultural policies affecting processes that drive changes in landscape patterns and ecological functioning. It was used to evaluate simplified common agricultural policy (CAP) and CAP reform decisions such as changing maize and cereals to fallow and temporary grassland (Gaucherel *et al.*, 2006). Houet *et al.* (2010) used *DYPAL* to simulate plausible future states of landscape features such as hedgerows, riparian wetlands, and agricultural land covers based on hypotheses about future land management related to European policies. The AgBioscape modelling approach has the intention to develop an impact assessment and decision support tool for land management options, including agri-environmental schemes (AESs). The potential of such a tool is to assist land managers and policy makers in identifying effective management options. The landscape generator by Slager (2011) was initially developed to generate plausible landscape configurations for participatory spatial plan-making. The study focuses on the construction of so-called policy (plan) scenarios as a fundamental activity in spatial plan-making.

3.7 Availability of the ALGs

ALGs that have an executable version are mostly openly available on the internet (Tab. 3). However, only *DYPAL* has open source code. Almost none of the ALGs are documented in detail; instead, only summary descriptions of the algorithms and data used are provided, which also limits the level of detail by which we could characterise the ALGs above.

Table 3. Availability of agricultural landscape generators (ALGs).

Publication	Name	Software application	Availability	Webpage / Download link	Code open-source	Documentation
Begg & Dye (2015)	AgBioscape	Y	not open to the public		N	Annex 1
Engel *et al.* (2012)	Landscape generator	N	not open to the public		N	Appendix S1
Gaucherel *et al.* (2006)	DYPAL	Y	free to use	http://amap-collaboratif.cirad.fr/pages-logiciels/?page_id=70	Y	n.i.
Inkoom *et al.* (2017)	SG4GISCAME	Y	usage fee	http://www.giscame.com/giscame/english_giscame_giscame_suite_sg4giscame.html	N	N
Le Ber *et al.* (2009)	GenExP-LandSiTes	Y	free to use	http://engees.unistra.fr/~fleber/Landsites/	N	N
Papaïx *et al.* (2014)	*Ddal* Landscape simulator	N	not open to the public		N	N
Pe'er *et al.* (2013)	*G-RaFFe*	N	on request		N	Y
van Strien *et al.* (2016)	Landscape Generator (LG)	Y	free to use	www.lg.ethz.ch	N	Y

n.i. No information available.

3.8 Other tools for generating neutral landscapes

Our definition of ALGs is rather restrictive, as we require the potential to systematically vary landscape features in an automated way. Such generators have many advantages, but there are also software tools, or generators, that are not ALGs according to our definition but are still certainly useful for more specific purposes. Here, we give a brief overview of such tools that we found in our survey of the literature. They have in common that they do not explicitly address agricultural landscapes or land use types.

RULE is a software package for the generation of neutral landscape models and the analysis of landscape patterns (Gardner *et al.*, 1987, Gardner, 1999, Gardner & Walters, 2002). Landscape patterns are generated either as simple random processes (random maps), or as a result of spatially correlated processes using algorithms derived from fractal geometry (multifractal maps). *QRULE* is a further development of *RULE*, retaining the essential features but providing statistical summaries based on area rather than pixel counts, improving the formats of ancillary data sets, and adding the potential for developing and analysing alternative neutral models (Gardner & Urban, 2007).

The *Fractal Realizer* (FR), developed by Hargrove *et al.* (2002), generates multiple-category synthetic landscape maps according to user specifications. The synthetic landscapes show statistical properties similar to those of a particular empirical landscape and can be used to generate replicated input to spatial simulation models. It generates fractal landscape patterns based on the midpoint displacement algorithm by Saupe (1988).

GradientLand (Cambui *et al.*, 2014) is a free software program for generating a wide range of habitat cover gradients as random and fractal neutral landscapes. In the fractal mode, varying the aggregation and land cover is realised by using the midpoint displacement algorithm. Completely random patterns are produced by using a uniform probability distribution.

NLMpy (Etherington *et al.*, 2015) is a Python software package for the creation of neutral landscapes within a general numerical framework. It integrates a range of NLM algorithms that differ in the spatial autocorrelation of the element values in a two-dimensional array. It is open source, can be used on any computer system and is easily combined with geographic information system (GIS) data.

NLMR and *landscapetools* (Sciaini *et al.*, 2018) are R packages for simulating and modifying neutral landscape models in a single environment. *NLMR* is a comprehensive collection of algorithms for creating neutral landscapes and *landscapetools* provides a utility toolbox which facilitates an easy workflow with neutral landscapes and other raster data.

Hiebeler (2000) describes a simple algorithm for generating landscapes with spatially structured habitat heterogeneities. The landscapes consist of rectangular lattices of sites, or patches, each characterised by a value indicating its habitat type. Hiebeler (2007) improved the landscape-generation algorithm by using stratified sampling of sites rather than simple random sampling.

Remmel & Fortin (2013) utilize a stationary random field simulator (Remmel & Csillag, 2003) to produce large numbers of binary landscapes with identical parameters. Clearly defined descriptors of spatial pattern (composition and configuration) can be parameterized within the R statistical computing environment.

4 Outline of a future generic ALG

Our Review shows that, currently, no generic, commonly used or useable ALG exists, but isolated solutions have been developed in specific contexts with specific purposes. It seems impossible to design a single unifying ALG that is able to cover all past and future applications and questions. A generic ALG would have to perform the balancing act of being both versatile and adaptable for case specific application. In order to solve this dilemma, in the following we would like to suggest a modular design for generic ALGs.

Specific requirements

Our idea behind generic ALGs is the development of a software tool in modular design. The activation and deactivation of modules should be controllable with regard to the final result, because the design of the landscapes can vary considerably depending on the final use. Likewise, defined interfaces should exist for adding own, specific modules. The design depends above all on the simulation models to which the landscapes produced by the ALGs serve as an input. These can be different types of dynamic models, for example, LUCC models, spatial ecological models that analyse the effects of landscapes on certain ecological aspects (e.g. animal species), or agent-based land use models, which analyse farmer‘s decisions under the influence of institutional (markets, policies) and natural (e.g. soil type) framework conditions.

Basically, the generic ALGs create and systematically vary hypothetical agricultural landscapes (spatial configuration and composition of field mosaics). The following requirements should be fulfilled: (1) Pattern- and/or process-based generation. (2) The generation of a large number of random landscape maps in order to make general statements as well as the generation of specific landscapes. (3) Random and targeted distribution of landscape elements. Table 4 presents a tentative list of features that such a generic ALG should have. This list is our subjective merger of features that we, based on our review, consider essential. All features should be separately testable and controllable. Essential elements are required to adapt the landscapes to a wide range of target model types. Optional elements allow tailoring the landscape to specific purposes. The size and number of fields and the assignment of crop types are obligatory inputs. Regarding crop types, coupling with a software program that generates crop types or rotations (Dogliotti *et al.*, 2003, Bachinger & Zander, 2007, Schönhart *et al.*, 2009, Castellazzi *et al.*, 2010) can be useful. Additional landscape elements, e.g., semi-natural landscape elements and hedges, are optional, but can be decisive, as in the case of pollinators. Even the addition of abiotic (e.g., soil, runoff, relief) and socio-economic factors should be possible.

Table 4. Essential and optional features of agricultural landscape generated with a generic ALG.

Essential features	Optional features
Field size	Natural and semi-natural landscape elements
Number of fields	Fringe structures, e.g., hedges, field margins
Shape of fields	Other land use types, e.g., set-aside, grassland
Crop types	Farm type
Spatial correlation (clustering, fragmentation, distribution)	Crop rotation schemes
	Abiotic factors, e.g., soil, runoff, relief

A gradient from very simple to very complex landscapes should be possible. Landscape complexity can successively be increased by adding certain features or modules. The resulting landscape must have an adequate degree of complexity, which is not just a technical or methodological question but a practice-oriented one (Seppelt *et al.*, 2009). Because scale, grid type, resolution and the degree of complexity have to be compatible with the simulation model to which the landscapes serve as input. This also has an effect on the design of the landscape, since some landscape elements are not visible at a particular resolution, e.g., fine fringe structures.

Model and software architecture

The optimal generic ALG framework is open source, fully documented and easy to use. With regard to an integrative approach, other open source models can be integrated quickly and easily. According to Agarwal *et al.* (2002), initiating such an open-source modelling effort will require several components: (1) a web site to support modelling collaboration (e.g., data and interactions among individuals, such as bulletin boards and FAQs); (2) the establishment of one or more modelling “kernels” (core components of models using various technologies) that are designed in a modular fashion and allow participants to make enhancements with relative ease; and (3) the development of mechanisms for sharing model enhancements that encourage participation and provide incentives that are comparable. Experience with the generic forest succession model Landis II (Scheller *et al.*, 2007) shows that it is crucial to employ modern software engineering techniques for software that is intended to be generic and used by many people (Scheller *et al.*, 2010).

A modular model architecture meets the requirements for generic ALGs. It enables the integration of building blocks of existing ALGs that have proven to be useful, in the form of modules. All modules can be easily revised, such that the latest expert knowledge can always be integrated. Furthermore, new modules can be added with little effort. The ALGs should be operated via an intuitive graphical user interface that consists of different toolboxes with open source code for model development. Ideally, the package concept of R would be adopted, but this would require the existence of a core architecture and scheduler. The algorithms and rationale of the ALGs have to be fully documented and a user manual and guided tour provided. The ALGs should not only be a stand-alone program but should also be usable as a library, i.e., it can be started from every simulation model and generate landscapes during the simulation process on the fly. Such coupling is important particularly for optimisation models. An interface to GIS should exist to be able to alter real imported maps automatically using the ALGs. Various software platforms or programming languages could be envisaged for developing generic ALGs; however, no single language will be accepted, and used, by all potential users. Still, if a certain widely used language is chosen, such as C++, Java or Python, programming interfaces should be provided that allow links to other languages, if possible. An alternative is to try to build on existing GIS software. One option is to use the ArcGIS ModelBuilder, but this might be suboptimal because this software is proprietary and still might not offer the full flexibility required.

A suitable solution might be based on a domain-specific visual dataflow programming approach. Visual dataflow programming (VDP) is a programming paradigm that represents a program as a directed graph of data flows between operations (Wikipedia, 2018). In visual dataflow programming, users can assemble a program by placing operations (nodes) and connecting their data inputs and outputs in a graphical user interface (Fig. 12).

The appeal of VDP is that it combines high flexibility and extensibility with ease of use and program readability without programming knowledge. In the context of an ALG framework, operations (i.e., nodes) would include basic GIS operations such as a raster calculator, buffer, filters or Voronoi diagram calculation, as well as generators such as Perlin noise or random points. Finally, it would include output operations for different file formats and visualisation for debugging. Every operation can have input and output slots for defined data types (e.g., floating-point number, integer raster, vector layer).

The framework would be easily extensible through the addition of operations. Although some programming knowledge in the framework's implementation language would be necessary for the creation of new operations, their mutual independence would greatly benefit flexibility and facilitate the exchange of operations between users or a central repository for official and community-based extensions (operations or bundles thereof).

One important advantage of this approach would be that it combines ease of use, without having to learn a specific programming language, with modularity and extensibility based on a widely used language. In the Supplementary Material, an example VDP program implemented in a prototype framework in Scala is explained in detail to further illustrate the concept of VDP.

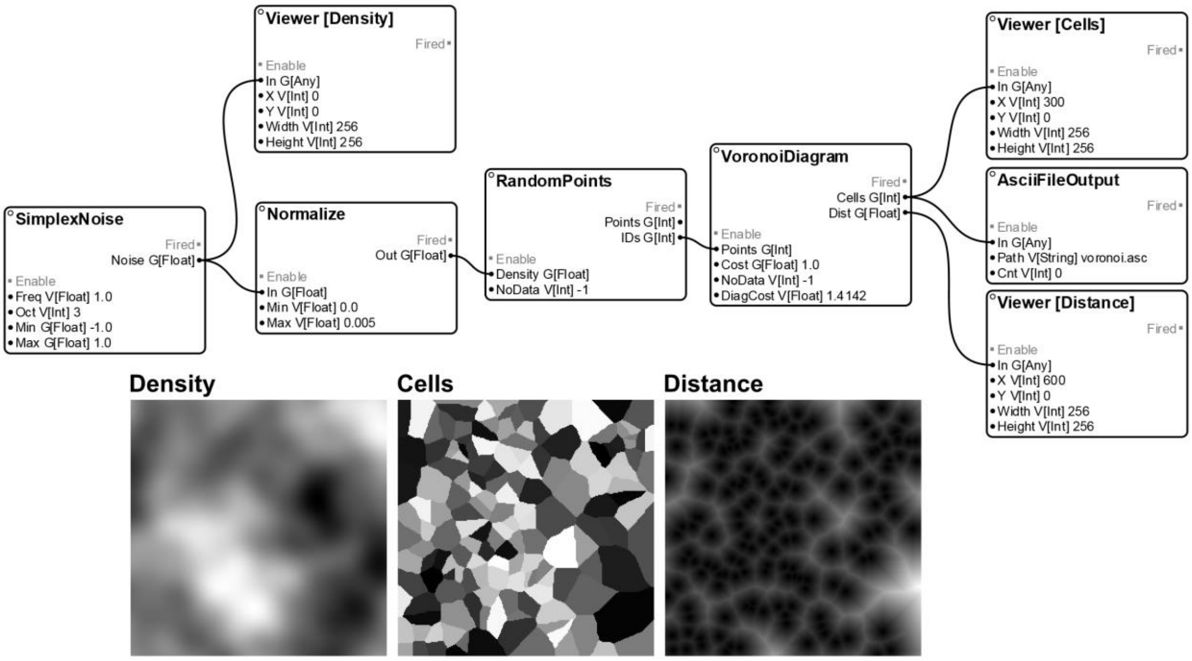

Figure 12. Example of Visual Dataflow Programming for generating a landscape based on Voronoi tessellation. Users compile programs by selecting and combining nodes that represent certain operations. Using Java, developers program further nodes. (See also Supplementary material.)

Another generic approach to implementing modular generic ALGs are the so-called "pattern grammars" (Gaucherel *et al.*, 2012), where landscape features are described using a certain syntax. This approach has been widely used for functional-structural plant modelling (FSPM), using L-systems grammar (e.g., Godin & Sinoquet, 2005, Wang *et al.*, 2018). The ALG DYPAL (Gaucherel *et al.*, 2006) is based on such a grammar and is one of the most flexible existing ALGs.

5 Discussion

The computer-aided generation of agricultural landscapes can provide a framework for understanding and evaluating how land use and land cover changes will take place and affect the environment. Being input to simulation models, the landscapes can serve as a basis for decision-making for policy makers or as support for discussion and negotiation. This especially concerns insights into the adequate scale of policies (national or more regionalised), but also into the relative importance of landscape structural elements for achieving societal goals with implications for the priorisation of conservation activities. The results are useful for communicating policies and the corresponding landscape changes and environmental effects to local actors and to the public.

Computer-generated landscapes offer a number of advantages: (1) they can fill the gap when real data are not available or are not at a suitable resolution, and they allow us (2) to test new landscape configurations, (3) to carry out systematic analyses of environmental gradients, and (4) to perform spatial sensitivity analyses as a basis for the regional transferability of the results. For these reasons, we believe that the future lies in the computational evolution of landscape patterns not limited to random change.

The aim of each ALG should be to design a plausible landscape with appropriate input parameters and algorithms. This means that the generated landscape should not be more complex than necessary to serve as input for a simulation model or to answer the scientific or policy question. Still, most existing ALGs represent rather simple agricultural landscapes via a few regularly shaped patches. Configurational changes are often realised by transition probabilities. Important questions, though, exist that would require more-complex landscapes by, for example, adding semi-natural habitats or applying advanced design rules which acknowledge the relationship between abiotic site conditions (e.g., soil, runoff, relief) and crop choice. In this way, all processes that lead to significant changes in configuration and use can be included.

Process-based approaches

The landscapes of process-based ALGs are the result of the implemented process and are therefore strongly dependent on the input parameters. Users thus need to know and fully understand the effect of each input parameter to design specific landscapes. An advantage of process-based or mechanistic ALGs is that the underlying processes can be explicitly adressed, so these landscapes are more likely to be relevant for the study of policy measures. The disadvantage is that it can be difficult to quantify all the necessary parameters.

We found only two process-based ALGs developed so far. *DYPAL* is the most complex and most advanced one. It integrates numerous algorithms as a basis for various processes. While neutral models are already implemented for the composition, landscape configuration is currently based on real landscapes. That's the reason why this approach does not fully meet our selection criteria and cannot be compared very well with the other approaches. Nevertheless, we have included it because it shows the possibilities of a process-based generator. The possibility to generate landscape configurations in a systematic and automated way would make the application of *DYPAL* much more versatile and flexible. *G-RaFFe* simulates the highly significant process by which primary forest is transformed into arable land. The conversion is performed by simulated farm establishment subsequent to the generation of roads. Unfortunately, agricultural fields cannot be assigned to different crops, so this generator can only be used for the investigation of landscape dynamics rather than for agricultural questions.

Pattern-based approaches

Pattern-based ALGs usually require fewer parameters and less computing time than process-based ones. There are a variety of landscape metrics that can be used as input parameters, e.g., the number and size of fields or the proportion of landscape elements. Most ALGs integrate only a few landscape metrics, which limits the design of the landscape pattern and, thus, the range of possible applications. To generate realistic agricultural landscapes, a set of different landscape metrics is needed. However, one should consider that more input parameters as well as a larger grid size lead to more-complicated algorithms and longer computing time. To solve this problem, efficient algorithms become important.

The strength of the AgBioScape landscape mosaic generator is that it is already coupled with four population modules (a ‘conservation’ species, an agricultural pest, and two functionally distinct natural enemies) to investigate the influence of management and land use patterns. The generator can be used to generate simple field patterns from rectangular fields with fringe structures and to assign four different crop types to the fields. The landscape generator of Engel *et al.* (2012), Everaars *et al.* (2014) also designs simple field patterns, which can be assigned to 15 different crop types. The focus is on the variation of crop proportions and the mean field size. However, it cannot vary the range of field sizes, leading to quite artificial landscapes composed of square fields. Using the ALG developed by van Strien *et al.* (2016), a differentiated landscape can be generated by integrating different landscape metrics that can be applied at either the field or class level. The number of different land use classes can be defined, but no specific crop types have been implemented so far.

With the tessellation methods applied in *GenExP-LandSiTes, SG4GISCAME* and the landscape simulator by Papaïx *et al.* (2014), irregular geometric field patterns are generated without fringe structures. Tessellations have the advantage that the general geometrical character as well as the spatial distribution of the agricultural landscape can be preserved. The difficulty is in controlling important landscape features, such as field sizes and shapes or distances between tessellation seeds. While the tessellation methods allow one to simulate numbers of fields and average field sizes that are similar to those of real landscapes, they do not correctly depict the shapes of the fields or the variability of these shapes within the landscape (Le Ber *et al.*, 2009). *GenExP-LandSiTes* produces slightly too-compact fields with little variability, while the rectangular tessellation produces over-elongated fields with too-high variability. As a result, this ALG generates landscapes with configurations of limited realism, and landscape dynamics cannot be investigated very well. In contrast, *SG4GISCAME* provides a better representation of field shapes through higher flexibility and refinement algorithms.

Simulating crops

In most existing ALGs, crop types are defined only marginally (Le Ber *et al.*, 2009, Papaïx *et al.*, 2014, Begg & Dye, 2015, van Strien *et al.*, 2016, Dislich *et al.*, 2018) or not at all (Pe'er *et al.*, 2013). This means that although there are various land use patterns from agricultural fields, there are few specific farming systems. However, representing land cover in more detail is essential to answer many questions regarding agricultural landscapes. The tessellation method, for example, offers little possibility for handling attributes of landscape composition, but it can be combined with crop rotation models. Statistical modelling of crop rotations in general and Markov models in particular are suitable tools for predicting crop types. However, Markov models per se are empirical, based on observed transitions of land use types, or are fully hypothetical. Therefore, they should be constrained by conditional rules based on expert knowledge.

Evaluating policy measures

By using computer-generated landscapes in simulation models, agricultural policies can be evaluated quickly and effectively at the local or regional scale. This is an advantage over field observations and long-term field experiments, which are more time-consuming but obtain higher precision and validity. Both methods have their pros and cons and should be used and supplemented according to their capabilities. The rapid evaluation and prediction of environmental effects related to policies are a great strength of simulation models and are essential to propose and implement sustainable and efficient environmental policies. However, since the so-far generated landscapes are often rather simple, the application as a tool for evaluating policy measures is still limited because the results are only partially transferable to reality. Validation not only of the simulation models used but also of the ALG is crucial in this context. Here, validation means providing evidence that ALG-generated landscapes are able to capture essential features of existing, and planned, landscapes.

Challenges & limitations

We have tried to describe existing ALG approaches in the same depth and with regard to the same criteria. Unfortunately, not only are the approaches themselves very different in terms of methodology, but they are also often incomplete in their description, which makes them difficult to compare. The approaches are not des cribed in the same way, terminologies differ, and much of the information we have been looking for is not available. This leads to incompleteness in the description of some approaches and, possibly, to biases. ALG descriptions following a standardised protocol would be extremely helpful and a desirable standard for publications in the future. For example, the ODD protocol, which is a standard format for describing individual- or agent-based models (Grimm *et al.*, 2006, Grimm *et al.*, 2010), could be adopted for this purpose.

The ALGs presented here have mostly only been used within the research group that developed them. Thus, they were not designed for general application and availability, nor were they designed with user friendliness in mind. Generic ALGs will require well-designed software and detailed documentation so that the user can quickly become familiar with the ALG's functionality and understand exactly which application possibilities are available. In our opinion, the generator developed by van Strien *et al.* (2016) comes closest to meeting these demands. A well-documented open source code can be further developed and enriched by the scientific community, and it can be used for other models. Until now, most ALGs have been developed in isolation. Because no consistent terminology regarding landscape generators exists, and because we had to focus on ISI-listed publications, we might have overlooked further relevant ALGs, although we could not find references to any further ALGs in the publications we reviewed.

Our review shows that the development of computer-generated agricultural landscapes is still in its infancy. Most generators generate simple landscapes, which limits the range of application. In most cases, the main focus is on either the spatial configuration or the composition, not on both in a balanced way. All approaches still have great potential for further development towards broader applicability. To address the multi-functionality of landscapes, generic ALGs should include other land-use types such as forests or settlements. Pattern-oriented modelling can be used, for example, to identify relevant landscape metrics. The alignment of computer-generated landscapes with real landscapes still has much potential for improvement. This can be made possible by the further development of specific algorithms and calibration or validation with high-resolution data on cultivation systems and abiotic factors. An important data input for future ALGs will be remote sensing data (e.g., Pettorelli *et al.*, 2016), which open up new approaches for identifying land use types (e.g., Joshi *et al.*, 2016).

Outlook

As a result of our review, we derived requirements for generic future ALGs that have the potential to be widely used in landscape science. Environmental questions can be examined on certain spatial scales and transferred to other scales that are not feasible to explore in reality. However, the difficulty of computer-generated landscapes is that they represent few landscape features well simultaneously. Each extension leads to a more complicated algorithm and longer calculation time. This leads to the classical modeller's dilemma of when functionality falls victim to complexity. Therefore, the complexity of the model has to be carefully considered as well as how versatile the ALG can and should be. It is important to give the user the possibility of prioritising, e.g., which landscape features should be exact and which can be approximate. Generic ALGs thus needs to have a modular and preferably open architecture. In the Supplement, we outline one possible solution, but other solutions certainly exist. In any case, although the architecture of generic ALGs might be set up by individual groups of researcher, e.g. the approach that we outline above and in the supplement, or approaches based on pattern grammar (Gaucherel *et al.*, 2006, Gaucherel *et al.*, 2012), establishing a modular framework would have to be a community activity. Ideally, national or European funding agencies would initiate such joint projects, similar to the EU initiative to standardize models of the fate of pesticides in surface waters (FOCUS, 2001).

6 Conclusion

Systematically generating variable virtual landscapes as a basis for spatially explicit assessment opens up new ways of exploring environmental issues. There is an enormous field of application, particularly with regard to the evaluation of policy measures. Critical threshold values of environmental effects can be identified at an early stage, and suitable mitigation measures can be developed. Due to the enormous application potential, the significance of landscape generators as a basis for spatial simulation models will continue to grow in the future, and they will be a useful complement to real landscapes and long-term field studies. However, considerable development work is still needed to improve and generalise the generators, which would be feasible in a joint project in the open source context. We expect that 5-10 years are needed to develop, successively, such a generic ALG framework and that considerable manpower and data sources are needed. In Europe, this would ideally be the scope of EU-funded projects. Hot topics such as the decline of pollinators such as honeybees (Potts *et al.*, 2010, Horn, 2017), evaluating and improving the effectiveness of the CAP reform (Pe'er *et al.*, 2014), or the safe provisioning of the full bundle of ecosystem services (Raudsepp-Hearne *et al.*, 2010) emphasise that investing in developing a generic, modular ALG would not only pay off well but might actually by crucial.

7 Acknowledgements

M.L. acknowledges funding of this project from the Heinrich Böll Foundation, Germany.

8 References

Agarwal, C., Green, G. M., Grove, J. M., Evans, T., P. and Schweik, C. M. (2002): A Review and Assessment of Land-Use Change Models: Dynamics of Space, Time, and Human Choice. *Gen. Tech. Rep. NE-297*. Department of Agriculture, Forest Service, Northeastern Research Station, Newton Square, PA: U.S.

Bachinger, J. and Zander, P. (2007): ROTOR, a tool for generating and evaluating crop rotations for organic farming systems. *European Journal of Agronomy* 26(2): 130-143.

Becher, M. A., Grimm, V., Thorbek, P., Horn, J., Kennedy, P. J. and Osborne, J. L. (2014): BEEHAVE: a systems model of honeybee colony dynamics and foraging to explore multifactorial causes of colony failure. *The Journal of Applied Ecology* 51(2): 470-482.

Becher, M. A., Twiston-Davies, G., Penny, T. D., Goulson, D., Rotheray, E. L. and Osborne, J. L. (2018): Bumble-BEEHAVE: A systems model for exploring multifactorial causes of bumblebee decline at individual, colony, population and community level. *Journal of Applied Ecology* 0(0).

Begg, G. S. and Dye, R. (2015): Exploring trade-offs and synergies between biological pest control and species conservation. *Scottish Natural Heritage Commissioned Report No. 692*.

Bonhomme, V., Castets, M., Ibanez, T., Géraux, H., Hély, C. and Gaucherel, C. (2017): Configurational changes of patchy landscapes dynamics. *Ecological Modelling* 363(Supplement C): 1-7.

Caldiera, G. and Presutti, E. (1974): Gibbs processes and generalized Bernoulli flows for hard-core one-dimensional systems. *Communications in Mathematical Physics* 35(4): 279-286.

Cambui, E., Nogueira de Vasconcelos, R., Boscolo, D., Rocha, P. and Miranda, J. G. (2014): GradientLand Software: A landscape change gradient generator. *Ecological Informatics* 25: 57-62.

Castellazzi, M. S., Matthews, J., Angevin, F., Sausse, C., Wood, G. A., Burgess, P. J., Brown, I., Conrad, K. F. and Perry, J. N. (2010): Simulation scenarios of spatio-temporal arrangement of crops at the landscape scale. *Environmental Modelling & Software* 25(12): 1881-1889.

Castellazzi, M. S., Wood, G. A., Burgess, P. J., Morris, J., Conrad, K. F. and Perry, J. N. (2008): A systematic representation of crop rotations. *Agricultural Systems* 97(1-2): 26-33.

Colbach, N., Monod, H. and Lavigne, C. (2009): A simulation study of the medium-term effects of field patterns on cross-pollination rates in oilseed rape (Brassica napus L.). *Ecological Modelling* 220(5): 662-672.

Cuddington, K., Fortin, M. J., Gerber, L. R., Hastings, A., Liebhold, A., O'Connor, M. and Ray, C. (2013): Process-based models are required to manage ecological systems in a changing world. *Ecosphere* 4(2): 1-12.

Dislich, C., Hettig, E., Salecker, J., Heinonen, J., Lay, J., Meyer, K. M., Wiegand, K. and Tarigan, S. (2018): Land-use change in oil palm dominated tropical landscapes—An agent-based model to explore ecological and socio-economic trade-offs. *PLOS ONE* 13(1): e0190506.

Dogliotti, S., Rossing, W. A. H. and van Ittersum, M. K. (2003): ROTAT, a tool for systematically generating crop rotations. *European Journal of Agronomy* 19(2): 239-250.

Engel, J., Huth, A. and Frank, K. (2012): Bioenergy production and Skylark (Alauda arvensis) population abundance – a modelling approach for the analysis of land-use change impacts and conservation options. *GCB Bioenergy* 4(6): 713-727.

Etherington, T. R., Holland, E. P. and O'Sullivan, D. (2015): NLMpy: a python software package for the creation of neutral landscape models within a general numerical framework. *Methods in Ecology and Evolution* 6(2): 164-168.

Everaars, J., Frank, K. and Huth, A. (2014): Species ecology and the impacts of bioenergy crops: an assessment approach with four example farmland bird species. *GCB Bioenergy* 6(3): 252-264.

Everaars, J., Settele, J. and Dormann, C. F. (2018): Fragmentation of nest and foraging habitat affects time budgets of solitary bees, their fitness and pollination services, depending on traits: Results from an individual-based model. *PLOS ONE* 13(2): e0188269.

Filho, B., Cerqueira, G. and Lopes Pennachin, C. (2002): DINAMICA - A stochastic cellular automata model designed to simulate the landscape dynamics in an Amazonian colonization frontier. *Ecological Modelling* 154(3): 217-235.

FOCUS (2001): FOCUS Surface Water Scenarios in the EU Evaluation Process under 91/414/EEC. *Report of the FOCUS Working Group on Surface Water Scenarios*.

Forman, R. T. T. and Godron, M. (1981): Patches and Structural Components for a Landscape Ecology. *Bioscience* 31(10): 733-740.

Gardner, R. and Walters, S. (2002). Neutral Landscape Models. *Learning Landscape Ecology: A Practical Guide to Concepts and Techniques*. Sarah Gergel and Monica Turner. New York, Springer-Verlag**:** 112-128.

Gardner, R. H. (1999). RULE: A program for the generation of random maps and the analysis of spatial patterns. *Landscape Ecological Analysis: Issues and Applications*. J. M. Klopatek and R. H. Gardner, Springer-Verlag**:** 280-303.

Gardner, R. H., Milne, B. T., Turner, M. G. and O'Neill, R. V. (1987): Neutral models for the analysis of broad-scale landscape pattern. *Landscape Ecology* 1(1): 19-28.

Gardner, R. H. and Urban, D. L. (2007): Neutral models for testing landscape hypotheses. *Landscape Ecology* 22(1): 15-29.

Gaucherel, C. (2008): Neutral models for polygonal landscapes with linear networks. *Ecological Modelling* 219(1-2): 39-48.

Gaucherel, C., Alet, J. and Garcia, C. (2017): Coffee monoculture trends in tropical agroforested landscapes of Western Ghats (India). *Environmental Conservation* 44(2): 183-190.

Gaucherel, C., Boudon, F., Houet, T., Castets, M. and Godin, C. (2012): Understanding Patchy Landscape Dynamics: Towards a Landscape Language. *PLoS ONE* 7(9): e46064.

Gaucherel, C., Fleury, D., Auclair, D. and Dreyfus, P. (2006): Neutral models for patchy landscapes. *Ecological Modelling* 197(1–2): 159-170.

Gaucherel, C., Giboire, N., Viaud, V., Houet, T., Baudry, J. and Burel, F. (2006): A domain-specific language for patchy landscape modelling: The Brittany agricultural mosaic as a case study. *Ecological Modelling* 194(1–3): 233-243.

Gaucherel, C., Griffon, S., Misson, L. and Houet, T. (2010): Combining process-based models for future biomass assessment at landscape scale. *Landscape Ecology* 25(2): 201-215.

Gaucherel, C., Houllier, F., Auclair, D. and Houet, T. (2014): Dynamic Landscape Modelling : The Quest for a Unifying Theory. *Living Reviews in Landscape Research* 8(2): 5-31.

Godin, C. and Sinoquet, H. (2005): Functional–structural plant modelling. *New Phytologist* 166(3): 705-708.

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., Huth, A., Jepsen, J. U., Jørgensen, C., Mooij, W. M., Müller, B., Pe'er, G., Piou, C., Railsback, S. F., Robbins, A. M., Robbins, M. M., Rossmanith, E., Rüger, N., Strand, E., Souissi, S., Stillman, R. A., Vabø, R., Visser, U. and DeAngelis, D. L. (2006): A standard protocol for describing individual-based and agent-based models. *Ecological Modelling* 198(1–2): 115-126.

Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J. and Railsback, S. F. (2010): The ODD protocol: A review and first update. *Ecological Modelling* 221(23): 2760-2768.

Hargrove, W. W., Hoffman, F. M. and Schwartz, P. M. (2002): A fractal landscape realizer for generating synthetic maps. *Conservation Ecology* 6(1).

Heistermann, M., Müller, C. and Ronneberger, K. (2006): Land in sight?: Achievements, deficits and potentials of continental to global scale land-use modeling. *Agriculture, Ecosystems & Environment* 114(2–4): 141-158.

Hiebeler, D. (2000): Populations on Fragmented Landscapes with Spatially Structured Heterogeneities: Landscape Generation and Local Dispersal. *Ecology* 81(6): 1629-1641.

Hiebeler, D. E. (2007): Competing populations on fragmented landscapes with spatially structured heterogeneities: improved landscape generation and mixed dispersal strategies. *Journal of Mathematical Biology* 54(3): 337-356.

Horn, H. S., Cody, M. L. and Diamond, J. M. (1975). Markovian properties of forest succession. *Ecology and evolution of communities*. M.L. Cody and J.M. Diamond. Cambridge, MA, Harvard University Press**:** 196-211.

Horn, J. (2017). A modelling framework for exploration of a multidimensional factory causing decline in honeybee health. Doctoral thesis, Department of Plant Ecology and Nature Conservation, University of Potsdam.

Houet, T., Loveland, T. R., Hubert-Moy, L., Gaucherel, C., Napton, D., Barnes, C. A. and Sayler, K. (2010): Exploring subtle land use and land cover changes: a framework for future landscape studies. *Landscape Ecology* 25(2): 249-266.

Houet, T., Schaller, N., Castets, M. and Gaucherel, C. (2014): Improving the simulation of fine-resolution landscape changes by coupling top-down and bottom-up land use and cover changes rules. *International Journal of Geographical Information Science* 28(9): 1848-1876.

Inkoom, J. N., Frank, S. and Fürst, C. (2017): Challenges and opportunities of ecosystem service integration into land use planning in West Africa – an implementation framework. *International Journal of Biodiversity Science, Ecosystem Services & Management* 13(2): 67-81.

Inkoom, J. N., Frank, S., Greve, K. and Fürst, C. (2017): Designing neutral landscapes for data scarce regions in West Africa. *Ecological Informatics* 42(Supplement C): 1-13.

Jackson, L. J., Trebitz, A. S. and Cottingham, K. L. (2000): An Introduction to the Practice of Ecological Modeling. *BioScience* 50(8): 694-706.

Johnson, G. D., Myers, W. L. and Patil, G. P. (1999): Stochastic generating models for simulating hierarchically structured multi-cover landscapes. *Landscape Ecology* 14(5): 413-421.

Johst, K., Drechsler, M., Mewes, M., Sturm, A. and Wätzold, F. (2015): A novel modeling approach to evaluate the ecological effects of timing and location of grassland conservation measures. *Biological Conservation* 182: 44-52.

Joshi, N., Baumann, M., Ehammer, A., Fensholt, R., Grogan, K., Hostert, P., Jepsen, M., Kuemmerle, T., Meyfroidt, P., Mitchard, E., Reiche, J., Ryan, C. and Waske, B. (2016): A Review of the Application of Optical and Radar Remote Sensing Data Fusion to Land Use Mapping and Monitoring. *Remote Sensing* 8(1): 70.

Kiêu, K., Adamczyk-Chauvat, K., Monod, H. and Stoica, R. S. (2013): A completely random T-tessellation model and Gibbsian extensions. *Spatial Statistics* 6(Supplement C): 118-138.

Lambin, E. F., Rounsevell, M. D. A. and Geist, H. J. (2000): Are agricultural land-use models able to predict changes in land-use intensity? *Agriculture, Ecosystems & Environment* 82(1–3): 321-331.

Langhammer, M., Grimm, V., Pütz, S. and Topping, C. J. (2017): A modelling approach to evaluating the effectiveness of Ecological Focus Areas: The case of the European brown hare. *Land Use Policy* 61: 63-79.

Lavigne, C., Klein, E. K., Mari, J.-F., Ber, F. L., Adamczyk, K., Monod, H. and Angevin, F. (2008): How do genetically modified (GM) crops contribute to background levels of GM pollen in an agricultural landscape? *Journal of Applied Ecology* 45(4): 1104-1113.

Le Ber, F., Benoît, M., Schott, C., Mari, J. F. and Mignolet, C. (2006): Studying crop sequences with CarrotAge, a HMM-based data mining software. *Ecological Modelling* 191(1): 170-185.

Le Ber, F., Lavigne, C., Adamczyk, K., Angevin, F., Colbach, N., Mari, J. F. and Monod, H. (2009): Neutral modelling of agricultural landscapes by tessellation methods—Application for gene flow simulation. *Ecological Modelling* 220(24): 3536-3545.

Le Ber, F. and Mari, J.-F. (2013). GENEXP-LANDSITES: A 2D Agricultural Landscape Generating Piece of Software. *Innovative Software Development in GIS*, John Wiley & Sons, Inc.**:** 189-214.

McGarigal, K., Cushman, S. A. and Ene, E. (2012): FRAGSTATS v4: Spatial Pattern Analysis Program for Categorical and Continuous Maps. Computer software program produced by the authors at the University of Massachusetts, Amherst. Available at the following web site: http://www.umass.edu/landeco/research/fragstats/fragstats.html.

Mosconi, M. and Porta, M. (2000): Iteration constructs in data-flow visual programming languages. *Computer Languages* 26(2): 67-104.

Papaïx, J., Adamczyk-Chauvat, K., Bouvier, A., Kiêu, K., Touzeau, S., Lannou, C. and Monod, H. (2014): Pathogen population dynamics in agricultural landscapes: The Ddal modelling framework. *Infection, Genetics and Evolution* 27: 509-520.

Parker, D. C., Manson, S. M., Janssen, M. A., Hoffmann, M. J. and Deadman, P. (2003): Multi-Agent Systems for the Simulation of Land-Use and Land-Cover Change: A Review. *Annals of the Association of American Geographers* 93(2): 314-337.

Pe'er, G., Dicks, L. V., Visconti, P., Arlettaz, R., Báldi, A., Benton, T. G., Collins, S., Dieterich, M., Gregory, R. D., Hartig, F., Henle, K., Hobson, P. R., Kleijn, D., Neumann, R. K., Robijns, T., Schmidt, J., Shwartz, A., Sutherland, W. J., Turbé, A., Wulf, F. and Scott, A. V. (2014): EU agricultural reform fails on biodiversity. *Science* 344(6188): 1090-1092.

Pe'er, G., Henle, K., Dislich, C. and Frank, K. (2011): Breaking Functional Connectivity into Components: A Novel Approach Using an Individual-Based Model, and First Outcomes. *PLOS ONE* 6(8): e22355.

Pe'er, G., Zurita, G. A., Schober, L., Bellocq, M. I., Strer, M., Müller, M. and Pütz, S. (2013): Simple Process-Based Simulators for Generating Spatial Patterns of Habitat Loss and Fragmentation: A Review and Introduction to the G-RaFFe Model. *PLoS ONE* 8(5): e64968.

Pettorelli, N., Wegmann, M., Skidmore, A., Mücher, S., Dawson, T. P., Fernandez, M., Lucas, R., Schaepman, M. E., Wang, T., O'Connor, B., Jongman, R. H. G., Kempeneers, P., Sonnenschein, R., Leidner, A. K., Böhm, M., He, K. S., Nagendra, H., Dubois, G., Fatoyinbo, T., Hansen, M. C., Paganini, M., Klerk, H. M., Asner, G. P., Kerr, J. T., Estes, A. B., Schmeller, D. S., Heiden, U., Rocchini, D., Pereira, H. M., Turak, E., Fernandez, N., Lausch, A., Cho, M. A., Alcaraz-Segura, D., McGeoch, M. A., Turner, W., Mueller, A., St-Louis, V., Penner, J., Vihervaara, P., Belward, A., Reyers, B., Geller, G. N. and Boyd, D. (2016): Framing the concept of satellite remote sensing essential biodiversity variables: challenges and future directions. *Remote Sensing in Ecology and Conservation* 2(3): 122-131.

Potts, S. G., Biesmeijer, J. C., Kremen, C., Neumann, P., Schweiger, O. and Kunin, W. E. (2010): Global pollinator declines: trends, impacts and drivers. *Trends in Ecology & Evolution* 25(6): 345-353.

Raudsepp-Hearne, C., Peterson, G. D. and Bennett, E. M. (2010): Ecosystem service bundles for analyzing tradeoffs in diverse landscapes. *Proceedings of the National Academy of Sciences* 107(11): 5242-5247.

Remmel, T. K. and Csillag, F. (2003): When are two landscape pattern indices significantly different? *Journal of Geographical Systems* 5(4): 331-351.

Remmel, T. K. and Fortin, M.-J. (2013): Categorical, class-focused map patterns: characterization and comparison. *Landscape Ecology* 28(8): 1587-1599.

Saupe, D. (1988). Algorithms for random fractals. *The science of fractal images*. H. O. Peitgen and D. Saupe. New York, Springer: 71-136.

Saura, S. and Martínez-Millán, J. (2000): Landscape patterns simulation with a modified random clusters method. *Landscape Ecology* 15(7): 661-678.

Scheller, R. M., Domingo, J. B., Sturtevant, B. R., Williams, J. S., Rudy, A., Gustafson, E. J. and Mladenoff, D. J. (2007): Design, development, and application of LANDIS-II, a spatial landscape simulation model with flexible temporal and spatial resolution. *Ecological Modelling* 201(3): 409-419.

Scheller, R. M., Sturtevant, B. R., Gustafson, E. J., Ward, B. C. and Mladenoff, D. J. (2010): Increasing the reliability of ecological models using modern software engineering techniques. *Frontiers in Ecology and the Environment* 8(5): 253-260.

Schönhart, M., Schmid, E. and Schneider, U. A. (2009): CropRota – A Model to Generate Optimal Crop Rotations from Observed Land Use. *Working Papers 452009*. Institute for Sustainable Economic Development, Department of Economics and Social Sciences, University of Natural Resources and Life Sciences, Vienna.

Schönhart, M., Schmid, E. and Schneider, U. A. (2011): CropRota – A crop rotation model to support integrated land use assessments. *European Journal of Agronomy* 34(4): 263-277.

Schröder, B. and Seppelt, R. (2006): Analysis of pattern-process interactions based on landscape models - Overview, general concepts, and methodological issues. *Ecological Modelling* 199(4): 505-516.

Schulze, J., Gawel, E., Nolzen, H., Weise, H. and Frank, K. (2017): The expansion of short rotation forestry: characterization of determinants with an agent-based land use model. *GCB Bioenergy* 9(6): 1042-1056.
Sciaini, M., Fritsch, M., Scherer, C. and Simpkins, C. E. (2018): NLMR and landscapetools: An integrated environment for simulating and modifying neutral landscape models in R. *Methods in Ecology and Evolution* 9(11): 2240-2248.
Seppelt, R., Müller, F., Schröder, B. and Volk, M. (2009): Challenges of simulating complex environmental systems at the landscape scale: A controversial dialogue between two cups of espresso. *Ecological Modelling* 220(24): 3481-3489.
Slager, C. T. J. (2011). Landscape generator : method to generate plausible landscape configurations for participatory spatial plan-making. Doctoral thesis, Technische Universiteit Eindhoven.
Slager, C. T. J. and de Vries, B. (2013): Landscape generator: Method to generate landscape configurations for spatial plan-making. *Computers, Environment and Urban Systems* 39: 1-11.
Sturm, A., Drechsler, M., Johst, K., Mewes, M. and Wätzold, F. (2018): DSS-Ecopay – A decision support software for designing ecologically effective and cost-effective agri-environment schemes to conserve endangered grassland biodiversity. *Agricultural Systems* 161: 113-116.
Turner, M. G. (1989): Landscape Ecology: The Effect of Pattern on Process. *Annual Review of Ecology and Systematics* 20(1): 171-197.
Usher, M. B. (1992). Statistical models of succession. *Plant Succession: Theory and Prediction*. D. C. Glenn-Lewin, R. K. Peet and T. T. Veblen. London, Chapman & Hall**:** 215-248.
van Strien, M. J. and Grêt-Regamey, A. (2016): How is habitat connectivity affected by settlement and road network configurations? Results from simulating coupled habitat and human networks. *Ecological Modelling* 342(Supplement C): 186-198.
van Strien, M. J., Slager, C. T. J., de Vries, B. and Grêt-Regamey, A. (2016): An improved neutral landscape model for recreating real landscapes and generating landscape series for spatial ecological simulations. *Ecology and Evolution* 6(11): 3808-3821.
Wang, M., White, N., Grimm, V., Hofman, H., Doley, D., Thorp, G., Cribb, B., Wherritt, E., Han, L., Wilkie, J. and Hanan, J. (2018): Pattern-oriented modelling as a novel way to verify and validate functional-structural plant models: a demonstration with the annual growth module of avocado. *Ann Bot* 121(5): 941-959.
Wätzold, F., Drechsler, M., Johst, K., Mewes, M. and Sturm, A. (2016): A Novel, Spatiotemporally Explicit Ecological-economic Modeling Procedure for the Design of Cost-effective Agri-environment Schemes to Conserve Biodiversity. *American Journal of Agricultural Economics* 98(2): 489-512.
Wikipedia (2018). Dataflow programming. Available at https://en.wikipedia.org/wiki/Dataflow_programming. Last access: March 14.
With, K. A. (1997): The application of neutral landscape models in conservation biology. *Conservation Biology* 11(5): 1069-1080.
With, K. A. and King, A. W. (1997): The Use and Misuse of Neutral Landscape Models in Ecology. *Oikos* 79(2): 219-229.

9 Appendix

Table A1. List of publications in which the ALGs are described and / or applied.

Name of the ALG	Main publication	Additional publications
AgBioscape	Begg & Dye (2015)	
Landscape generator	Engel *et al.* (2012)	Everaars *et al.* (2014)
DYPAL	Gaucherel *et al.* (2006)	Gaucherel *et al.* (2006), Gaucherel *et al.* (2010), Houet *et al.* (2010), Gaucherel *et al.* (2012), Houet *et al.* (2014), Bonhomme *et al.* (2017), Gaucherel *et al.* (2017)
SG4GISCAME	Inkoom *et al.* (2017)	Inkoom *et al.* (2017)
GenExP-LandSiTes	Le Ber *et al.* (2009)	Lavigne *et al.* (2008), Colbach *et al.* (2009), Le Ber & Mari (2013)
Ddal Landscape simulator	Papaïx *et al.* (2014)	
G-RaFFe	Pe'er *et al.* (2013)	Pe'er *et al.* (2011) Dislich *et al.* (2018)
Landscape Generator (LG)	van Strien *et al.* (2016)	Slager (2011), Slager & de Vries (2013), van Strien & Grêt-Regamey (2016)

Agricultural landscape generators for simulation models: a review of existing solutions and an outline of future directions

Supplement 1: VDP Example

[a]Maria Langhammer*, [a,b]Jule Thober, [a]Martin Lange, [a,c,d]Karin Frank, [a,c,e]Volker Grimm

[a]Helmholtz Centre for Environmental Research – UFZ, Department of Ecological Modelling, Permoserstr. 15, 04318 Leipzig, Germany

[b]Helmholtz Centre for Environmental Research – UFZ, Department of Computational Landscape Ecology, Permoserstr. 15, 04318 Leipzig, Germany

[c]German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Deutscher Platz 5e, 04103 Leipzig, Germany

[d]University of Osnabrück, Institute for Environmental Systems Research, Barbarastr. 12, 49076 Osnabrück, Germany

[e]University of Potsdam, Institute of Biochemistry and Biology, Department of Plant Ecology and Nature Conservation, Am Mühlenberg 3, 14476 Golm (Potsdam), Germany

*Corresponding author

Keywords: agricultural landscape, field pattern, agricultural landscape generator, landscape simulator, neutral landscape model, process-based model

To illustrate the concept of visual dataflow programming (VDP) within the context of agricultural landscape generators (ALGs), the simple example program shown in Fig. 12 is explained in detail. The figure shows a screenshot from a prototype VDP framework implemented by the authors in Scala. The purpose of the example program is to generate a simple patchy "landscape" with a spatially heterogeneous patch size ("Cells" in Fig. 12).

Operations/Nodes

Each box in Fig. 12 represents an operation, afterwards called a node. Nodes can be seen as functions taking parameters (called inputs) and returning results (called outputs). Taking the left-most node *SimplexNoise* as an example, the inputs are the black items on the left in the node, while the outputs are the black items on the right.

SimplexNoise has four inputs (*Freq*[ency], *Oct*[aves], *Min* and *Max*). The accepted data type is noted in short form (e.g., *V[Float]* for floating-point value or *G[Int]* for an integer grid) behind the name of the input. In this example node, none of the inputs is wired from another node, and their values are all set to constants.

SimplexNoise has one output named *Noise*, which has the type floating-point grid. This node generates the image "Density" (Fig. 12).

In addition to the black inputs and outputs, nodes have so-called triggers (left) and ports (right) shown in grey. This special type of "inputs" and "outputs" handles simple trigger signals instead of data, allowing for complex control structures such as iterations and feedback loops. For more details, see Mosconi & Porta (2000).

Data Flow

Data flows from outputs to inputs are shown as curved connections in Fig. 12. As soon as a node has all its inputs available, it "fires" by calculating its outputs and sending the results to all nodes wired to the outputs. In this way, data are propagated through the graph and transformed by nodes until no node has to fire anymore.

In the example in Fig. 12, the program works as follows:

1. Tick
    a. *SimplexNoise* has all inputs available. It fires and generates the map "Density". The result is propagated to nodes *Normalize* and *Viewer [Density]*.
2. Tick

a. *Viewer [Density]* has all inputs available. It fires by opening a new window showing the image "Density" with the screen position and dimensions given by its inputs (*X*, *Y*, *Width, and Height*). *Viewer [Density]* generates no output.

b. *Normalize* fires and normalizes the output of *SimplexNoise* to the range [0, 0.005]. The result is propagated to *RandomPoints*

3. Tick

   a. *RandomPoints* fires and generates a grid initially filled with -1 (input *NoData*). Then, each grid cell is set to a unique ID with the probability given by input *Density* (received from node *Normalize*). The result (*IDs*) is propagated to *VoronoiDiagram*. The additional output *Points* (1/0 for point/no point) is not used further.

4. Tick

   a. *VoronoiDiagram* fires and constructs a Voronoi diagram (output *Cells*) for the point IDs generated by *RandomPoints*, i.e., each grid cell receives the ID of its closest point. Additionally, the distance to the closest point is generated as output *Distance*. *Cells* is propagated to nodes *Viewer [Cells]* and *AsciiFileOutput*. Distance is propagated to node *Viewer [Distance]*.

5. Tick

   a. *Viewer [Cells]* shows image "Cells".

   b. *Viewer [Distance]* shows image "Distance".

   c. *AsciiFileOutput* writes the Voronoi tessellation to file voronoi.asc in Esri ASCII raster format.

6. Tick: No node can fire, so the program is completed and ends.

Note that points a, b, ... of each tick have no particular order and can inherently be processed in parallel, taking advantage of multi-core architectures.

Groups

A feature not shown in the example is "node groups". A node group is an assemblage of nodes encapsulated to be handled as the node itself. Node groups can have inputs and outputs just like ordinary nodes and can be collapsed/minimized in the GUI to appear like nodes.

The purpose of node groups is twofold:

1. Closely related nodes in a program can be grouped to sub-programs and minimised to make the graph clearer.

2. Node groups can be saved to a file and exchanged between users. Later, they can be added to a program graph like ordinary nodes, making VDP programs more modular and reusable.

Automation

In the context of ALGs, a VDP program would often have to run many times, with or without parameter variation. Through the above-mentioned iterations and loops, such automation can be accomplished in the VDP program itself. However, because the program graph and its components (nodes, links) are accessible as a hierarchical data structure that can be manipulated from Java code at runtime, more-convenient means of automation can be easily implemented. Desired possibilities include:

- Setting node inputs when running a program from the command line
- Parameter files that specify node inputs and their variations
- Adaptation of the user interface, allowing one to set variations for inputs