Structured Argumentation for Simulation-Based Research

Teodor Ghetiu

Doctor of Philosophy

University of York

Department of Computer Science

September 2012

Abstract

Scientific research is a vanguard domain of human activity. Researchers aim for a systematic, objective approach, but also for pushing forward the boundaries of knowledge through the use of ever-more advanced instruments and techniques. Computer simulations represent such scientific instruments, capable of harvesting information in response to questions beyond the scope of traditional experimental techniques. The benefits of using them must, however, be considered together with aspects that led to criticism and lack of confidence e.g. they are difficult to analyse and validate, assumptions are only partially managed.

This thesis scopes down the vast domain of simulation-based research, to the use of agent-based simulations for studying complex systems. The use of structured argumentation in a scientific research context is studied as a means for addressing the core limitations of simulation-based research. The Goal Structuring Notation has been used effectively in its originating domain – Safety Critical Systems – in addressing similar problems to simulation-based research. Through the use of this notation, this research emphasises the difficulty of expressing compelling arguments, even in journal publications; in addition, it propose a set of extensions to the notation in order to adapt it to the scientific discourse. Finally, it shows the implications of studying a model in a rigorous, exhaustive manner, over the claims that can be made through it.

Contents

List of Tables

List of Figures

Acknowledgements

To Mihaela, the starting point of this long and life-marking journey. To Cristina and Corina, for being close throughout.

To Fiona and Jim, for infusing hope, creativity and humanity in all our interactions.

To all members of the CoSMoS project, for their collaboration, openness and experience in bringing smiles about.

Author's Declaration

The material presented in this thesis has not been previously submitted for an award at the University of York or any other institution. The author notes that Chapter 6 has been previously published in the following articles:

- *Equivalence arguments for complex systems simulations – A Case-Study* [36]
- *Simulation validation: exploring the suitability of a simulation of cell division and differentiation in the prostate* [68]

Chapter 1

Introduction

Knowledge is *important*. The reality of each human achievement and innovation is based, amongst others, on our capacity to acquire knowledge and to use it for achieving our aims. This also leads to differentiating between the *purpose* for which knowledge has been acquired, and the multitude of purposes for which it is being used; the history of social or environmental achievements (or disasters) can only strengthen this point. If the important topic of the *purpose* belongs to a different discussion – an ethical, moral, philosophical one – this thesis is focused on addressing the topic of **obtaining and conveying knowledge, driven by scientific interest**.

The domain in which this thesis is framed is complex systems research. A complex system is a living or artificial system that "cannot be fully understood simply by analysing its components" [29, p. viii], due to the interactions among its constituents or between the system and its environment. Such a system differs from a complicated one, as these relationships "are not fixed, but shift and change, often as a result of self-organisation" [29, p. viii], ultimately resulting in novel features (emergent properties). Complex systems are found everywhere in the biological realm, but also in engineering or at the frontier between the two (as encountered in Artificial Life (ALife research [55]); this emphasises also the scope of complex system research.

Due to the difficulty of studying complex systems, modelling and simulation (M&S) are often used together with (or instead of) traditional experimental research techniques such as *in vivo* or *in vitro*; an increasing number of scientific disciplines have adhered to the use of M&S for research purposes over the past decades. This type of approach, called in this thesis simulation-based research (SBR), delivers means only imagined before: models concentrate the conceptual and creative abilities of their authors which, coupled with the disrupting processing power of modern computers, enables the (low-cost) study of potentially any topic. It is understandable then why SBR is so appealing to a part of the scientific community.

If SBR yields itself to the study of speculative realities [18], it is also used in supporting risk analysis activities in disciplines such as toxicology, or hazard-analysis activities such as in safety-critical systems (SCS), [12]. The role of SBR can become critical, albeit to a different extent, also in more "traditional" fields of research, where scientific theories are formulated based on SBR results. If these are flawed, consequences follow [45].

No benefits come without costs and SBR's potential is also marked by limitations: computer simulations are more difficult to analyse [27], understand and communicate than their analytical counterparts; this, in turn, affects their reproducibility (a pillar of scientific research) and credibility [40]. Flawed SBR [45] or less suiting modelling paradigms [39] can also have long-term detrimental effects: generations may end-up learning from and using flawed knowledge. Uncertainty is also an important challenge for SBR [82]; there may be insufficient data for a "good" model to be built on or validate against. Last, but not least, "all models are wrong, but some are useful" [24, p. 424]. This is why adding rigour to the process of obtaining and conveying knowledge through SBR is important.

1.1 Simulation-Based Research (SBR)

SBR stems from the use of modelling and simulation (M&S) techniques in scientific research. Having already become indispensable in domains ranging from engineering, to military or motorsport, where they are used for more traditional purposes such as assisting design, identifying and optimising solutions e.g. [59], supporting the training of personnel or predicting scenario outcomes, M&S techniques are becoming ubiquitous through scientific disciplines[1].

SBR complements or (in cases) replaces traditional experimental approaches such as *in vivo* or *in vitro*, contributing to the rise of what has been coined *in silico* experimentation. In this thesis, the author adopts the more expressive title of simulation-based research (SBR), when referring to scientific research employing M&S techniques.

While not being a separate discipline, SBR is a type of research activity that integrates within the established methods of scientific research[2], where primary (or publishable) results are obtained from computer simulations rather than from (or in addition to) probing directly the (often inaccessible) reality of the problem being addressed. SBR is, consequently, an *indirect* and *analogous* (Section 1.2.1) way of performing research: studying a reality by evaluating a different one.

1.1.1 Scope of SBR

M&S, the core of SBR, is prominent amongst traditional engineering domains; there are conferences dedicated to it (e.g. the Winter Simulation Conference,

[1] While the domains listed before e.g. military, motorsport, carry out their own research and development programmes, when referring to "scientific research", we mean in this thesis the natural and social sciences, especially those focused on studying complex systems.

[2] Adapted to each discipline in particular. As Medawar has pointed out [60], there is no unique scientific method, guaranteed to turn questions into answers; the "scientific method" is more accurately described as a scheme for explaining systematic research (*a posteriori*) rather than conducting research, as research is not an objective process (subjective, and consequently informal, aspects of human nature, such as intuition or imagination, come into play in the research process).

the Summer Simulation Conference), journals (e.g. Modelling and Simulation in Engineering), and even university degrees (masters and PhDs). The impact of M&S has gone far and wide, such that for some purposes, it has become indispensable; in aircraft design or motorsport, for example, Computational Fluid Dynamics (CFD) simulation techniques have become mainstream since the 1960's; more recently, the National Science Foundation has reported on the precepts for a "Simulation-based Engineering Science".

1.1.1.1 Traditional Application Domains

There is extensive research employing M&S in a number of fields. Examples of themes for which M&S is used are:

- Healthcare: healthcare modelling, epidemic modelling, design of healthcare systems, simulation of patient flow, of ambulance services or emergency departments.

- Social Science and Organization: economics and management, planning, social behaviour;

- Construction: simulation of construction operations, of construction scheduling; energy simulations and simulation in health and safety.

- Production and Logistics: warehouse logistics and inventory management, shipbuilding and maritime applications.

- Environmental: power grid simulations, decision support, traffic

- Military: military analysis, combat modelling and mission analysis, military simulation methodologies, military logistics

Other themes for which M&S is used are computer networks and computer systems, manufacturing and transport.

1.1.1.2 Applications in Complex Systems Research

SBR is often applied to the study of complex (natural or artificial e.g. virtual or physical) systems and phenomena. The aim of such studies is to address hypotheses concerning unknown (or partially known) structures, processes, mechanisms and interactions concerning entities and environmental conditions – physical (biotic or abiotic) or artificial – that generate the observable complexity e.g. identifying the factors generating a type of cancer, studying the evolution of life forms. Seeking knowledge on partially observable, partially measurable complex systems is different from and at the same time more difficult than seeking to optimise or engineer[1]. The study of complex systems can be argued as being the most difficult task M&S is applied to and a disciplinary approach is generally insufficient for answering its key questions.

The above suggests that SBR is, generally, of *interdisciplinary* nature: it is both widely spread throughout scientific disciplines, and constitutes a platform for interdisciplinary research, bringing together scientists from different disciplines. Computer science skills are an implicit requirement for this interdisciplinary equation.

Examples of such M&S range from simple, small-scale models such as the Game of Life [44] to large and multi-scale, multi-model simulations e.g. [11] in ecology. Among simulators used in studying natural evolution, Tierra[2] is a an established example in the ALife domain.

[1]Research is more difficult (albeit not independent from) engineering in the same sense that obtaining knowledge through the use of tools is more difficult than building tools through the use of knowledge.

[2]http://life.ou.edu/tierra/

1.1.1.3 Thesis scope

For the purpose of this thesis, complex systems research employing M&S techniques is targeted. Arguably, this scope is too wide[1], so the scope is narrowed down to the use of agent-based models (ABMs), in particular for ecological research. Ecology has shown both enthusiasm towards M&S and great motivation towards pushing the boundaries e.g. developing larger, more complicated M&S, simulating more scales in order to better accommodate the vast areas of land relevant that ecology can study.

1.1.2 Models

There are many classes of models used for scientific research. Works such as Cooley and Solano [31] or Jakeman et al. [51] provide a short overview of these techniques. For the purpose of this thesis, models are only classed into formal and semi-formal models; representative for formal category are equation-based models, whereas agent-based models exemplify semi-formal models.

1.1.2.1 Equation-Based Models

Termed equation-based models [67], explicit mathematical treatments [27] or simply mathematical models, these are the most common representative of scientific formal models. The reality they model is described as a system of mathematical equations. An example is the Lotka-Volterra model (also known as the predator–prey model), describing the dynamics of two interacting species, one being a predator and the other its prey. Their evolution in time is expressed through the following interdependent equations:

[1] Since nature is formed of complex systems, all natural and social sciences can be considered as being instances of complex systems research.

$$\frac{dx}{dt} = x(\alpha - \beta y)$$
$$\frac{dy}{dt} = -y(\tau - \delta x)$$

Here it is important to distinguish between non-spatial and spatial models. The mean field approach adopted in the above model suggests that every individual has equal probability of interacting with every other individual, assumption which speaks about the idealistic "average" individual. Criticism of this modelling paradigm focuses on the fact that it "violates the biological principle that each individual is different, with behaviour and physiology that result from a unique combination of genetic and environmental influences" and "violates the biological principle that interactions are inherently local" [50].

1.1.2.2 Agent-Based Models

At the other end lie semi-formal models: models described in natural language, but which reflect a degree of formalism e.g. they contain algorithms which may contain mathematical equations. Agent-based modelling (ABM) is a relevant approach for this class of models. Different from mathematical models, where variables were used as a summarising or homogenising mechanism (e.g. in the Lotka-Volterra equations, x and y are population-level variables, describing the number of the predator and respectively prey), one of the main features of ABM is that of accounting for agency: individuals can be represented explicitly, meaning they can be different (heterogeneous).

This couples well with the use of spatial representations; individual agents interact locally rather than having equal effects over all the other agents, as often assumed in mathematical models [50]; the Lotka-Volterra model, previously described, can be converted from a population- level, non-spatial representation, to an ABM where agents have behaviours and population dynamics emerging from their interactions within an environment (as shown in Figure 1.1).

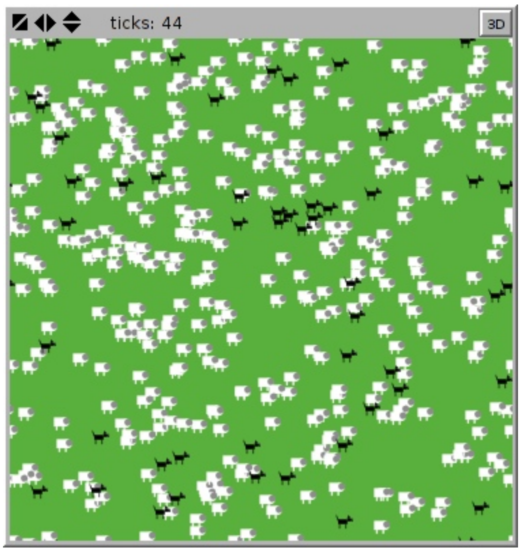


Figure 1.1: Visualisation of a agent-based representation of the Lotka-Volterra model

In a review on the use of ABM for ecological modelling spanning ten years of research, Grimm [39] differentiates between pragmatic and paradigmatic motivations for using ABM[1] instead of EBM. Pragmatic reasons are those that lead to ABM utilisation because it allows the study of aspects inaccessible through EBM e.g individual variability, local interactions, spatial representations [20]. Paradigmatic reasons are those that see ABM be more desirable than EBM because of "the suspicion that much of what we have learned from state variable models about the theoretical issues mentioned above, e.g. regulation etc., would have to be revised if the discreteness, uniqueness, life cycles and variability of individuals were to be taken into account, as well as the fact that most interactions are local and that space matter" [39].

There is a trade-off between how easily these models can be analysed and de-

[1]In the review, Grimm actually focuses on individual-based modelling (IBM), which ecologists consider a sub-type of ABM that is defined by the fact that agents are discrete individuals, whereas ABM allows agents to represent communities of individuals, organisations or any other composite group. Durrett and Levin actually identify three levels of agency: "patch models that group discrete individuals into patches without additional spatial structure; reaction-diffusion equations, in which infinitesimal individuals are distributed in space; interacting particle systems, in which individuals are discrete and space is treated explicitly" [33]

veloped, following Breitenberg's "law of uphill analysis and downhill invention" [25]. Mathematical models are harder to construct, but their analysis becomes straightforward as they are formal systems; ABMs on the other hand are easy to develop, but harder to analyse (the models may not be defined according to a formal language, but through a natural language; the resulting computer simulation most often is written in an ad-hoc manner, with potential unidentified flaws; if off-the-shelf software is used, transparency may be reduced, etc.).

1.1.3 Positives of SBR

Versatile M&S can be applied to any domain. Since models are necessarily wrong [24, p. 424], the whole purpose of modelling is that of simplifying a problem domain, and this implies losing information [83]. In fact, M&S have been advocated as a way of "extending ourselves" [49], in the sense that simulations can be used as scientific instruments, in a similar manner in which other instruments have been used throughout history; no instrument is perfect, but described by a range of accuracy (or error) and simulations fit this description.

M&S can also be applied to scenarios (parametrisations or assumptions) that might never occur in nature [83, citing (Zelditch and Evans, 1962)]. Furthermore, there is an extensive literature on the use of M&S for studying virtual realities. The domain of ALife is essentially aimed at studying "life as it could be"[1], and in doing so it capitalises on the potential insights gained from intentionally unrealistic models [18].

Powerful M&S can be used for expanding or compressing time [83, citing (Mize and Cox, 1968)]. With the simple change of a "time" parameter, the entire history of the universe can be traversed in a matter of passing moments. These expansion or compression properties are applicable also to the time-related dimension of space: we can perform studies at a molecular [71] or particle scale, as well as

[1]Although it often succumbs to (and builds on) studying life as it is.

spread the simulation compass over ecological or cosmic scales, using different spatial [20] or temporal representations.

Another important benefit of this approach is that it requires fewer assumptions and simplifications compared to classical approaches. As Levin et al. put it, "classical approaches to population biology – like classical approaches to other problems in biology – emphasized deterministic systems of low dimensionality, and thereby swept as much stochasticity and heterogeneity as possible under the rug" [58]. Even if their view is focused on the domain of biology, it is not inappropriate to extrapolate it towards other domains that are taking up SBR.

Easier to develop As deriving from the law of downhill development and uphill analysis [25], ABMs are said to be easier to develop than mathematical models, since they can be constructed from simple observations and assumptions [39]. While mathematical models can also be very simple (the above Lotka-Volterra model fits this description),

Non destructive No living creatures are affected during the process of SBR – as long as validation or parametrisation of the underlying M&S does not require the procurement of empirical data through destructive observation techniques e.g. taking measurements of a living creature (plant or animal), that lead to its death.

Experimenting through M&S is also cheaper and safer [66] than through prototypes; for some scenarios e.g. nuclear war, it is indeed undesirable that such physical experiments are carried out.

1.2 SBR Problems

As many as are the positives of SBR, there are a number of aspects that make its use problematic.

1.2.1 The Analogy Problem

Previously we have mentioned that SBR is an indirect and analogous way of performing research, as it studies a reality by evaluating a different one. Analogy is not a new approach, nor is its use restricted to SBR; in fact, there are claims that the use of analogy has marked the beginning of research activities, for humanity [17]. If such claims are harder to probe, what is clear is that analogy has been used in the natural sciences and, albeit criticism from different pragmatic views, it has withstood and proved a fecund approach for performing research [48].

The analogy problem does not bear a real solution; it is a state of facts which can be summarised by the famous aphorism "all models are wrong, but some are useful"[24, p. 424]. All the reasons for which models are indeed wrong are subsumed by this expression; they are simplifications, always limited compared to the system being studied; they are of a different nature and made of a different fabric compared to the modelled system e.g. conceptual models, digital simulators versus living organisms (see research in the ALife domain for more in depth discussions on wet-lab (traditional) versus dry-lab (computer-based) experimentation e.g. [18; 55]).

1.2.2 The V&V Problem

V&V are "two crucial parts in the process of performing a simulation study" [61]. The topic of validity – arguably the more important of the two – has been a challenge since the beginnings of M&S [83], when validity was called "the most elusive of all the unresolved problems associated with computer simulation techniques" [83, quoting (Naylor et al., 1966)]. Almost five decades later, the situation has not radically improved. V&V remains a challenging topic both on a *paradigmatic* level – converging on commonly accepted definitions for the two terms, and a *pragmatic* level – establishing commonly accepted, effective techniques for probing into these two aspects.

In the context of complex systems simulations, traditional V&V techniques do not suffice, as expanded further down. Consequently, there is a need for heuristic validation or, more concretely, for *justifiable confidence* in the validity of a M&S effort.

1.2.2.1 The Paradigmatic Side of the V&V Problem

From a paradigmatic perspective, we find that the verification and validation terms bare different (sometimes similar) meanings, in different domains. In software engineering, for example, Beizer [19] made the famous distinction: verification assesses that a software program was built *right* and validation assesses that the *right* program was built. The two concepts are complementary, one not implying the other, although in practice, verification precedes validation. Moving towards a more research-oriented, scientific view, Stanislaw [83] differentiates between program, model and theory validity, criticising those seeking *one* type of absolute validity. Sargent [77] adds data validity to this equation, while replacing Stanislaw's three-faceted validity with a conceptual model validation, computerised model verification and operational validation sequence (that do not fully overlap with Stanislaw's proposals). Sargent also describes works that associated validity with scores; a model is subjectively scored through different tests, its overall validity being determined by its (overall and category) scores being greater than a passing score; Sargent mentions, however, that this approach is not used in practice frequently.

Sales et al.'s [61] view verification as an activity aiming to "debug the model"[1], while validation assesses that the model is sufficiently accurate in representing a system being studied; in light of Stanislaw's differentiation between program, model and theory, Sales et al. confuse program verification with model verification, whereas validation refers only to the model, making no reference to the validity of theories underlying the model. Going further, Cooley and Solano [31] summarise a different subset of validity definitions, focusing on Zeigler's approach of distinguishing between replicative validity, predictive and structural validity.

[1]They actually assess if the conceptual model was implemented correctly.

Stepping into a domain such as ecology, where scientists have been developing and using software simulators in their research, Schmolke et al. [80] baptise the process of assessing that a "model is working according to its specifications" as verification, leaving validation to represent the evaluation of the model against novel scenarios.

These views represent only a small sample from the pool of paradigmatic opinions concerning the topic of V&V, but a sufficient one to clarify their implicit diversity. Diversity is beneficial (especially biological diversity) but can often have negative effects due to a high degree of relativism (or lack of standards). Even amongst computer scientists, verification and validation are often used interchangeably, thus creating more confusion: Stanislaw [83] criticises authors that fail to make the difference between program, model and theory validity, but then makes no reference to verification (apart from mentioning that a "program must be fully debugged"); Sargent [77] describes the dynamic verification of computerised models through the use of "validation tests"; Cooley and Solano [31] address the topic of model verification by pointing towards the need of debugging the simulator (note that, as highlighted by Stanislaw, a model and a software simulator are two different entities).

To summarise, the paradigmatic side of the V&V problem refers to the meaning, scope and dimensionality of the two terms. To stem out of the different connotations that the two terms may have, these should be defined explicitly and justifiably, according to the purpose of the study. Unsubstantiated claims for the "validity" of the simulation are no more unhelpful than substantiations that do not justifiably fit the purpose of the study.

1.2.2.2 The Pragmatic Side of the V&V Problem

From a pragmatic perspective, innovation in the V&V domain has not been revolutionary, in recent times; the critical mass of V&V literature has been published in the second half of the 20^{th} century, when the majority of tests available today have been introduced as [16; 76; 83] point out. With respect to V&V testing, no "silver bullet" test or V&V strategy exists; in fact, the detailed and articulate view reflected in these early contributions seems to have diluted with the spread of computer simulation techniques towards other scientific domains, as suggested by the meanings of verification and validation adopted by ecologists [80].

V&V is problematic also due to *incompleteness*, referring either to M&S specifications or validation data. Since M&S built for research purposes are speculative to a degree, being aimed at reducing the uncertainties in knowledge, their specifications are necessarily incomplete.

In addition, there may be no sufficient or adequate data through which to validate a M&S [31; 77]. ALife research, for example, intended to study "life as it could be, not as it is", may address scenarios of virtual worlds that are highly divergent with our reality; validation data may not be available in such cases, or even in some non-ALife SBR addressing real-world but out of scope problems (data being be too costly or impossible to obtain). It can be said then that validation testing is purpose and data dependent. In traditional engineering, data is considered valid if it is appropriate, impartial, complete and accurate (and the model uses it correctly [15]; in a complex systems research environment, such objectives often remain in the realm of "high ideals".

Furthermore, the way stochasticity is represented in a simulation can have a relevant effect. Some simulation outcomes may depend on the choice of the initial random number (seed), especially in the case of simulations that are path-dependent [31]. Stochasticity also implies that expected results are not available; consequently, it is difficult, if not impossible, to formally verify complex systems simulations.

Another pragmatic question regarding V&V is who should perform this process. Sargent [77], for example, describes three types of approaches: V&V performed by the development team, by the model's users in collaboration with the development team, or independent V&V (IV&V), which is carried out by an independent third party.

In addition to the paradigmatic and pragmatic facets of the V&V problem, we need to mention the (ignored) *criticality* problem: although awareness exists in terms of the importance of a detailed approach to V&V (see [39; 79] for detailed reviews), SBR literature provides evidence for a rather superficial, shallow attitude towards V&V.

1.2.3 The Assumptions Problem

This refers to the way assumptions (modelling but also argumentation assumptions) are managed and communicated. Regarding the former aspect, at the current stage of SBR maturity, we cannot even speak of a management of assumptions. If managing equates with "being in charge of", as well as with "maintaining control over" or "controlling the use of" (Oxford dictionary), then SBR is certainly not an exemplar of such an assumption-oriented activity.

To be in charge of something, one needs to know first what it is in charge of: in this case, the (modelling) assumptions. Irrespective of the SBR study, we can not find an explicit listing[1] of *all* the assumptions that underlie it (so that we are presented the individual assumptions to be controlled by the authors), nor an explicit, justifiable criteria used in choosing to make public an assumption or not (to differentiate between explicit and implicit assumptions).

Furthermore, an individual account of the rationale and support behind *each* (or most) of the assumptions is not usually be provided; at best, some assumptions (of interest to authors) will be justified, but an explicit, justifiable reason for which the other assumptions were not explained, will not be provided. Since we are not aware of the *number* of assumptions that go into a study, knowing what and how many assumptions have been tested will not be sufficient for accurately approximating the uncertainty of the model and this uncertainty will, more often than not, be disregarded in the concluding claims of the study.

The above contour also the communication part of the assumptions problem:

[1]Be it in the main body of text or provided as supplementary information.

there is a lack of assumption transparency [13]. Not all assumptions that are declared explicitly are also discussed, as emphasised in Chapter 2.

Albeit assumptions constitute an essential dimension of research validity, since both in computer science [77; 83] and in other scientific disciplines e.g.[40; 80], assessing assumptions is mentioned as essential towards claiming validity, little evidence is provided that this is done on a systematic basis. If assumptions are not managed, model interpretation is rendered more difficult, opening the door for further flaws. There are cases of published SBR that are shown to be flawed due to inadequately treated assumptions, such as the case analysed by Bullock and Cliff [28], or even blatantly flawed, from premises to conclusions, as is the case criticised by Ostermann-Kelm et al. [63].

1.2.4 The Argument Problem

Represents a three-fold problem. Firstly, in SBR publications, arguments are scattered, difficult to access or inaccessible (beyond what constitutes the top claim of an argument). Scattered claims may span the length of a research publication, but their corroboration does not equate to obtaining cohesive, compelling, detailed, first-class arguments. SBR authors do not explicitly provide such arguments and, in turn, this may affect the reviewing and understanding of their work (much as an inconclusive description of a simulation model may prevent others from replicating it [89; 91]).

Secondly, the arguments put forward in SBR contain many leaps of faith [1, Annex 2.6.4] that are not filtered by the peer-review process. The analogy problem constitutes itself an *intricate* and irreducible leap of faith, accepted by all modellers; however, accepting this situation should not constitute grounds for making claims that extend this "leap" beyond an acceptable boundary. In line with SCS, in complex systems research, which is practically ubiquitous throughout scientific disciplines[1], consequences may be undesirable. In addition, arguments can simply

[1]Every physical object is eventually constituted from a multitude of interacting particles; interaction takes place both on the microcosm scale and the macrocosm scale

be shallow, ambiguous, too general or absolute; they may contain terms with implicit (and often unclear) meanings, but also fail to specify the assumptions they make. There are numerous aspects that can affect the quality of an argument, a complete listing going beyond the scope of this thesis.

Finally, even if scientific arguments were highly accessible and sound, another question is: does a study put forward *all* the arguments that are necessary for gaining confidence in it? Are, for example, arguments for the fitness-for-purpose of a given work, or for the contribution of that work to the wider understanding of the domain problems?

Note that, albeit the other SBR problems have received considerable attention in the literature, the domain and importance of arguments has been under-explored. In addition, weaknesses in the argumentation layer of a SBR effort, can be associated with one of the three following cases: a) the research was "good", but the arguments were "diluted and dissipated" in the write-up process, b) the research was "bad", hence its published report could not have put forward "good" arguments, in the sense hinted above and c) the problem lies somewhere between cases a) and b). While the first two cases are necessarily (optimistically) exaggerations, the latter case carries the biggest relevance, and is in line with the thesis put forward here.

1.3 Thesis statement

1.3.1 Motivation

All of the problems signalled in Section 1.2 are important; while they do not represent an exhaustive listing of SBR issues, they point towards interrelated areas of concern.

For the purpose of this thesis, the argument problem (Section 1.2.4) represents the focal point. To an extent, all of the other mentioned problems can be (retrospectively) related to the argument problem as *arguments and their supportive evidence summarise (and justify) the essence of a research effort.* In other words, a deficiency in addressing an SBR issue can be reduced to the 1) *lack* of an argument supporting the adequate treatment of the issue e.g. for V&V, an argument coupled with evidence in favour of the adequate addressing of V&V, or 2) a *flaw* in the argument or evidence [53; 87; 88] supporting the adequate treatment of the issue (the fallacy concerning the argument's reasoning, evidence or a combination of the two).

Finally, if justifiable confidence in SBR is needed, to an extent that researchers can express clear and compelling arguments supporting the adequacy of their work, inspiration can be taken from other domains which 1) work with complex systems 2) in a manner that requires such confidence. The area of critical systems engineering (or safety critical systems – SCS) is such a domain. SCS provides an extensive literature on the use of structured argumentation (SA) for the purpose of conveying, in a clear and comprehensible manner, why engineered systems are safe, dependable, secure, etc. (as addressed in Chapter 3). Documenting such properties is a requirement and the successful assessment of the resulting report e.g. a safety or dependability case, is what leads to the certification of the system. Moreover, failure to carry out correctly or to analyse effectively such documentation, can have disastrous consequences e.g. the Nimrod disaster [43].

If in SCS, standards in terms of expressing arguments and evidence – such as SA – are better than SBR's, then SBR can and should learn from them.

1.3.2 Statement

This thesis is putting forward the following statement regarding structured argumentation (SA):

SA is needed for the development of rigorous SBR and for adequate communication of SBR

The reminder of this research aims to substantiate why and how SA can be useful in the context of (performing and communicating) SBR. While hard evidence for the "need" of using SA in SBR is beyond the scope of this work – only a systematic and continued use of SA for SBR purposes can yield such evidence – claiming this need is based on our expectation that the benefits of using SA outweigh the costs (and risks [52]). Throughout the rest of the thesis, the "need for SA" will bare this meaning.

1.3.3 Thesis Plan

In addressing the thesis statement, a case study-based approach is adopted. Theoretical insights are useful, but experience gained through the analysis of a case study is fundamental. For this purpose, a SBR study published in a high-profile journal (Chapter 4) was used.

The argument problem was highlighted as a focal point; it is also the starting point for this research. Substantiating the existence of this problem (the claim that SBR publications, while delivering information necessary for gaining confidence in their quality, do not excel in providing first-class, compelling arguments) is pursued by using a SA notation for extracting the argument threads from our case study (Chapter 5). Insights are gained, both into the suitability of the SA notation to SBR (as this has been developed for the SCS domain, and not for SBR), but also on the structure of SBR arguments – and on the existence of the argument problem.

Establishing that there is indeed an argument problem does not necessarily mean that its source has been identified; the information content of a research can be distinguished between pre-publication and publication, but published research issues cannot be fully isolated from issues existing during the research. In other words, it is not that researchers had *justifiable confidence* in their work, but this lost its clarity or comprehensibility during the write-up process. The author aims then to show how and why SA is useful in other stages of SBR; assuming that the benefits of SA outweigh the costs, then we can claim SA is needed – SBR problems are important and need to be addressed.

In addressing the two main claims of the thesis (that SA is needed for the a) rigorous development of SBR and b) adequate communication of SBR), we then focus on the V&V and Assumption problems. For the rigorous development need, we investigate the use of SA in expressing fitness-for-purpose arguments (Chapter 6), together with assurance-giving sub-arguments, while clarifying the assumption problem (Chapter 7) will provide further substance to these arguments. The adequate communication need is substantiated through solving the argument problem.

1.4 Thesis Outline

This thesis proceeds as follows:

Chapter 2 describes an original survey of research articles published in 2011, that use the Overview, Design Concepts and Details (ODD) protocol [40] for describing IBMs. The survey has a two-fold aim: its immediate objective is to gather information on ODD's uptake within the scientific community e.g. spread across domains, number of papers using it, the accuracy with which its specifications are being followed; more importantly, the survey seeks to identify the relation between the use of ODD and the rigour or studies (described by V&V activities, by the provision of justifications, the transparency of assumptions, etc.). We seek to scope the role of SA for SBR publications, among that of documentation standards such as ODD (Section 2.1).

Chapter 3 focuses on SCS and the way SA became a tool of choice in the construction of safety cases. GSN is introduced and its use in constructing safety cases explained.

Chapter 4 is a bridge between the first part of this report – the background one – and the second part – the original contributions one. It presents the case study used throughout the remainder of the thesis.

Chapter 5 presents a systematic reverse-engineering method, introduced by the author in order to converting the argument threads of a publication into GSN arguments. By using the method, insights are gained into the necessity for using SA in SBR: if the reverse-engineering would easily produce compelling arguments, this would imply SA is not needed as the plain text form in which information is laid out suffices. The chapter's results suggest the opposite.

Chapter 6: shows why SA is needed in the exercise of developing rigorous SBR. Exemplification takes place through the development of equivalence and fitness-for-purpose arguments.

Chapter 7: provides further evidence in support of the claim that SA is needed for communicating SBR. Building on the GSN arguments obtained from the reverse-engineering exercise in Chapter 5, the chapter proceeds by thoroughly assessing the case study model's assumptions, using a customised version of the Hazard and Operability Study (HAZOP), an established SCS hazard analysis technique.

Chapter 8 concludes the thesis, summarising the contributions and detailing further research avenues.

Chapter 2

Survey of SBR articles using ODD

This chapter is aimed at presenting new evidence on how aspects of research rigour, such as V&V and assumption management, are treated in SBR publications that adhered to a more systematic and standardised way of presenting their information. The criteria for selecting these publications is that of using the Overview, Design Concepts and Details (ODD) protocol, a wide spread model description format intended to promote rigour (Section 2.1). Previous surveys e.g. [39; 79] have already emphasised that a low proportion of ABM research publications are taking into account and thoroughly addressing topics of V&V nature; in this context, use of documentation formats such as the ODD should, in theory, correlate with a more rigorous delivery of research altogether.

The survey described in this chapter does not represent a comparison between publications that use ODD and ones that do not. ODD is used as a criteria for selecting publications, which are then subjected to our analysis. Apart from assessing the way V&V aspects are considered, the survey addresses the volatile but essential topic of assumptions. Assumptions come in different flavours: albeit the most prominent in the SBR domain are modelling assumptions, assumptions can be found in arguments, in the way part of research data and knowledge is abstracted away from publication, etc.

Secondary aims include assessing how well ODD's specifications are being followed, on the assumption that, if ODD usage is largely according to its specifications, then it is sufficient for its purpose. If that is not the case, then there is room for it to be complemented by further structured information: structured arguments.

The chapter proceeds as follows: Section 2.1 introduces the ODD protocol; Section 2.2 details findings from previous relevant surveys, whereas Section 2.3 details the materials and methods used by this survey. Section 2.4 presents the findings of the survey, the chapter concluding with a discussion in Section 2.5.

2.1 The ODD Protocol

Coming from the ecological domain, the ODD protocol [40; 41] is a documentation format for describing ABMs. In a research domain abounding in models described in a (incomplete, ambiguous, etc.) manner that does not facilitate replication [39; 79], ODD has been aimed at becoming a standard that will raise the level of thoroughness and rigour. Introduced in 2005 and used, to date, in more than 100 publications that span disciplines such as ecology or the social and earth sciences, ODD is the product of a wide collective effort (28 authors on the original paper [40]).

ODD's specifications are defined so that the result is marked by a: 1) a succinct purpose, 2) a comprehensive model description, and 3) assurance of design decisions; these, in turn, enable the replication of baseline experiments via reimplemented simulator. The protocol is made up of three parts (see Table 2.1): firstly, the overview, giving a baseline understanding of the purpose and model constituents; secondly, the design concepts describes the way general concepts from the field of Complex Adaptive Systems are specified in the model, so that "each individual-based and agent-based model is integrated into the larger frame-

Overview	Purpose
	Entities, state variables and scales
	Process overview and scheduling
Design concepts	Basic principles, emergence, adaptation, learning prediction, sensing, interaction, objectives stochasticity, collectives, observation
Details	Initialization
	Input Data
	Submodels

Table 2.1: The ODD protocol (second version [41])

work of the science of Complex Adaptive Systems" [40]; finally, the details section contains information necessary for completely reimplementing the model and running baseline simulations.

More comparative information regarding the two versions of ODD can be found in Appendix A.

2.2 Previous Surveys

Grimm et al. published in 2010 a survey on ODD's usage [41], assessing publications from the year 2006 – when ODD was introduced –, to 2010. Their review, based on a pool of 54 articles containing ODD model descriptions, found that 75% of model descriptions were complete and correct (or only one ODD element was missing or incorrect), whereas 62% provided incorrect information for the Input element. The review focused only on the ODD model descriptions, not evaluating other qualitative aspects of the papers in which the models had been published.

In the same year, Schmolke et al. [79] performed a different review, this time addressing the modelling practise for pesticide risk assessment (a safety-critical domain). Looking at modelling aspects such as model type, complexity, parameterization and evaluation, they found that, out of 62 pesticide risk assessment models published between 2000 and 2008, only 3% had been validated, while 24% had not been evaluated at all by the authors. Verification had been performed more frequently than validation, but with only 26% of the models verified, there was no reason for excitement. Concerning parametrisation, 15% of the models did not make use of any empirical data, while the majority – 56% – were partially parametrised from empirical data.

In an earlier review of ecological IBMs, Grimm [39] found that, out of 50 selected articles, only 36% referred explicitly to validation or corroboration, this percentage being considering as "less than one might assume in view of the higher degree of realisms of individual-based models". One notable aspect was that all papers employing pattern-oriented modelling (POM) [42] had explicitly considered validation.

Relevant are also the reviews of Grimm [39] and Schmolke et al. [79]. In [39], published approximately ten years before [41], Grimm surveys ecological IBMs. Out of 50 articles, he finds that validation had been performed in only 36% of the cases. Although Grimm appreciates that this percentage "is probably much more than a random sample of 50 state variable models would show, but still less than one might assume in view of the higher degree of 'realisms' of individual-based models", which should be better testable. More recently, surveying pesticide risk assessment models, Schmolke et al. [79] find that out of 62 published models, 15 (24%) were not evaluated at all by the authors, 16 (26%) underwent verification whereas sensitivity analysis was conducted (at least on a subset of parameters) on 38 (61%) of the models. Validation using independent data was performed only on 2(3%) models.

All these reviews underline the V&V problem (Section 1.2.2), both from a quantitative perspective i.e. very small percentage of published models have been rigorously validated and a qualitative one i.e. models validated without using empirical data [79], model descriptions not according to ODD standard [41].

The survey presented in this chapter targets articles published in 2011 (previous surveys considered articles published up to 2010) so that results can be put in perspective with those published by earlier reviews. The author assumed that the absorption and diffusion of ODD throughout disciplines, as observed in [41], would extend further in time: new disciplines would start using it, ODD model descriptions would be more accurate and the positive qualitative impact of using ODD would be further emphasised. The survey was also motivated by the fact that Grimm et al.'s earlier review [41] had only covered the topic of ODD usage, not accounting for the other qualitative markers studied by [79] and [39], and consequently not providing a more comprehensive image of the rigour of papers using the ODD protocol.

2.3 Materials and Methods

For this survey, the Web of Knowledge[1] and Google Scholar reference databases were used in order to retrieve articles published in 2011 that cited the original ODD paper [40]. Out of the 105 publications, 40 did not contain ODD model descriptions (they only referenced the article in which ODD was introduced). The survey used the remaining 65 papers which contained ODD model descriptions and these are listed in Appendix B. Out of the 65 articles, 60 were journal papers, 2 PhD theses, 2 conference papers and one technical report. These were assessed according to the following characteristics:

2.3.1 Model Evaluation

In order to assess how the models were evaluated (or tested), the author assessed if verification, validation and sensitivity analysis had been performed on them. Different from other surveys that treat findings in a boolean manner, we use a fuzzy approach to recording findings; for each of the three types of evaluation, the following categories are used:

- missing: no reference to an evaluation method is made the article;

[1] http://apps.webofknowledge.com

• mentioned: references are made, but in theoretical terms or in relation to work published in other papers;

• claimed: the authors claimed they have evaluated (e.g. verified) one or more elements of their research, but have not provided evidence of that;

• supported: authors provided evidence that assures the evaluation.

Because of the discrepancy between the meaning of verification and validation, as used by some ecologists and computer scientists, these need to be further expanded.

Verification In their review, Schmolke et al. [79] consider a model as verified "if the whole or parts of the model output were compared to the data set initially used for model parameterization". They consider such a comparison test as evidence that the model "works as intended by the developers". While it can be accepted that this type of comparison provides information on the relation between a model's output and another data set, it can be shown that faulty models can produce such desired results. In fact, Grimm, co-author of [79], introduced in an earlier article the concept of pattern-oriented modelling [42], which is aimed at reducing the likelihood of such misleading situations altogether.

Schmolke et al.'s definition of verification overlaps with what a computer scientists calls validation; in fact, according to Sargent's view [77], such an understanding of "verification" spans actually over a full cycle of conceptual model validation, computerized model verification and operational validation. What the survey understands through verification is either Schmolke et al.'s view, or the literal computerized model verification.

Validation Ecologists refer to validation when comparing model outputs with independent (not used for parameterization) empirical data [79]. This equates to what others refer to as prediction [31]. Sargent also mentions data validation as a step in his modelling process [77]. In assessing validity, the survey adopts the former view, and notes that for a more thorough evaluation, each type of validity (conceptual model, data and operational validity) should be accounted for.

2.3.2 Assumptions

The survey assess the way assumptions are exposed, by identifying explicit declarations e.g. “we assume”, “it was assumed”. These references are then classed into different categories: assumptions regarding the author’s own work (e.g. about the published model) or about someone else’s work, assumptions that accompanied by a rationale (justification) and unjustified ones, assumptions that are discussed in the Discussion section of the article.

Additionally, the survey records the section in which each assumption was exposed. The ODD model requires the declaration of modelling assumptions, hence it is useful to note how well this requirement is implemented.

From the set of 65 papers, the author randomly selected 10 articles which were further assessed with respect to the number of assumptions they explicitly declared, the sections in which they were declared and the ownership of these assumptions (if they referred to their own model or to someone else’s work).

2.3.3 Contextual Information

In order to gain more contextual information, each article was assessed for which of the two ODD specifications had been used, what scientific discipline did the model belong to and which of ODD’s co-authors was also a co-author on these articles.

2.4 Results

2.4.1 General Information

Out of the 65 model descriptions, 17 (26%) used the updated format, whereas the remaining 48 (74%) used the initial ODD format. This may suggest that, although the update had been introduced in the previous year, the initial version of ODD had been sufficient for the majority of researchers employing it. Since the differences between the two versions are not many (see Appendix A for more details), one can hypothesise that the researchers had not been interested in following ODD's evolution – in making better use of it.

The distribution of articles across scientific disciplines is shown in Figure 2.1. It is observable that ecology remains the field where ODD has its biggest uptake, as identified by Grimm et al. in their review of ODD usage [41]. However, the number of disciplines embracing ODD has increased in comparison with Grimm et al.'s findings.

Figure 2.2 presents the number of surveyed articles that were co-authored by the researchers authoring the initial or updated ODD format. If Grimm et al. [41] found that in 13 out of 87 publications(24%), one of the original co-authors of ODD, was a co-author. This survey found that 10 articles out of 65 (15%), had been co-authored by one of the 28 co-authors of the original ODD paper [40], whereas the additional co-author of the updated version (Eva Rossmanith) had co-authored one article.

2.4.2 Verification, Validation and Sensitivity Analysis

Figure 2.3 and Table 2.2 present the way the three main topics of verification, validation and sensitivity analysis were considered in the surveyed papers. We notice that, out of the 65 papers, in 59 papers (meaning over 90%) the topic of verification was missing, while validation and sensitivity analysis were missing in

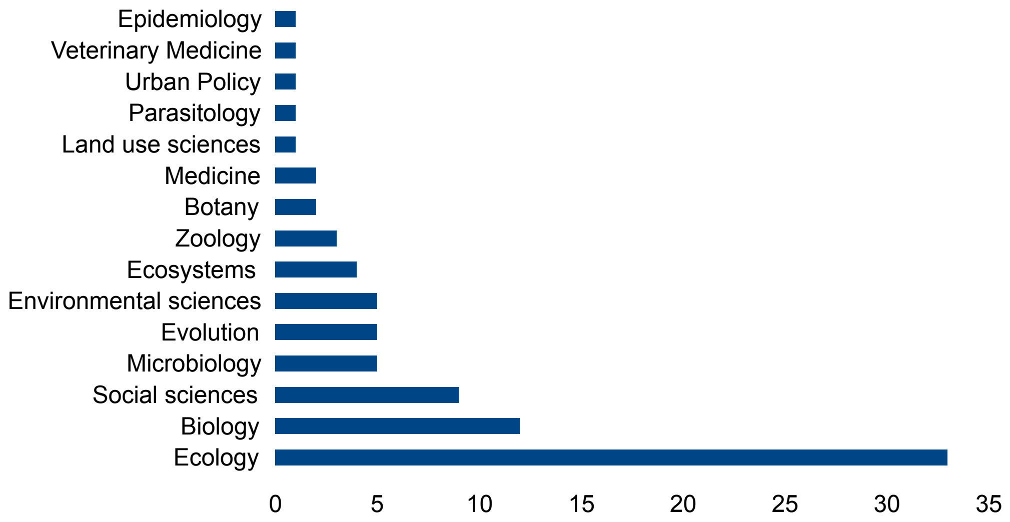


Figure 2.1: Distribution of reviewed articles across scientific disciplines.

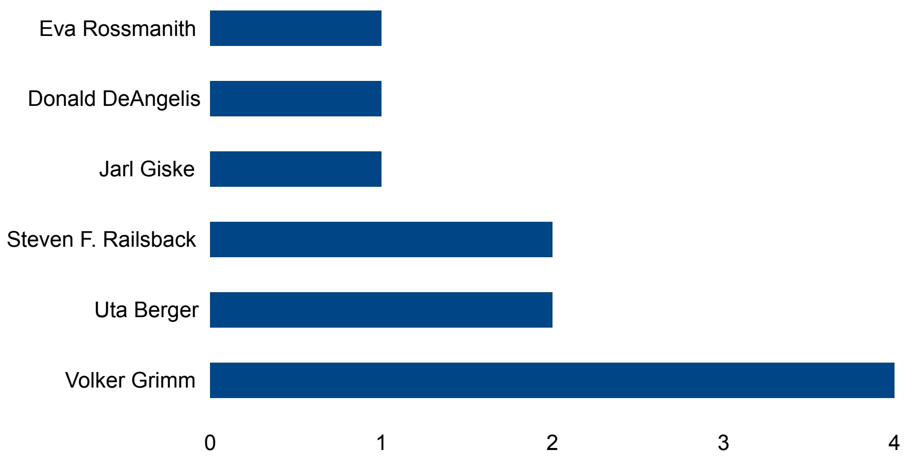


Figure 2.2: Number of surveyed papers co-authored by ODD's authors

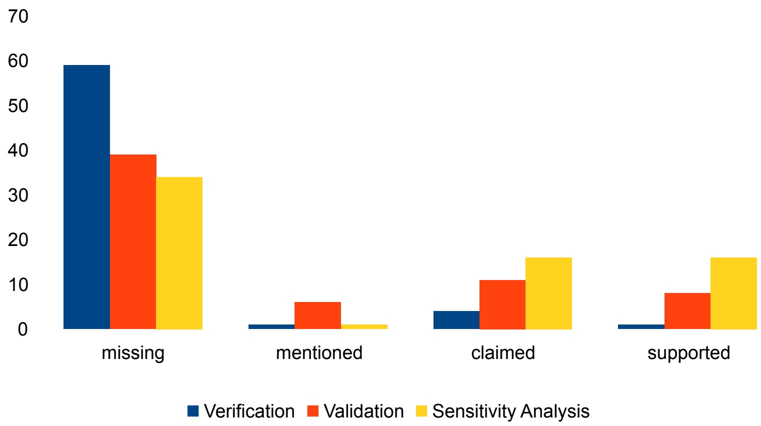


Figure 2.3: Verification, validation and sensitivity analysis, in surveyed papers. Results are listed in Table 2.2

	Verification	**Validation**	**Sensitivity**
missing	59 (90.77%)	39 (60%)	34 (52.30%)
mentioned	1 (1.54%)	6 (9.23%)	1 (1.54%)
claimed	4 (6.15%)	12 (18.56%)	15 (23.08%)
supported	1 (1.54%)	8 (12.31%)	15 (23.08%)

Table 2.2: Verification, validation and sensitivity analysis, in surveyed papers

almost 40 papers (approximately 60%). The number of articles that provide hard evidence about these three topics is alarmingly small.

While verification was claimed in four articles (6%), the survey found only one article that provided evidence in support of such claims. With respect to validation, 12 articles claimed it while 8 provided a minimum amount of evidence. Sensitivity analysis was claimed but not supported in 15 articles, whereas in another 15 it was substantiated.

2.4.3 Explicit Assumptions

The way assumptions were declared and used in a number of 10 papers, chosen arbitrarily, was analysed in more detail.

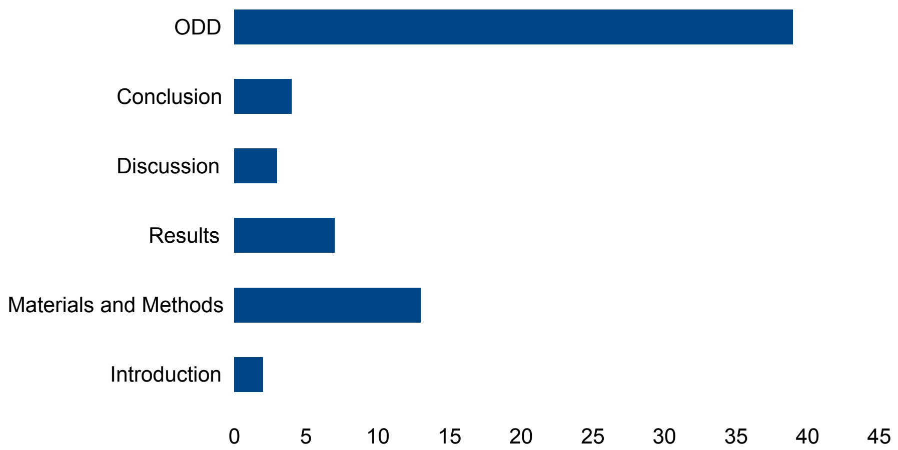


Figure 2.4: Declaration of assumptions throughout article sections

A total number of 73 explicit assumptions was identified, across the 10 papers. Out of the 73, 68 were unique references while 5 times an assumption was repeated. Out of the 10 papers, in only one case was an explicit assumption thoroughly analysed and discussed. Only one explicit assumption was accompanied by justifying information. Most assumptions were of modelling nature (45), followed by assumptions regarding experimentation (9); a number of 5 assumptions were declared as being introduced for simplification purposes, whereas 2 represented parameter values (for which real values were not available).

With respect to the positioning of assumption declarations, the majority were defined in the ODD description of the model (53%); 17% of these unique references occurred in the Materials and Methods section of each paper, following by the Results section with approximately 10%. The following figures expand on these results. Figure 2.4 details this aspect.

2.5 Discussion

This chapter has focused on a subset of SBR articles (the majority published in journals) that already use a documentation standard – the ODD. Following ODD is expected to lead towards obtaining easier to reproduce models, but also more rigorous (since it comes equipped with assurance giving specifications – see Table A.2). Taking into consideration previous ecological surveys that have emphasised the persisting problem of V&V not being widely applied in ecological SBR, this survey aimed if ODD stimulated an improvement in this situation. Additionally, it assessed a smaller number of articles, in terms of their approach towards making assumptions explicit.

ODD is becoming more popular; researchers from more domains are beginning to use it. Part of the authors that proposed it, still use it and propose new developments, five years after its introduction. However, the above results suggest that, even if researchers using ODD have taken a step towards a more rigorous approach to communicating SBR, the majority of efforts fail short of providing sufficient assurance information.

The fact that in so many articles, a direct reference to verification, validation or sensitivity analysis is missing, is indeed a problem. Different from previous surveys, articles were assessed here according to a fuzzy classification. It can be observed, this way, that a larger number of unsupported claims were made about V&V and sensitivity analysis (32 in total) whereas supporting evidence was found in 25 cases.

Verification is the topic that is most often ignored and even when it is claimed (4 times), only once it is accompanied by supporting evidence. Consequently, apart from the focus on validation, it would be useful to investigate why verification receives this little attention.

Validation was completely absent as an explicit term, in more than 60% of the analysed papers. A number of six articles mentioned validation without pursuing it, due to reasons such as insufficient information; in only one of these articles was verification referred to (mentioned, not claimed or supported).

Sensitivity analysis was claimed and substantiated in more articles than validation or verification. This may be partly due to the fact that sensitivity analysis generates results that can be considered contributions (they represent investigations into the unknown behaviour of the model); as such, it can be more rewarding to detail results from various sensitivity analysis experiments, than to expand on how (or if) the simulation has been verified and validated. Another explanation for this situation is that, in performing sensitivity analysis one does not need validation data (that is so hard to obtain in some cases) – the model is explored, rather than tested.

The domain of assumptions can hardly be called under control. There is an average number of 7 references towards assumptions per paper (albeit the variance is high, the maximum being 17, while the minimum being 2); in only one instance, an explicit assumption was mentioned in the Discussion section of the article. Approximately half of the explicit assumptions are declared in the ODD description of the model, however, the remaining half is scattered across the length of the publication.

Although the survey does not expect that the explicit assumptions identified are the only one in the assessed articles (only the simplest of models could be composed of 2 assumptions), the focus here was on explicit, recognisable assumptions. The easier it is to identify a piece of information as an assumption, the higher the awareness of the reader, reviewer and author. By moving assumptions, from the implicit domain towards the explicit one, SBR can approach what in SCS is called assumptions management [81].

Chapter 3

Safety-Critical Systems and Structured Argumentation

This chapter introduces the domain of Safety-Critical Systems (SCS), that represents one of pillars of this thesis. SCS deals with the engineering and certification of systems where failure equates with disastrous consequences e.g. loss of life, environmental damage. An important interdisciplinary potential exists between SBR and SCS; SBR can be augmented by the rigour-sustaining tools and techniques developed in SCS, while SCS can benefit from the modelling and simulation literature generated in so many disciplines employing SBR. The chapter aims to link the two domains by focusing on structured argumentation (SA), an approach developed in SCS but, from the author's perspective, evermore relevant to SBR.

The chapter proceeds as follows: Section 3.1 defines relevant domain concepts, Section 3.2 describes what a safety case represents, while Section 3.3 expands on the relevant SCS concepts of claims, evidence and arguments. Section 3.4 introduces structured argumentation, describing two examples of structuring notations (GSN and Debategraph) and their relation. The chapter is summarised in Section 3.5. Note that the chapter makes extensive use of Kelly's seminal work [54] and the GSN standard [1].

3.1 Domain Concepts

The domain of SCS has its own ontology and, before addressing the main topic of structured argumentation, it is useful to define its core domain concepts.

A *system* can be defined as "a set of things working together as parts of a mechanism or an interconnecting network; a complex whole" [10]. The fabric of a system is not particularly important: in the context of SCS, safety applies equally to software applications and to systems of equations and procedures that keep a satellite on the right orbit .

Safety represents "the condition of being protected from or unlikely to cause danger, risk, or injury" [8]. As a property (modifier), it denotes something "designed to prevent injury or damage" e.g. a safety helmet. The probabilistic nature of the above definition reflects the limits with which safety can be established: what separates safety from the complementary "danger, risk or injury" is unavoidably contextual.

The adjective *critical* is defined as "having a decisive or crucial importance in the success or failure of something" [3]. In SCS, success is related to the preservation of safety. Failure points towards two further concepts: *hazard* and *risk*. A hazard denotes "a potential source of danger" [6], whereas risk implies "the possibility that something unpleasant or unwelcome will happen" [7]. Hazards need to be mitigated in order to minimise their associated risks.

From the above definitions it results that, for a system to be *safety-critical*, it must be "designed or needing to be fail-safe for safety purposes" [9].

3.2 The Safety Case

An important concept in SCS is that of a *safety case*. In Kelly's words, a safety case communicates "a clear, comprehensive and defensible argument that a system is acceptably safe to operate in a particular context" [54, p.22].

An engineered system that is deemed unsafe should not be used, nor enter mass production. Consequently, the stakeholders behind the development of a safety-critical system must motivate why their product is safe and they must show how their claim for safety is adequately supported by evidence – this is the basis for *certification*.

The safety case (also known as safety case report) represents the means through which stakeholders substantiate their safety claim. As such, the safety case delivers an *argument* (of a particular quality e.g. clear, comprehensive, defensible) that shows how its authors have concluded that the "system is acceptably safe from the evidence available" [54, p.22]. Each word of the above definition must be taken into account. The attribute *clear*, for example, points towards the subjective dimension of a safety case (and its safety argument): information itself is useful provided it is communicated in a manner that facilitates understanding and evaluation. Any argument must also be placed within a *context*; absolute safety is unattainable, hence an argument can only aim for the *acceptably* safe. Finally, the safety argument should be *defensible* i.e. justifiable (in both structure and content); the more *comprehensive* the argument is, the more ground it has for being defended.

A safety case generally consists of three main types of components: a) safety *requirements and objectives*, b) safety *evidence*, and c) a safety *argument* that bridges evidence and requirements (see Figure 3.1). The safety requirements and objectives dictate the type of safety evidence that must be collected; the safety argument transforms the implicit (and incomplete) link between evidence, requirements and objectives, into an explicit and justifiable one.

The safety case represents an unavoidable middle layer in an acceptance process (see Figure 3.2). Consequently, it represents a "single point of failure". In fact, the image depicted by SCS is paradoxical to an extent, and this also due to the high level of effort required in addressing safety : on one hand, there is the engineering of highly complicated systems; on the other hand, safety cases

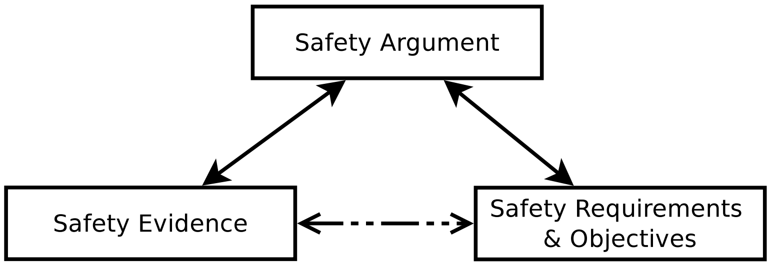


Figure 3.1: Safety case components (derived from [54, p.25])

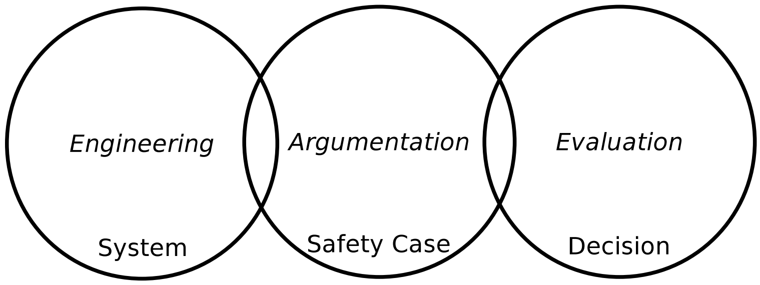


Figure 3.2: Phases and deliverables in a SCS life-cycle (abstract view)

communicated in a subjectively argumentative manner, undergoing a subjective, (albeit expert) evaluation; the process concludes with a binary decision (system acceptably safe or not) that may bare important consequences. Making good use of this unavoidable subjectivity, implicitly added by the human factor, is essential. Here is where the quality of a safety case must be addressed.

The quality of a safety case is directly dependant on the quality of the available evidence, the clarity of its safety requirements and objectives and that of the safety argument.

3.3 Claims, Evidence and Arguments

A *claim* represents "an assertion that something is true"; to claim something means to "state or assert that something is the case, typically without providing evidence or proof" [2]. Consequently, in its purest form, a claim is simply an assertion, unsupported by evidence; to accept it represents an act of *faith*, requiring the existence of a belief in its truthfulness (having confidence in something that is not proved certain). The opposite of an unsupported assertion is a *fact* ("a thing that is known or proved to be true" [5]).

The road from claims to facts is necessarily paved with elements that substantiate the claim's veracity e.g. *evidence*, *justifications* (see Figure 3.3). Evidence is composed of facts (or information) that indicate "whether a belief or proposition is true or valid" [4]. Justifications "show" why claims are right or reasonable.

The relation between a claim and the elements that support it may, however, not be unequivocal. This is why, as a research topic, the organisation and use of safety evidence has received attention in SCS. Kelly and Hawkings [53] expand on this topic, emphasising why evidence may not be sufficient in supporting a claim by itself. Firstly, one has to assess the capability of the evidence type e.g. derived from testing, analysis, reviewing or field experience, and establish if a particular type of capability is sufficient in supporting a type of assurance claim (e.g. a software safety assurance claim). Secondly, the capability of the evidence instance, to support an assurance claim, needs to be evaluated; it is possible that software testing provides adequate evidence in supporting a claim, but a particular test may be inadequate as it may be irrelevant (e.g. if applied to a different version of the software), be executed in a different context or under a different set of assumptions, it may not cover all the required properties or may not contain information to the level of detail required. Finally, but equally important, authors question the trustworthiness of evidence.

In front of the *unknown* or *uncertain*, the acceptance of a partially supported claim represents an act of trust in the *assurance* achieved through the supporting evidence. Assurance is another important (and problematic) concept in SCS.

In addressing the above topics, arguments may be used. A safety argument must "demonstrate that sufficient assurance in the safety" of the safety-critical system has been achieved [53]. The safety of a safety-critical system can be considered an ideal, ideal that cannot be absolutely or objectively established (hence dotted line in the figure). As shown in Figure 3.1, generating safety

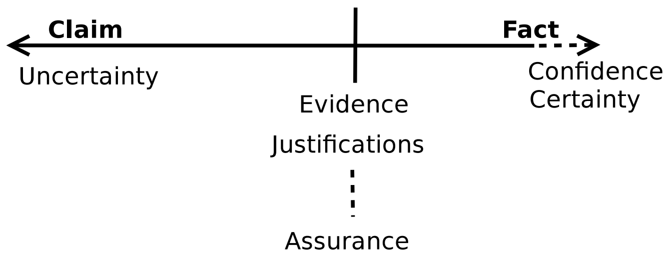


Figure 3.3: Relation between claims, evidence and facts

evidence is important, but in order to support a claim of safety, a safety argument is constructed. Building blocks can be other (sub)claims, evidence, justifications.

3.4 Structured Argumentation

The expression of arguments can be achieved both through free-form text, and structured approaches. The latter approach has received considerable attention and different methods have been proposed, especially in the domain of SCS e.g. tabular structures, claim structures, traceability matrices, Bayesian belief networks [54]. Other submissions argument interchange format (AIF) [70] or risk-based argumentation [34].

The chapter proceeds with an introduction of two notations for structuring arguments – the Goal Structuring Notation (GSN) and the Debategraph notation.

3.4.1 The Goal Structuring Notation (GSN)

The Goal Structuring Notation (GSN) [1] is a graphical notation for presenting structured arguments, that has been introduced and developed initially on the EPSRC-funded ASAM-II project. Building on the Toulmin form for structuring arguments [85] (i.e. in terms of claims, warrants, backing, rebuttal etc.), GSN differed by making supporting evidence a first class element. Having established the basic GSN, the ASAM II efforts were continued by Kelly [54] in order to

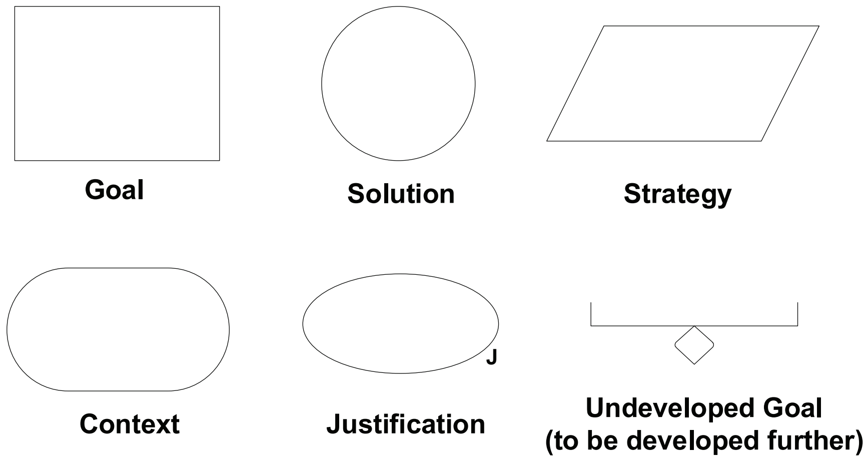


Figure 3.4: Basic GSN notation

provide methods for the construction of goal structures, to clarify the semantics of elements of the notation, and address deficiencies in GSNs expressive power.

Features: visible decomposition of the chain or reasoning; supporting evidence represented explicitly in the argument; claims are contextualised.

The main benefit of GSN is claimed to be fact that it improves "comprehension amongst key stakeholders", which in turn is expected to "improve the quality of debate" and also shorten the time needed for reaching agreement on the argument [1, p.6].

3.4.1.1 GSN Elements and Relation Types

Goal structures/hierarchies are composed of elements e.g. goals, strategies, solutions, contexts, assumptions, justifications (see Figure 3.4), linked through "SupportedBy" or "InContextOf" relations. Elements have a minimum of two boolean properties: developed and instantiated.

Goal The *goal* element represents a claim, expressed in a noun-phrase + verb-phrase formula.

Strategy The *strategy* element declares the reasoning step when decomposing a goal into subgoals.

Context The *context* element represents a reference to contextual information or a statement. It describes the context in which the goal or strategy to which it is linked should be interpreted. The *context* defines or constrains the scope of a claim, and is considered an assertion for the whole argument strand supporting the claim. “Claims can only be asserted to be true in a specified context” [1, p.12] as there is no ‘universal validity’ [1, p.28].

Assumption The *assumption* element contains an “intentionally unsubstantiated statement” which needs to “hold valid” for the interpretation of the goal or strategy to which it is linked.

Justification The *justification* is a “statement of rationale”, explaining why authors consider the linked claim or strategy acceptable. The *justification* applies only to the element to which it is linked.

Solution The *solution* element contains a reference to a piece of evidence, supporting the “truth” of a claim [1].

3.4.1.2 Constructing GSN arguments

The GSN Community Standard, version 1 [1], details two approaches to constructing GSN (safety) arguments: Kelly’s six step top-down approach [54], and a complementary bottom-up approach. Apart from specifying how a GSN argument can be built, these two methods also suggest how we can evaluate the quality of a GSN argument. The GSN standard stipulates as objectives for this notation: argument clarity, comprehensibility and veracity [1, p.25]; the hierarchical, annotated and evidence-based nature of a GSN argument should substantiate these three desired traits.

In addition to the above mentioned methods, Hawkins et al. [46] propose "a clearer way to constructing safety arguments", oriented on separating the core (safety) argument from the assurance arguments required in supporting the inferential and evidential relationships of the (safety) argument elements. This method also requires an extension of GSN beyond its standard [1]; for the purpose of this thesis, we focus only on the six-step method, as this corresponds to the standard GSN which is supported by the argumentation tools (editors) at our disposal.

3.4.2 The Debategraph Notation

Debategraph[1] is a web-based system providing functionality for recording debates. Equipped with a visual interface enabling the browsing and collaborative editing of debates, Debategraph comes with its own debate-mapping notation. This notation is relevant for our purposes as it has been shown to capture debates from many domains, including scientific ones.

The notation currently supports 15 types of elements and 13 types of relations[2]. The elements describe the debate content e.g. through *positions* or *supportive arguments*, but also include references to the *protagonists*[3] involved, the *decisions* taken in response to debates *issues* or *tasks* that should be addressed by the debate's participants. For the purpose of this thesis, we focus only on the elements that describe the debate content:

[1] www.debategraph.org
[2] Inter-element relations are called cross-links in Debategraph
[3] Significant actors in a debate, that have recorded arguments.

- *Map*: starting point for a debate; a *map* will contain one or more *issues*;

- *Issue*: a topic (e.g. question) considered by the debate (map);

- *Position*: an answer or option concerning a raised *issue*. Complex *positions* are broken up in *components*;

- *Supporting (Opposing) argument*: an argument supporting (opposing respectively) a *position* or another *argument*;

- *Argument group*: a set of arguments that are net supportive or opposing;

- *Part-argument*: a co-premise that can be coupled with other co-premises in order to support an argument or conclusion.

The set of relations between Debategraph element is rich. We focus again on those types of relations that are limited strictly to describing the debate content:

- *Causation*, *Consistency* and *Inconsistency*: reflects a causal link between two elements, two elements are consistent/inconsistent respectively;

- *Equivalence*, *Explanation*: the related elements are equivalent, an element explains another element respectively;

- *Relevance*: a generic way to express that one element is relevant (in an unspecified way) to another;

- *Variation*: points towards the fact that one element is a variation of another element.

To differentiate between the many types of elements and relations, Debategraph uses both shape and colour. Elements are represented as “bubbles” (circles of different sizes, with a (limited in length) text label overlaying them); colour is used to differentiate between the existing types of elements. Relations are displayed as simple (one or two-headed) arrows, differentiation taking place also through the use of different colours.

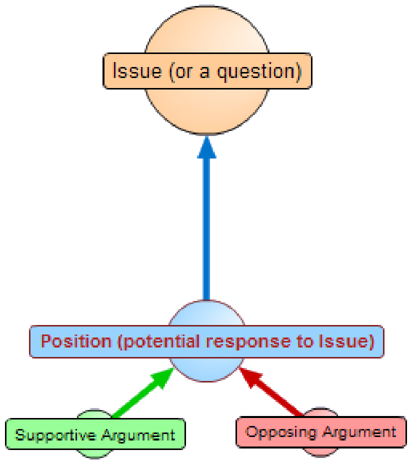


Figure 3.5: Debategraph notation example (bubbles format)

3.4.3 Relating Debategraph to GSN

Table 3.1 shows a parallel between Debategraph and GSN elements. It can be seen that, in order to express some Debategraph elements and relations, with the standard GSN, a degree of "engineering" is required. For example, through GSN we cannot directly place an argument within a specific a specific domain or topic (i.e. the *map* and *issue* Debategraph elements), but we need to create a *goal* claiming that a different element (*goal* or goal structure) is relevant to a domain or topic.

In terms of relations, Debategraph's abundance can mainly be accommodated by the standard GSN through a *goal* claiming the relation type holds e.g. *goal* claiming that consistency of two other *goals*.

While Debategraph is richer in terms of the relations between elements, GSN is already a standard [1] applied to no less than SCS. For this reason, and due to the better support for expressing evidence in GSN, we make use of it for the reminder of the thesis.

Debategraph	**GSN**
Elements	
map	*goal* claiming relevance of a different element, towards a topic (map) referenced in a *context* element
issue	the same solution as above
position	top *goal*
component	*goal*
supportive / opposing argument	a *module* containing an entire goal structure
argument group	*goal* claiming the existence of a group and referencing out towards each *module*
part argument	a public *goal*, goal structure or *module*
Relations	
causation	a *goal* claiming the existence of a causal link between two elements defined, if necessary, in separate *contexts*[1]
consistency, inconsistency, relevance	a *goal* claiming the consistency (respectively inconsistency, relevance) of two elements defined, if necessary, in separate *contexts*
explanation, equivalence	a *strategy* can be used as intermediary between the *explicandum* and *explanandum*. Alternatively, the same solution as above (*goal* claiming explanation exists and *context* referring to it)

Table 3.1: Parallel between Debategraph and GSN elements

3.5 Summary

This chapter has focused on three main concepts: SCS, SA and GSN. We have shown why the domain is SCS is relevant in the context of this thesis, when considering SBR. We do not intend to deploy a full-fledged SA, but to customize GSN for SBR purposes: we need to understand how useful is GSN, in its current format; to what extent it is readily usable.

Chapter 4

Case Study

Previous chapters have introduced the domain of this research (SBR), the problem that we are addressing (accessibility and availability of arguments in SBR) and our thesis (SA is needed both in conducting rigorous SBR and in communicating adequately its outcomes). For the remainder of this thesis we will be using a case study – a published piece of SBR – to address the topics of argument accessibility and GSN's capability of capturing argument threads (Chapter 5), GSN's usage during research (Chapter 6) and assessing the uncertainty of simulation models in comparison with the scientific claims derived from them (Chapter 7).

The case study that we are approaching has been selected based on a series of criteria:

1. it has been published in a reputable journal, consequently it has passed through a series of thorough reviews;

2. the publication has not been contested and has not been shown as being flawed;

3. the study exemplifies SBR (Section 1.1) through the use of a domain-grounded, validated IBM;

4. the simulator's source-code is accessible, upon request; its main author collab-

orated with us on this research;

5. the study inspires confidence [1].

The chapter proceeds as follows: Section 4.1 provides a brief introduction into the ecological background of the case study. Section 4.2 describes the plant ecology case study addressed throughout the remainder of this thesis.

4.1 Ecological Background

Ecology is concerned with the study of interactions among organisms and their environment. Topics of interest include diversity, competition and facilitation among organisms and within ecosystems, number and distribution of populations.

4.1.1 The “Holy Grail” of Ecology

Ecology also has a so called “holy grail”; coined in 2002 by Lavorel and Garnier [57], this highly sought objective consists of uncovering the link between environmental conditions, species traits and community composition. In other words, understanding how environmental conditions will eventually affect the distribution and abundance of organisms, in accordance with their specific traits.

This objective is both important and hard to achieve as it spans multiple scales and focuses on features of different nature. Research addressing the “holy grail” involves sub-fields such as soil science, ecophysiology, population and community ecology. Depending on the scale of the research, other ecological disciplines may be involved, e.g. ecosystem ecology, system ecology. Due to this abundance of domains and information, SBR is becoming an evermore popular approach to addressing the “holy grail” of ecology.

[1] It has been validated, the model and experiments have been thoroughly described so they can be reproduced.

One necessity along the path of studying this objective, is that of taking into account intraspecific variation.

4.1.2 Intraspecific Variation

Intraspecific variation refers to the variation of traits amongst the individuals of a species, rather than between those of different species. In more detail, a species can be described in terms of plant traits (e.g. required uptake, death probability). Plant traits are the bases of differentiation between species: two species will have different values for a set of traits, situation also known as [interspecific variation]. If the values of traits vary amongst the individuals of a single species, this is called intraspecific variation.

Many previous research efforts, especially using EBM, have considered individual variation as not being significant, when looking at larger scale communities. Thus, mathematical or matrix life-cycle models have implied populations are formed of identical individuals, differences appearing only from species to species. Such models have either neglected intraspecific variation, haven't simulated at population level or have had too strong assumptions.

ABM and SBR are modern prerequisites for studying intraspecific variation.

4.1.3 Plant competition

Competition arises naturally in a system that has bounded resources and two or more individuals requiring the same types of resources. If each individual would have its separate, unique niche, competition would not exist or would have minimal effects.

Competition between plants usually occurs with respect to resources (e.g. water, nutrients, light) and space. In terms of effects, competition drives species extinction or succession, mapping plant traits to environmental features. Competition leads to the "survival of the fittest".

In ecology, plant competition has been studied for a long period of time, both experimentally and theoretically, more recently scientists adding plant facilitation as a complementary and equally important process [26].

4.2 The Case Study

The case study used in this research is a plant ecology study, published in 2007 in the Journal of Ecological Modelling. Authored by James Bown and collaborators from the SIMBIOS Centre[1], Macaulay Institute[2] and Scottish Crop Research Institute[3], the study was published in a series of two articles, [23] introducing an individual-based model and demonstrating its relevance to ecological research, whereas [64] made use of the model for generating new ecological results. For the purpose of this thesis, we focus on the first paper [23].

4.2.1 The Model

The plant ecology model proposed in [23] is evolved from Bown's PhD work [22], and was initially published in [65]. Adopting a widely used 2D grid spatial layout and running in discrete time, the model enables the study of plant diversity throughout space and time.

Purpose The purpose of the model is to allow the study of the contribution of variation among individuals, to the functioning of ecosystems. "To understand the structure, dynamics and function of communities within an environment" [23], by creating a model that combines intraspecific and interspecific variation.

[1] SIMBIOS Centre, University of Abertay Dundee, Bell Street, Dundee DD1 1HG, UK
[2] Macaulay Institute, Craigiebuckler, Aberdeen AB15 8QH
[3] Invergowrie, Dundee DD2 1RJ

The main objective sought in [23] was to showcase the importance of accounting for individual variation (also known as "intraspecific variation"); contrary to the widespread use of mathematical models which treated individuals as homogeneous masses, Bown et al. [23] provided a validated, biologically-grounded example of IBM which did not only reproduce familiar ecological patterns, but it suggested explanatory insights into the relation between structure and function, as mediated by trait trade-offs.

Entities Two types of entities are represented: plant individuals and spatial cells.

State Variables Plants are defined in terms of a genotype (defined by 12 directly-parameterizable traits) and a phenotype reflecting the state of the plant (see Table 4.1). A spatial cell is described by its resource substrate; the corresponding state variables are listed in Table 4.2.

Scales In terms of organisational scales, the model sees the development of populations of plants sharing the same genotype (clones). In terms of time scales, a time step represents one day in the real world, and experiments run for 50000 time steps. The environment is formed of spatial cells that correspond to a surface of 10×10 cm^2 in reality; the studied spatial scale varies between 10×10 to 50×50 cells wide (1 to 25 m^2).

Process overview and scheduling Each time step, plants execute, sequentially, the following processes: resource uptake, resource allocation, development or reproduction. The model is resource-driven: a single, abstract resource that fills the environment and replenishes the grid cells at the end of each time step. Figure 4.1 contains a state diagram depicting the scheduling of the plant and environment processes.

Initially plants uptake resources from the environment, the acquired quantity

No.	State variable	Details
Genotype		
1	essential uptake	$U_e(s)$
2	requested/essential uptake ratio	$U_r(s)$
3	spatial distribution of uptake	$D_u(s)$
4	compartment partition	$P_s = 0.8$
5	structural store release rate	$r_t \in [0.28, 0.143]$
6	surplus store release rate	$r_u \in [0.140, 0.153]$
7	time dependent fecundity	R_t
8	development dependent fecundity	$R_d(s)$
9	storage/fecundity relation	R_f
10	seed dispersal pattern	Dp
11	resource shortfall threshold	$V_t = 0.1$
12	resource shortfall period	$V_p = 5$ days
13	plant death probability	$P_d = 0.001$
Phenotype		
1	age	≥ 0
2	development stage	$s \in [0, 50]$
3	structural compartment	$S_s \geq 0$
4	structural store	$S_t \geq 0$
5	surplus store	$S_u \geq 0$

Table 4.1: Plant state variables

No.	State variable	Details
1	resource intake rate	$R_{i.r} = 3$
2	resource release rate	$R_{r.r} = 3$
3	initial resource level	$R_{i.l} = 3$
4	resource saturation level	$R_{s.l} = 3$
5	resource level	$r \in [0, R_{s.l}]$

Table 4.2: Resource substrate state variables

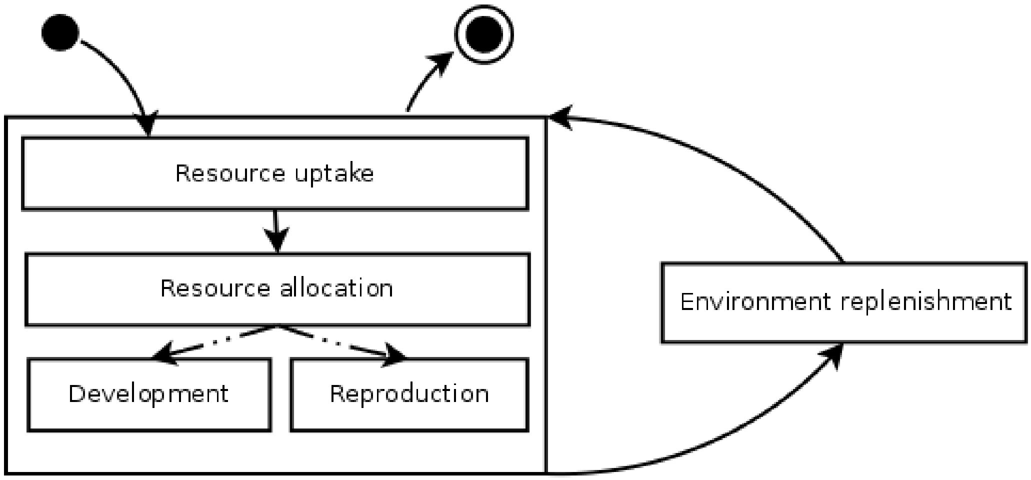


Figure 4.1: Scheduling of plant processes (state diagram)

being a trade-off between the requested amount and the availability of each environment cell. Resources are then distributed amongst the three resource compartments of the plant. If a plant suffered from insufficient resources over a specified time interval, it will die. Development is a consequence of resource levels in the general store exceeding a certain threshold, while reproduction may be time or resource mediated. Plants may also die at the end of a time step, according to the P_d probability (trait).

Emergence Population dynamics and competition emerge from plant-to-environment interactions. Plants compete for space and for resources; competition for space results from the fact that grid cells that cannot be occupied by more than one plant, operate on a first-come first-served basis, whereas fecund plants disperse their seeds over their surroundings; competition for resources is caused by the direct uptake of finite resources from the environment.

Interaction Individuals do not interact directly, but mediated through the environment. The plant life-cycle is driven by the availability of resources, and as the environment contains is limited in this sense, competition emerges.

Stochasticity Various individual traits are sampled from normal distributions of species values. Individuals have a random death probability. Plants disperse their seeds in random locations positioned in their vicinity. Stochasticity is also involved in determining the order in which plants disperse their seeds, so that biases are avoided.

Observation For evaluation, the number of distinct genotypes present within the environment and the abundance of each of them are recorded.

Initialization The resource substrate of the environment is initialised homogeneously, with all state variables valued at 3 units. With respect to the individuals, a number of 75 plants is generated initially. Each of these has a unique combination of trait values drawn from the species distributions, while their resource compartments are empty. The values for the plant related state variables are shown in Table 4.1. These are derived from laboratory observations of the Rumex acetosa species.

Initialisation may vary between simulations. A unique random number generator is used, and this can be seeded differently across simulations.

4.2.2 The Experiments

The model proposed in [23] was evaluated based on the genotype-area and genotype-abundance patterns it produced. For the first type of evaluation, experiments were perform to establish the relation between the diversity of genotypes (number of unique genotypes) remaining at the end of a simulation, and the size of the 2D environment. Figure 4.2 shows the genotype-area curve obtained in Bown et al.'s [23] experiments, which is considered as being consistent with standard species-area curves.

In terms of the relative-abundance distribution of genotypes, the model produces a familiar lognormal curve, that is said to be "consistent" with (species) lognormal

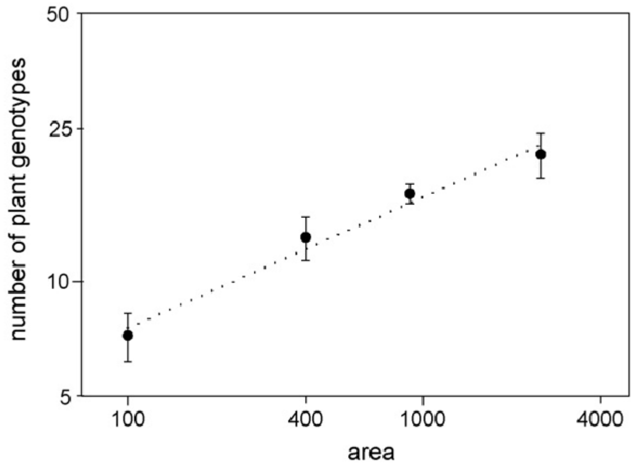


Figure 4.2: Genotype-area curve produced by Bown et al's model (from [23, p.11])

distributions observed for other ecosystems. Since both the genotype-area curve and the relative-abundance distribution are consistent with the literature, the model was considered valid.

Chapter 5

Reverse-Engineering Publications into GSN Arguments

One of the points of interest of this thesis is the communication of SBR. Motivated by what we call SBR's argument problem (Section 1.2.4), we aim to see the ***accessibility*** *and* ***quality*** *of arguments increase* in SBR publications; in particular, we seek to *stimulate the use of SA for the construction of SA* as primary or supplementary sources of information accompanying SBR publications. Towards these objectives, we view the adopting of SA in mainstream SBR as a precursor to a solution.

There exist a number of notations for developing SA (see Section 3.4) but for the purpose of this thesis we focus on GSN – a well-established SA notation in the SCS domain, who's features[1] make it a candidate to structuring SBR arguments. Moving from SCS to SBR, we need to establish if, in what context and to what extent is GSN, in its current format [1], adequate for SBR use.

GSN is traditionally used for expressing safety, security or assurance arguments, that is arguments in favour of a position or claim. Science, however, reflects a

[1]GSN is a generic notation: its elements can contain information pertaining to any domain, including science and SBR.

networking of ideas that goes beyond the strictly positive affirmation of a (system) property; ideas are contrasted, opposed or generally related in more ways than GSN allows for (see Section 3.4.1). Science uses persuasive forms of expression, not only logical ones. Questions arise then about the scope or applicability of GSN within a scientific context: can GSN be used to capture the threads of argument in existing SBR publications? If the standard GSN is not fit for this task, how would we need to alter it in order to succeed? This is the first objective of this chapter.

Some researchers have started to demand a *higher rigour* from SBR publications (see the ODD [40] protocol, addressed in Chapter 2), while the peer-review system is being generally criticised for its deficiencies, mainly for not efficiently filtering out flawed publications[1]. Through rigorous construction of SA, it may be possible to address the two issues described above. Assessing how existing SBR publications map onto GSN is therefore motivated by the potential of identifying GSN as a solution and a means for improvement.

This chapter explores the level of rigour in existing papers, presenting exemplars in detail. We study how easy it is to reverse-engineer the arguments of our case study publication (Chapter 4), assessing the hypothesis that, since scientific articles present peer reviewed, systematic research, the obtained SAs should be clear and sufficiently supported by evidence. The corollary of the hypothesis is that, if GSN is appropriate for capturing structured arguments in SBR publications, then creating GSN diagrams of the arguments should be a seamless mapping process.

In fact, the mapping of a SA in GSN from the SBR publications considered is not seamless. There are two possibilities: that the arguments are not adequately expressed in the publications, or the GSN SA is not appropriate for this purpose. We explore both alternatives.

The chapter proceeds as follows: Section 5.1 motivates the need for clear, evidence-based SBR arguments, while Section 5.2 describes the reverse-engineering ap-

[1] We can argue that peer-review deficiencies are also due to the current plain-text format of publications, which often lack accessible, structured arguments.

proach we developed; Section 5.3 exemplifies the reverse-engineering approach through as applied to our case study.

5.1 The Need for Clear, Evidence-Based SBR Arguments

We have argued that SBR and its publications, in general, manifest a so called argumentation problem (Section 1.2.4). Albeit it can be said that science learns also from mistakes, it is far more important to have solid, sound research, putting forward clear, compelling arguments even in situations when the difficulty (or complexity) of the published research represents a certain challenge.

To discuss about the need for clear, evidence-based arguments requires a set of clarifications. Firstly, structured argumentation has only been used for SBR purposes in a number of publications, first-authored by Fiona Polack or Teodor Ghetiu; currently there is no edifying body of SA research within SBR. Secondly, it is important to note that the *structure* of information is comparably as important as its actual *content*. While this thesis does not delve into the formalism of language formation or related topics e.g. grammar, we emphasise through a simple example that plain text can efficiently present concise, well-structured information (Section 5.1.1), but that this relation does not scale up to research articles (Section 5.1.2).

5.1.1 Clearly Structured Information: A Definition

For this exercise, we choose a succinct piece of information, that is straightforward to structure: a definition is a good enough example. Let us study the definition of an assurance case, a concept relevant to this thesis. The assurance case has been defined as "a reasoned and compelling argument, supported by a body of evidence, that a system, service or organisation will operate as intended for a

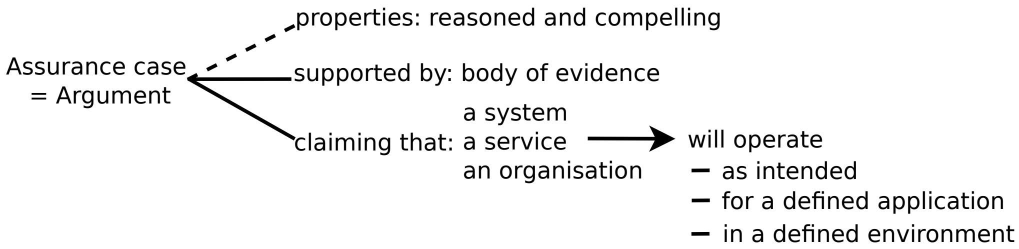


Figure 5.1: Decomposition of assurance case definition

defined application in a defined environment"[1, par 0.2.2].

Assuming we are unfamiliar with the domain of SCS and this definition is providing novel information, how should we understand it, "compile" it? While not seeking a unique answer, we can simply decompose the definition in the manner presented in Figure 5.1, based on the relations between its concepts (and not their accurate semantic). An assurance case is thus a supported argument, of an explicitly defined quality, addressing a contextualised[1] claim of noun – verb nature.

Although the assurance case definition is written in plain text, it reflects a meta-structure, a conceptual organisation[2], which can be said to reflect the structured thinking of its authors[3]. The conclusion is simple and not innovative:

well-structured, concise information can be efficiently presented through plain text[4].

[1]Contextualisation is achieved through the properties underlying the verb 'will operate'
[2]enhanced by parsimony
[3]Expressed also in, and stimulated through their work on structured argumentation.
[4]Poets and writers have often shown the power of words.

5.1.2 Why research articles are more problematic

A definition is a concise piece of information, encapsulating *one meaning.* In the above assurance case definition, *no term could be omitted with harm*:if we leave out the two qualities, the argument might as well be erratic; if we leave out the 'supported by' attribute, it could be understood that an incomplete (unsolved) assurance case would be sufficient for certifying a SCS; if we do not mention what claim the argument addresses or we do not contextualise this claim, then the argument would be ill-scoped, requiring an excessive level of induction. Let us note that the approach for structuring the definition is *ad hoc* (classifying and relating the concepts) and it does not directly use GSN, although the awareness of GSN has helped (especially GSN's *context* element).

If the above conclusion regarding plain text holds in the small, does it hold in the large? If a short message conveying one meaning can be efficiently "serialized", can the efficiency be maintained in the large, when the message spans over the length of an entire research publication and a varied set of positions is expressed? The earlier discussion on ODD and the peer-review system should point towards an optimistic *it depends* answer.

Our view is that the length of the publication only sets the scene, whereas the (difficult) topic of SBR instantiates it (as will be shown in the Chapter 7). A research paper does not have the same aim as that of a definition; while the definition needs to unequivocally clarify the meaning of a concept, the purpose of a research paper is to convey a contribution which is acceptably incomplete[1] with respect to an ideal which may not even be explicit (or scoped). If in a definition, every word, every concept must be accounted for, how different is the situation in a research article! If dictionaries represent sorted collections of unequivocal definitions, then how different are the expectations for scientific journals and proceedings? If one will use a dictionary for clarifying the meaning of a term, how much more varied is the set of purposes for which someone will access a scientific publication!

[1]Knowledge is forever open; we will always have work to do [62].

The length and purpose of a piece of information are important. A research paper cannot fit into a paragraph, in the same manner as a concise definition can. Conversely, once there is sufficient room for a larger, complicated piece of information, we may not see the woods for the trees i.e. the concise, compelling scientific argument (if it exists).

5.2 GSN Reverse-Engineering Method

Knowing that GSN is a SA notation standardised within the SCS domain [1], being used for structuring safety, security and assurance cases of high[1] rigour, it is relevant to see what we can find out about SBR arguments, through GSN's perspective. In other words, how well do SBR arguments fit the GSN format. We also study the use of GSN for developing repositories of scientific arguments, in a similar manner as there currently exist databases of scientific abstracts and publications. Since scientific arguments are traditionally not first-class elements in research publications [2].

In order to address these topics, we need a method to go from the plain text of SBR publications, to the GSN structures: to reverse-engineer the publications into GSN diagrams. The interesting part is that this activity is what any peer-reviewer, formally or informally must do: to identify and evaluate the scientific arguments and evidence put forward by a publication. Since we cannot conceive of a scientific publication that does not consist of at least one argument (its premises being its results, while its conclusions being its claimed contributions), those involved in peer-reviewing should be well-experienced in reverse-engineering and assessing arguments.

[1]Ideally; in reality, there are examples of GSN safety cases that were poorly constructed, leading towards catastrophic failures e.g. [43]

[2]Neighter abstracts nor research papers deliver explicit, structured arguments (on a regular, systematic basis).

5.2.1 Motivation of an Approach

The GSN standard [1] describes two methods for constructing GSN arguments: i) a top-down approach (Kelly's 6-step method [54]) and ii) a bottom-up approach (adaptation of the top-down approach, comprising 7 steps). There is, however, no guidance on how to use GSN for reverse-engineering existing publications into structured arguments. If we were to perform this action without a method, then the procedure could be summarised through a subjective process of reading, 'understanding' the publication and structuring its arguments according to the two approaches.

The outcome of such an ad-hoc approach would be limited in the sense that it would not benefit from the added confidence of having followed a method, nor would there be a guarantee of its repeated success. Let us look in more detail into what using the two argument construction methods, for reverse-engineering GSN arguments, implies.

5.2.1.1 Using The Top-Down GSN Argument Construction Method

Reverse-engineering a publication through the 6-steps method implies:

1. identify *goals*: the **GSN user will decide**[1], upon scanning through the article[2], which are the claims put forward by the publication. Since the formulation of step 1 does not refer to "*top-goals*", the **GSN user will decide** which *goals* to start with (or rather identify *all goals*?). In addition, the **GSN user will decide** upon the most compelling formulation for each *goal*'s statement.

2. define basis of *goals*: The GSN standard requires any unclear term of a *goal* statement to be clarified within *context* elements, apart from the purpose of the statement which will be addressed through the supporting sub-argument e.g. if the *goal* claims that "System X is safe", then "safe" – the purpose of the claim

[1] "identify" [1, Sec. 2.3.2]
[2] Its Abstract, Introduction and Conclusion sections.

– will be defined by the underlying argument [1, par. 2.3.3.5]. The **GSN user will decide** which terms should be defined in *context* elements and which will be addressed through the goal structure.

3. identify *strategy* used to support *goals*: the **GSN user should identify** the reasons that were put forward in favour of each *goal*'s truth (or how the goals were substantiated);

4. define basis for *strategy*: the same applies here as in step 2 (identifying the bases of the *goals*);

5. elaborate *strategy*: the **GSN user should identify** the goal statements that follow from the *strategy*; the **GSN user should repeat** steps 1 – 5 until lowest-level *goals* can be directly supported by evidence;

6. identify *solutions*: the **GSN user should identify** the information that was used as supporting evidence for lowest-level *goals*.

Any realistic reverse-engineering based on using this method requires that the GSN user familiarises itself with the publication (i.e. reads it and tries to understand its essence). The GSN user, familiarised with SCS arguments, might be surprised by the lack of *context* information or strategies; additionally, confusion might be raised by the double meaning of terms such as "assumption" (which can be a modelling assumption or an argument assumption) and "purpose" which can mistaken for a *goal* instead of the *context* of a claim. Table 5.1 lists the type of information that should be associated with each GSN element, and what should be avoided in a SBR context.

5.2.1.2 Using The Bottom-Up GSN Argument Construction Method

Reverse-engineering a publication through the 7-steps, bottom-up method is also possible. This method requires that **the GSN user establishes** the top claim(s), after which **the GSN user should identify** the items of evidence provided by the publication and **establish** what can be claimed for them e.g. in SCS, if

GSN elem.	**SCS**	**SBR**	
goal	claim needing substantiation e.g. "System X is safe"	Yes	(scientific) claim
		No	research purpose
strategy	how to decompose goals e.g. "Argue over all hazards"	Yes	(scientific) reasoning
		No	experimental approach[1]
solution	supporting evidence e.g. "Fault-tree analysis table"	Yes	-=-
		No	algorithm, math. solution
context	definition of terms or reference to information	Yes	-=-
		No	loose uses of "context"
justification	motivation of *goal* or *strategy* adequacy	Yes	explanation or justification
		No	Purpose
assumption	condition for *goal* or *strategy* adequacy	Yes	-=-
		No	modelling assumption

Table 5.1: Relation between GSN elements in SCS use, and what should (not) be used in scientific GSN arguments.

the top claim was that a system was safe for operation, this phase would require "obviously to ascertain what evidence for system safety exists, and precisely what can be claimed for it"[1, par 2.4.2.1]. The remaining 6 steps of the method could be "translated" into what the GSN user **should** or **would**, but it is not the purpose of this thesis to continue this line of work.

Rather than assuming that the GSN user will be able to effectively go through the steps of the top-down or bottom-up method, and produce GSN arguments that adequately reflect the publication's arguments (rather than the GSN user's interpretation of those arguments), we study the use of a brute-force reverse-engineering method. While this still maintains a degree of subjectivity, it has the added benefit that *it will process, methodically, each piece of information within the publication.*

5.2.2 The Reverse-Engineering Method

In order to identify the argument structures as they *emerge* from the plain text document, we investigate a brute-force approach. This is aimed at accounting for each phrase within the document and, even more, for each statement (or GSN element) within each phrase. Analysing publications in a sequential, brute-force manner can provide insights into the scattered nature of scientific arguments (due to the absence of explicit, first class, SA).

The method we propose here is iterative, a process composed of two groups of activities: activities applicable to each section of a publication and activities applicable to the whole publication.

Activities addressing the section level:

1. *segment* the publication into paragraphs and phrases;
2. *segment* the phrases into statements;
3. *classify* each statement;
4. *insert* statements into GSN arguments;
5. *assemble* the GSN arguments;

Cross-section (whole publication) level activities:

6. *substantiate* GSN arguments with *away* elements;
7. *assess* the resulting GSN arguments.

Figure 5.2 depicts a process of applying these GSN reverse-engineering activities. Activities 1 to 5 can be applied to each section of the analysed publication; they could be sequentially applied, from Abstract to Conclusions, the end of this phase marking the passage towards activity 6.

For reasons of efficiency, however, we suggest applying the above activities in a gradual, incremental manner. The GSN user could start by reverse-engineering the Abstract and the Conclusions sections; these initial arguments will guide the rest of the reverse-engineering process e.g. will lead to the reverse-engineering of the whole or part of the Introduction and Results sections. The process would continue and move towards reverse-engineering other sections as needed (in fact,

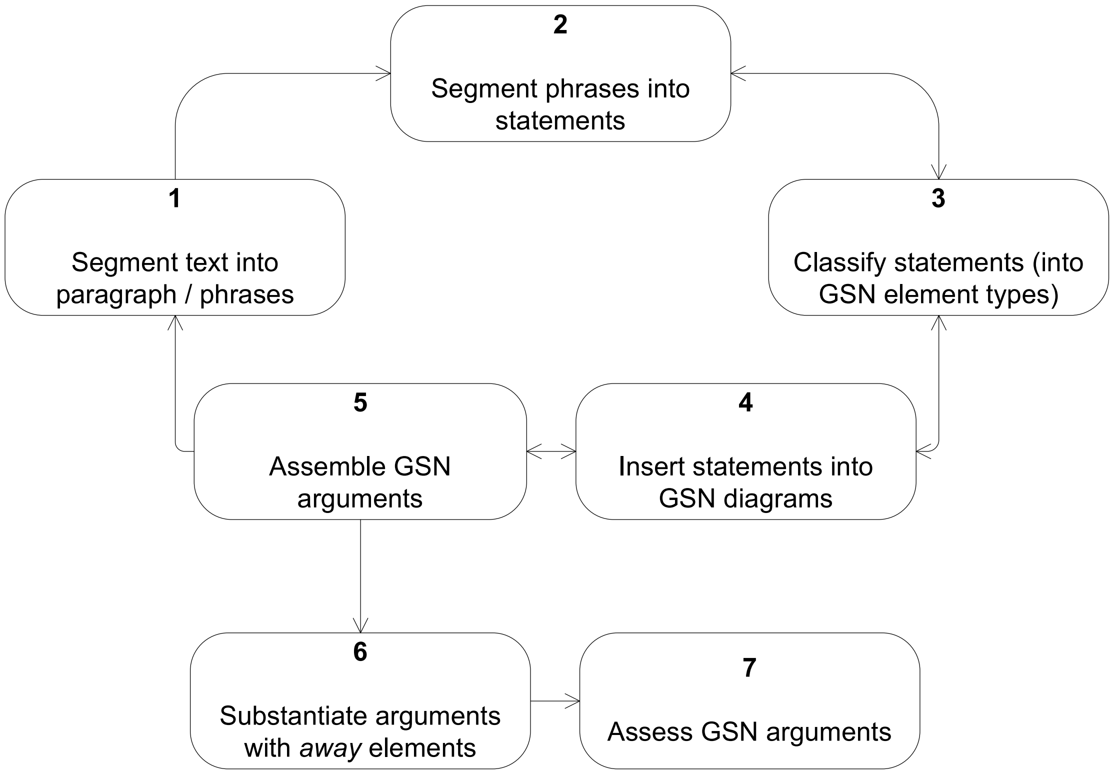


Figure 5.2: Overview of the GSN reverse-engineering method

this gradual approach can be applied to steps 1 to 5 also i.e. only parts of a section being reverse-engineered, as needed). Figure 5.3 reflects this approach.

In detailing each activity, the author turns to the case study and the Introduction section of [23], noting that a more extensive exploration of [23] is provided in Section 5.3.

5.2.2.1 Activity 1: Segment Article into Phrases

The first activity is a necessary must, for this is a bottom-up approach and we want to take into consideration each phrase written in the article. In performing this segmentation, the GSN user does not need to pay attention to the significance of each phrase. We can use tabular structures to facilitate this process: using

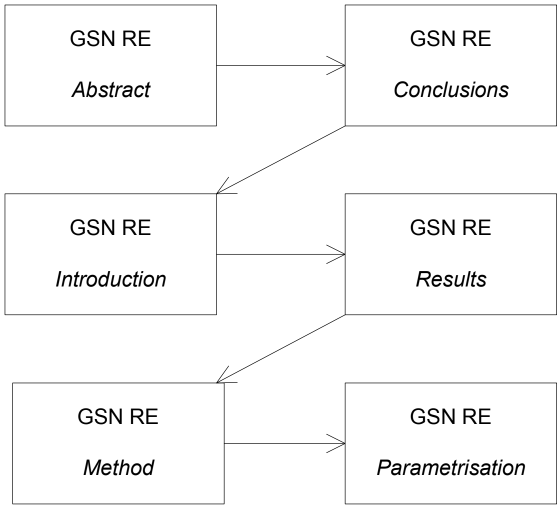


Figure 5.3: Potential approach to gradually applying the GSN reverse-engineering activities

Par.	Phr.	Text
1	1	"There exists a considerable literature [...] (2002)"
1	2	"In spite of [...] (Mouquet et al., 2002)"

Table 5.2: Activity 1 example

a separate table for each article section and inserting each phrase in a new row, recording in addition to the text, its paragraph number and phrase number.

To briefly exemplify this activity, we look at how the first two phrases from [23]'s Introduction can be processed. Table 5.2 depicts the development of the tabular structure. After the completion of this activity, each phrase has a unique identifier, composed of its section, paragraph and phrase number e.g. the first phrase can be identified as phrase 1_1.1.

5.2.2.2 Activity 2: Segment Phrases into Statements

The second activity derives from GSN's rule: a *goal* should only contain a single claim; phrases often encapsulate multiple claims (together with other types of elements). Here we go through each of the table rows (phrases) and identify

Par.	Phr.	Sta.	Text
1	1	a	"There exists a considerable literature based on both the structure and function of biological communities"
1	1	b	"with a particular focus on exploring the link between the two, e.g. [...] Mouquet et al. (2002)"
1	2	a	"In spite of a considerable effort to seek general relations between species richness and function, e.g. productivity"
1	2	b	"no such generalisations have been found"
1	2	c	"and it is now [...] (Mouquet et al., 2002)"

Table 5.3: Activity 2 example

composite statements. We separate these up, inserting the additional statements in subsequent rows and assigning them unique identifiers e.g. Latin characters, as Table 5.3 shows.

In our example, we have separated the first phrase into two statements that constitute separate claims: [23]'s authors first claim the existence of a reality or fact (a "considerable literature" in this case), then they claim the existence of a state or property concerning the previously mentioned fact (in this case, the fact that the "considerable literature" has a particular focus). Here it can be observed that activity 2 is interlinked with activity 3: we need to have a good idea about the kind of content that constitutes a statement (and here GSN's elements are our guideline for what kind of statements to seek), so that we know where to draw the line between a composite phrase and a "simply long" one.

Having mentioned that GSN's element provide a fist guideline in this process, we provide a second aid. In Figure 5.4. we sketch a map of types of relevant information that the GSN user (Me) is expected to find when reverse-engineering a publication. Obtained during the reverse-engineering of our case study (Section 5.3) and of other publications, Figure 5.4 tells that, through the process of reverse-engineering a publication, the GSN user *observes* the content put forward by the publication's authors (Them). By claiming, describing, justifying, etc., this content can refer to the authors themselves, their work, others or others' work, but also to an aspect of reality (fact/state) or theory for which agency has not been specified.

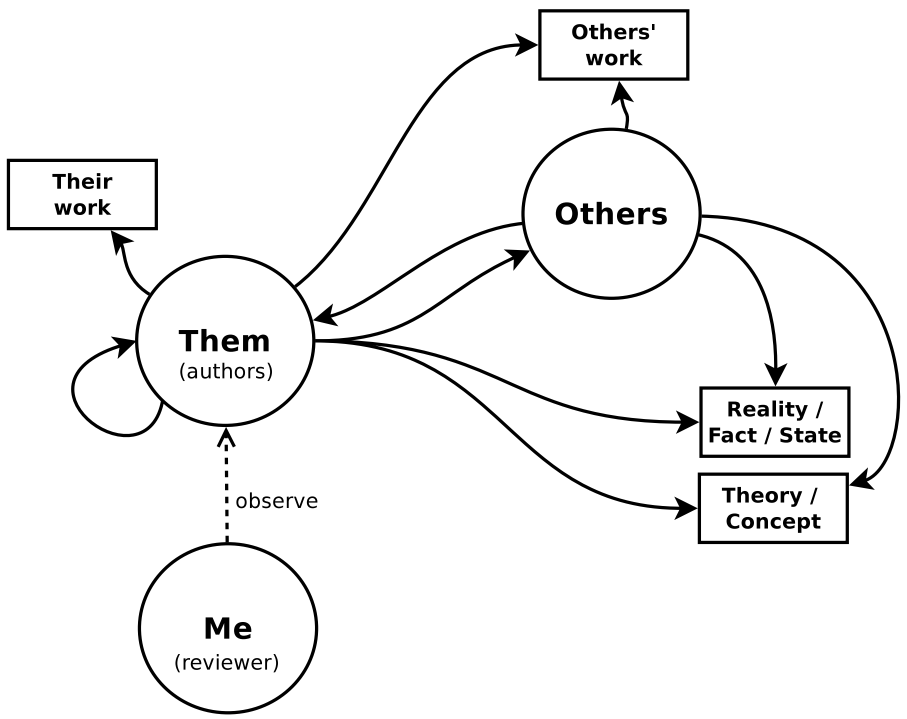


Figure 5.4: Relations between the GSN user (Me) and the observed structure of the publication.

While not suggesting that claims should receive a different treatment, depending on their focus (e.g a reality or a concept), Figure 5.4 should firstly help the GSN user consider all relevant claims (i.e. not overlooking claims about a theory or someone's work, that may seem not important); in addition, this differentiation between claims could help us develop an agency-wise understanding into the claim structure of a SBR publication: what is the percentage of claims concerning the authors' and their work, compared with that referring to others, or to realities and concepts of unspecified agency? Are there radical biases towards one type of claim or another? For each type, what is the percentage of substantiated, justified and contextualised claims?

5.2.2.3 Activity 3: Classify Statements

The objective is to classify statements into the six types of GSN elements (Section 3.4.1.1); as papers are not plain text representations of structured arguments, but combine descriptive passages with reasoning and evidence, this activity carries a degree of subjectivity. There are questions that need to be addressed here: where should the line be drawn between a composite phrase and a single statement phrase? Should a statement be classed in only one category (type of element) or in more? How should be text that does not seem to correspond to any of the six GSN elements (e.g. editorial information) be treated?

In order to provide answers, we return to Figure 5.4. We suggest that the best way to go about this classification process is to pursue it from the GSN user's point of view: we (the GSN user) *will* not be able to exactly capture [23]'s authors' intended argument and trying to do this can actually prove too costly; the alternative is to classify statements according to what we perceive them as being e.g. a claim remains a claim, even if it is supported by evidence; if something seems to be a "factual statement", this still represents a *claim* that can be justified or supported by evidence. We can treat any statement contained in the paper as a *claim*, unless it is obviously a different GSN element. The **GSN user will decide** the type of the statement and the granularity of phrases.

Par.	Phr.	Sta.	Text	Type
1	1	a	"There exists [...] communities"	*claim*
1	1	b	"with [...] Mouquet et al. (2002)"	*claim*
1	2	a	"In spite of [...] productivity"	*claim*
1	2	b	"no such generalisations have been found"	*claim*
1	2	c	"and it is now [...] (Mouquet et al., 2002)"	*claim*

Table 5.4: Activity 3 example, first iteration

Returning to our example, a first iteration of the reverse engineering process could yield the classification shown in Table 5.4. Using the above rule – if a statement is not obviously a different type of element, then it is a *claim*, put forward by the authors (Them) – all the five statements can be classed as claims or GSN *goals*. Another reason motivates the focus on claims: in GSN, *goals* and *strategies* represent the building blocks of hierarchical arguments (the only elements that can be *SupportedBy*).

What we have segmented and classified in Table 5.4 is still too coarse-grained to constitute adequate GSN. Statements 1.1.a could be further decomposed into two separate claims, whereas statement 1.1.b contains additional information ("e.g. Naeem and Li (1997) [...] and Mouquet et al. (2002)") that goes beyond the noun + verb phrase specification of a *goal* element. Similar observations can be made for phrase 1.2 and its statements. A second iteration of this activity should see the additional information being (completely or partially) separated out; activity 4 constitutes another opportunity for further separating statements, crystallising them into concise GSN elements.

5.2.2.4 Activity 4: Insert Statements into GSN Arguments

The fourth activity is open to even more degrees of freedom. The **GSN user should identify** the relation between the statement to be inserted and the developing GSN argument. Since it cannot be assumed[1] that the first statement within a paper is also the (or a) top claim, and that subsequent statements substantiate

[1] Nor expected.

it, this activity should allow the argument structures to *emerge*.

There can be cases when it is not obvious in what type of relation should a new statement be appended to an existing argument. Also, since the publication is composed of sections, we need to decide if the reverse-engineering phase should attempt the construction of one GSN argument per section, multiple arguments per section or should there be other criteria for deciding where to stop an argument and where to start another.

To facilitate the execution of this activity, we use GSN's modular extension [1, Annex B1]. A GSN module encapsulates one argument; through modularisation, argument reuse is enabled, while the visualisation of complicated arguments is simplified. In our case, however, we do not know in advance how many arguments the reverse-engineering process will generate, hence we can start with a module for each section of the publication. Moreover, to reflect the separation of ideas into phrases and paragraphs, we similarly reverse-engineer the publication's content firstly into separate GSN elements, with only obvious *InContextOf* and *SupportedBy* relations between them, then assemble these into more complicated (consolidated) 'paragraph' and 'section' arguments (as presented in Activity 5).

In instantiating GSN elements, it has to be noted that the we are not following all GSN standard specifications. Goals, for example, are filled with more than the *noun phrase + verb phrase* specified by the standard. In order to respect the standard altogether, after classifying the statements (Activity 3) we would need an additional phase of *normalising* the statements, by extracting the exact content appropriate for a GSN element. In a follow-up study, this alternative could be assessed. In this effort, however, the aim is to be 1) as objective as possible by 2) altering as little as possible the original text, so that we will not create the need for further assuring the reader about the adequacy of our 'normalisations' or adaptations. The reader of GSN arguments obtained from the reverse-engineering process should be able to reconstruct the core text of the article from which content is assessed.

Because GSN elements require unique identifiers, we choose to instantiate these with encoded texts reflecting the location of the statement they are encapsulating. In doing so, we use the following encoding:

> ⟨GSN element type⟩ ⟨Section identifier⟩_⟨Paragraph number⟩.⟨Phrase number⟩[⟨Statement identifier⟩]

meaning that inserting the first statement into a GSN diagram will lead to the creation of a *goal* element with identifier G 1_1.1.a and summary containing the statement's text. Figure 5.5 exemplifies the result of a first iteration of this activity.

In Figure 5.5, we have extracted three more elements, in addition to the five claims: two *solutions* (Sn 1_1.1.b and Sn 1_1.2.c) and a *context* (C 1_1.2.a). The information contained in Sn 1_1.1.b substantiates claim G 1_1.1.b, hence they can be connected with a *SupportedBy* relation; in addition, because statement 1_1.2.b contains a claim (G 1_1.1.b) and a single *solution*, both GSN elements can use the '1_1.2.b' identifier. Note that although G 1_1.1.a and G 1_1.1.b are part of the same phrase, these cannot be connected through any of the two relations described by the GSN standard, although the second claim (G 1_1.1.b) makes reference to the subject of the first claim (the "considerable literature"); accordingly, at this stage we can only leave them as separate *goal* elements.

Looking at phrase 1_1.2, we find that the lack of a standard GSN relation between claims, observed in the previous phrase, is repeated. Although the claims refer to the same "effort" and "general relations", they do not substantiate each other[1] hence the only available standard relation between GSN *goals* – the *SupportedBy* relation – cannot be applied. We are left with three separate claims. Table 5.5 presents the result of a second iteration of this activity.

[1] In fact, claim G 1_1.2.a is a condition for stating G 1_1.2.b and G 1_1.2.c and should consequently be defined also as a GSN *assumption*.

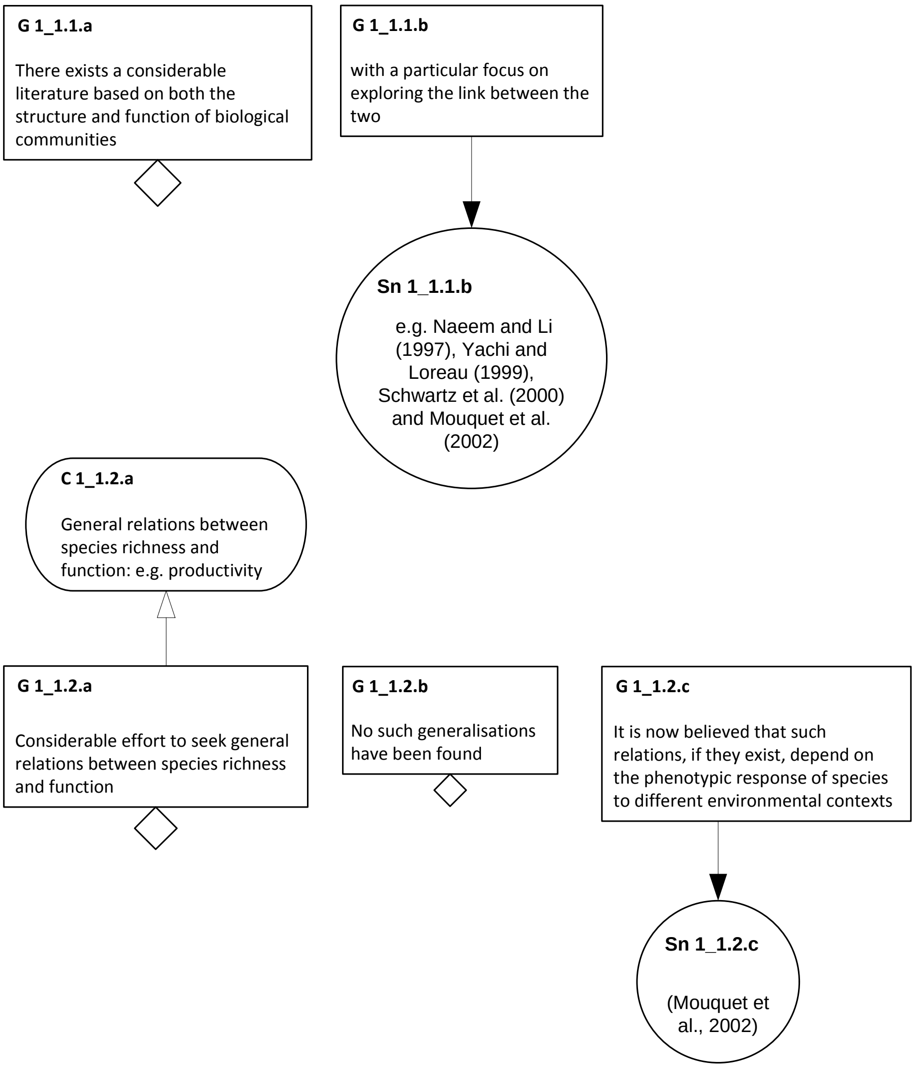


Figure 5.5: Activity 4: converting the first two phrases into GSN elements

Par.	Phr.	Sta.	Text	Type
1	1	a	"There exists [...] communities"	*claim*
1	1	b	"with [...] Mouquet et al. (2002)"	*claim*
1	1	b	"e.g. Naeem [...] Mouquet et al. (2002)"	*solution*
1	2	a	"In spite of [...] function"	*claim*
1	2	a	"e.g. productivity"	*context*
1	2	b	"no such generalisations have been found"	*claim*
1	2	c	"and it is now [...] two "	*claim*
1	2	c	"(Mouquet et al., 2002)"	*solution*

Table 5.5: Activity 4 example, second iteration

5.2.2.5 Activity 5: Assemble Arguments

This is an activity that can be applied on multiple levels; as Figure 5.5 suggests, we need to first assemble 'phrase' arguments, then assemble these into 'paragraph' arguments and, if possible, proceed to higher levels of organisation e.g. 'section' arguments. Consequently, the goal of this activity is that of interlinking the separate GSN *goals* previously inserted in a GSN diagram. If in activity 4 we only used the two standard GSN relations, here we go a step further and propose a series of new relations, represented in Figure 5.6, with the aim of leaving no GSN *goal* unconnected:

- *CoupledWith*: between two GSN elements mapping to two adjacent statements within a phrase e.g. G 1_1.1.a and G 1_1.1.b;

- *FollowsFrom*: between the last (or any) GSN element from one phrase and a GSN element mapping to the first statement within the following phrase e.g. G 1_1.1.b and G 1_1.2.a;

- *InconsistentWith*: between two GSN *goal* elements who's claims point towards an inconsistent (or contradictory) state of facts e.g. G 1_1.2.a and G 1_1.2.b; note that, as in Debategraph (Section 3.4.2), this relation is different from an *OpposedBy*.

- *PartSupportedBy*: between a GSN *goal* and another *goal* or *solution* that are

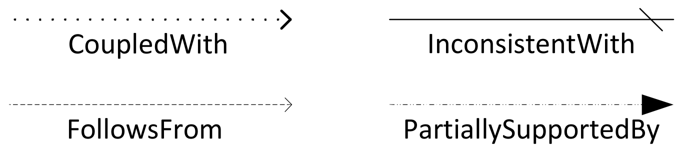


Figure 5.6: New types of GSN relations

believed by the GSN user to provide a degree of support, the relation being less compelling than the *SupportedBy*. We suggest leaving *goals* supported through this type of relation, in an undeveloped state, to further emphasise that it is not sufficient.

Note that *CoupledWith* and *FollowsFrom* have a narrative rather than logic purpose: they record how the text was written. In a different context, we could imagine these being replaced by *ConsistentWith* and *(Part)SupportedBy*, however, for the purpose of reverse-engineering plain text, the role of these two relations is limited. At the other end, *InconsistentWith* and *PartSupportedBy* have clear connotations.

Figure 5.7 depicts one possible result of activity 5. The non-standard GSN argument presented here exposes the difficulty of reverse-engineering an SBR publication into GSN arguments, through a brute-force approach. Note that the *InconsistentWith* relation can be formulated between *goals* at any hierarchical level within a GSN diagram; the vertical arrangement of *goals* G 1_1.2.b and G 1_1.2.c above G 1_1.2.a is only due to spatial constraints, and should not be understood as reflecting a hierarchy in the manner of *SupportedBy* relations.

Strategies can be employed for further assembling arguments, for example *SupportedBy* relations can be used between *goals* that state the existence of a reality or concept, and *goals* that claim a property of this reality or concept. Figure 5.9 exemplifies this strategy on the first three sentences of [23]'s introduction; here G 1_1.1.b is *SupportedBy* G1_1.1.a because first the "considerable literature" must exist, and then something can be claimed about it; similarly for G 1_1.3 being

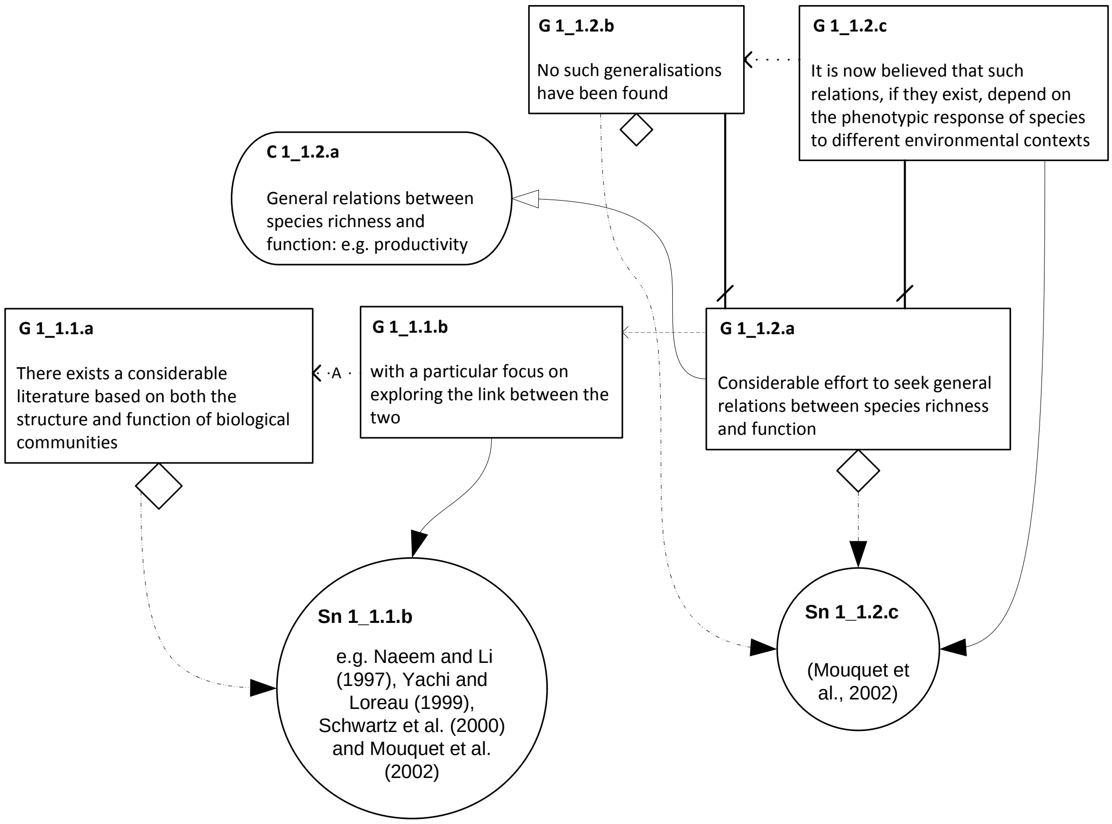


Figure 5.7: Activity 5: connecting GSN elements into a ‘paragraph’ argument

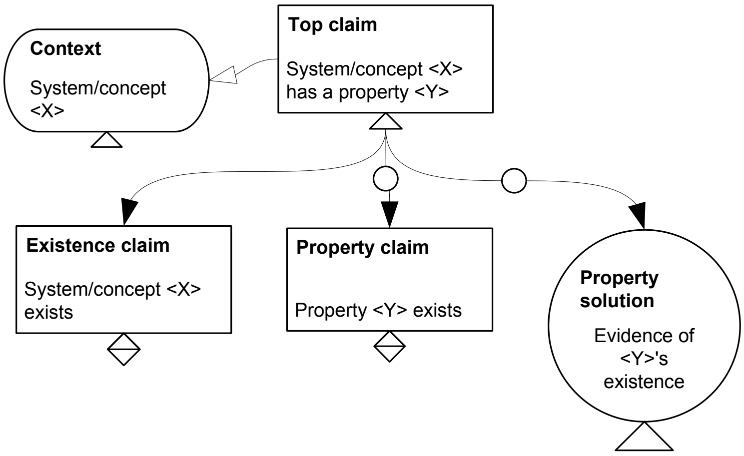


Figure 5.8: Property existence GSN pattern

supported by G 1_1.2.b and G 1_1.2.c.

Going further, we can extract a GSN pattern that reflects the above strategy. Figure 5.8 exposes what we call the Property existence pattern. Note that the two *SupportedBy* relations marked with an empty circle represent optional links [1, Appendix 1]. The Existence claim and Property claim *goals* constitute the starting point for potential sub-arguments. Finally, the GSN elements need to be instantiated (the abstract entities ⟨X⟩ and ⟨Y⟩ need to be replaced with references towards concrete systems, concepts or properties), as emphasised by the triangle shape underlying each of them.

At the end of this activity, it is recommended that all top *goals* in a paragraph argument are aligned horizontally; all top *goals* should be equally important.

5.2.2.6 Activity 6: Substantiate GSN elements with their *away* correspondents

This aim of this activity is to interconnect the 'paragraph' arguments obtained in the previous activity; for this there are two possibilities: replacing or substantiating as many elements as possible, with their *away* correspondents. We opt for the latter option, so that we preserve the original sequence of GSN elements.

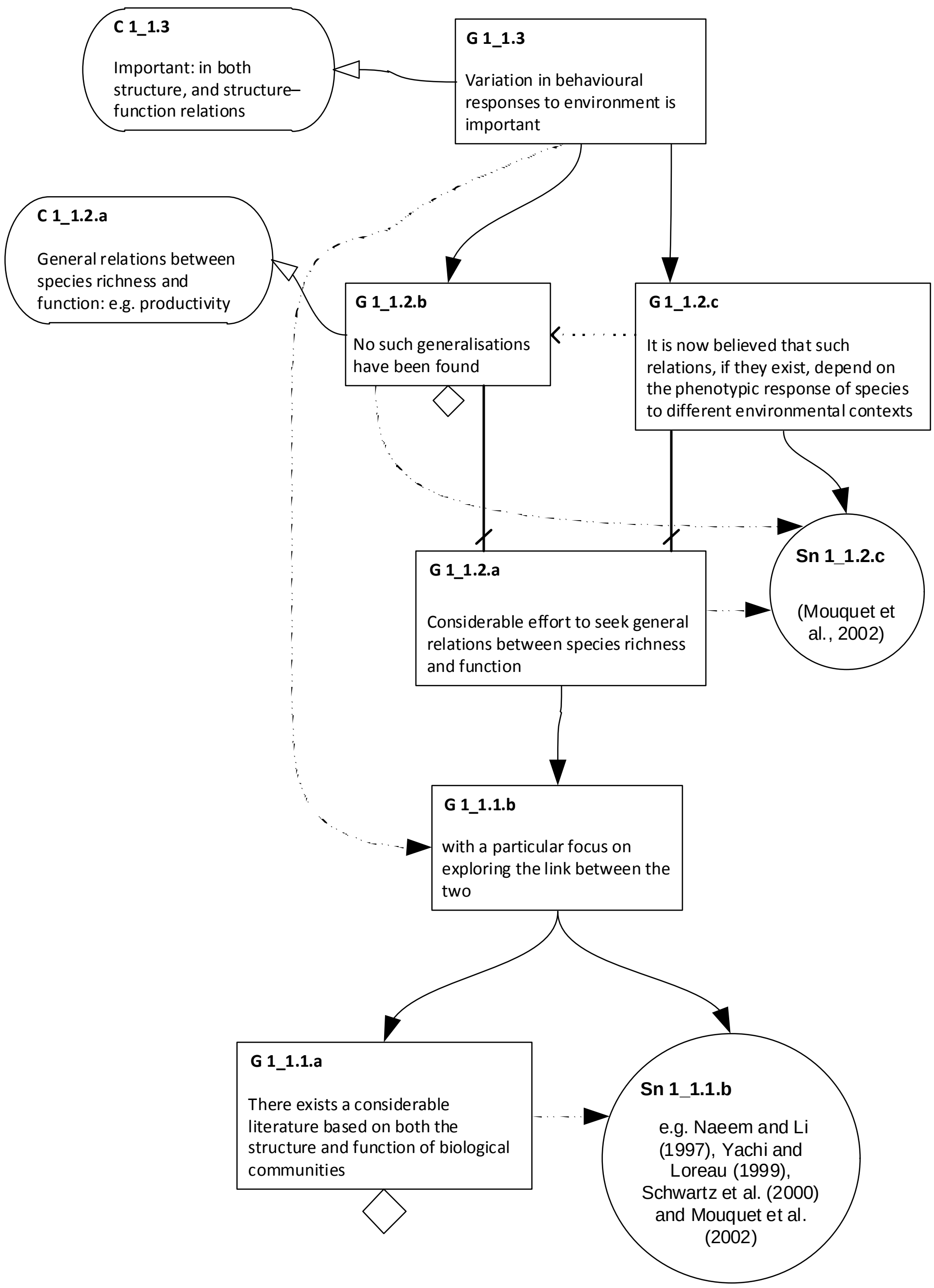


Figure 5.9: Activity 5: further assembling the ‘paragraph’ argument firstly depicted in Figure 5.7

For increased efficiency, we process the publication in a backwards manner, from the Conclusions towards the Introduction.

Looking at the two phrases we have been analysing, we observe two unsubstantiated *goals*: G 1_1.1.a and G 1_1.2.b. These are connected through *PartSupportedBy* relations to Sn 1_1.1.b and Sn 1_1.2.c respectively; the reason for not using *SupportedBy* is that the text is vague in both cases: it is not clear if the *solutions* support only their adjacent claim (G 1_1.1.b and G 1_1.2.c respectively) or all claims within their corresponding phrase. Because we are looking at the first phrases within the publication, there are no *away* GSN elements that can be used to substantiate them, at this stage. Progressing through the publication, we may find such supporting elements or we may find that these will remain unsupported claims.

5.2.2.7 Activity 7: Assess the resulting GSN arguments

Here we look for the top *goals* of the resulting GSN arguments. We also look for the existence and positioning of validity claims. Finally, we take into consideration the unsubstantiated claims composing the Abstract, and the claims constituting the Conclusions section of the case study. Since up to now we have only looked at only the introductory phrases in [23], this activity will be further exemplified in the following section (Section 5.3.5).

5.3 Case Study Example

In this section, we apply the above method to our case study (Section 4.2). In doing so, the author focuses on activities 4 to 7. It is worth reiterating, however, that subjectivity is implicit in all the activities, bar the first.

According to Figure 5.3, the first step we take is to reverse-engineer the Abstract (Section 5.3.1), followed by addressing the Conclusions (Section 5.3.2). Since contrasting the arguments corresponding to these two pivotal sections, and knowing that the Abstract's claims will not be substantiated, the process shall proceed with substantiating the Conclusion claims, this motivating the gradual reverse-engineering of other sections of the case study e.g. the Results or "Predicted Patterns of Diversity" section (Section 5.3.3).

The following subsections are named after the sections found in [23].

5.3.1 Abstract

The Abstract makes seven[1] "promises" (unsubstantiated claims or *goals*). G A_1.1 to G A_1.3.b point towards the Introduction (the background literature) of the paper, G A_1.4 to C A_1.6 towards the Modelling approach and Sample parametrisation sections, whereas G A_1.8 to G A_1.9b towards the Predicted patterns and Conclusion sections. Figure 5.10 reflects this arrangement[2], whereas Figure C.1 proposes a linear arrangement, where *goal* hierarchies are only due to *SupportedBy* relations.

In obtaining these figures, activity 5 required the use of three of the new relations: *FollowsFrom*, *CoupledWith* and *InconsistentWith*. Statements A_1.5 and A_1.6 were classified as GSN *contexts* as they describe (or clarify) two terms from G A_1.4: individuals and "physiologically based" respectively; they could have, however, be left as unsubstantiated *goals*. It was our subjective decision to use the *context* representation.

[1]In Figure C.3 we identify eleven "promises"; note that G A_1.4 can be further divided into three separate claims; meaning that we actually can identify thirteen unsubstantiated *goals*.

[2]Note that this is not the only compact arrangement of GSN elements that we can suggest. Figure C.2 provides an even more compact arrangement, based on the use of *PartSupportedBy* relations.

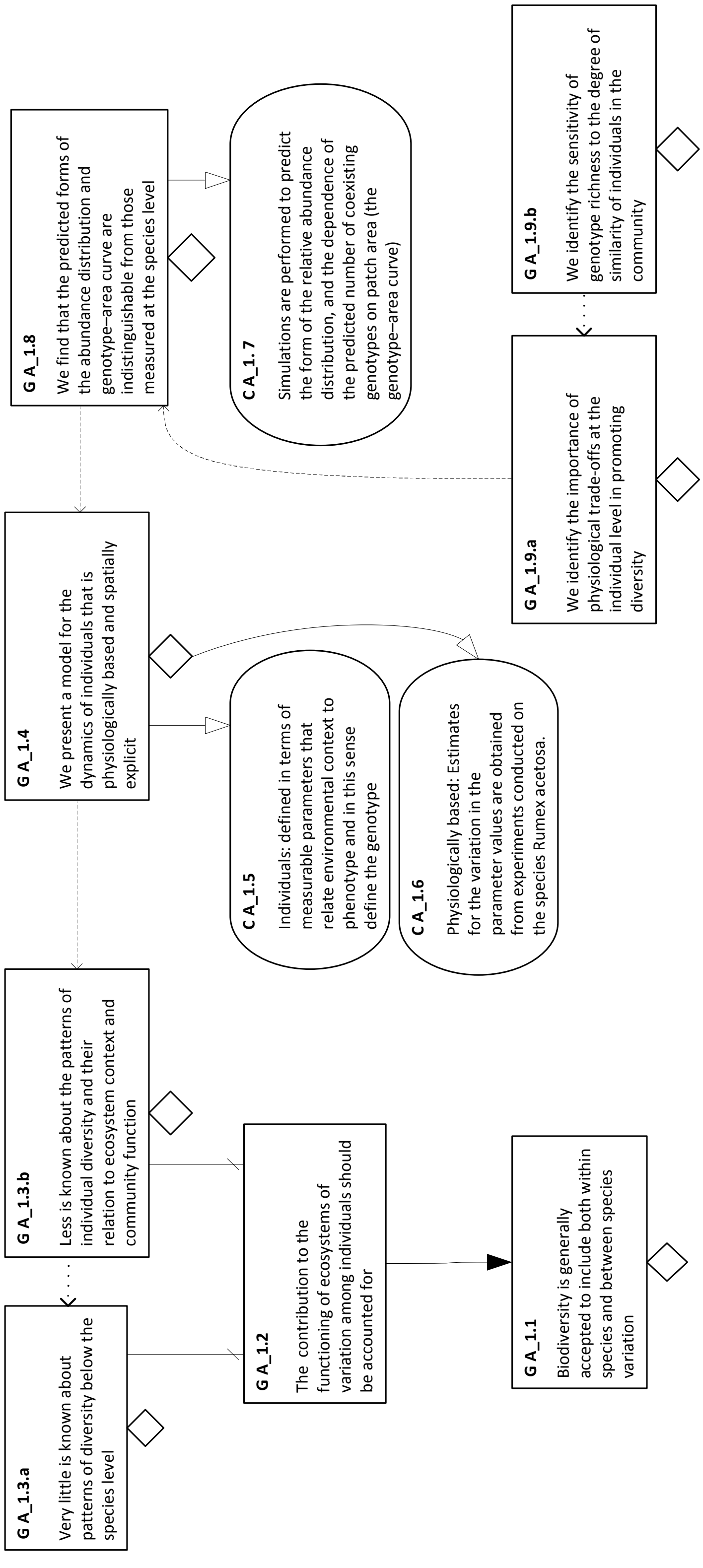


Figure 5.10: GSN representation of [23]'s Abstract (compact arrangement); alternative to Figure C.2

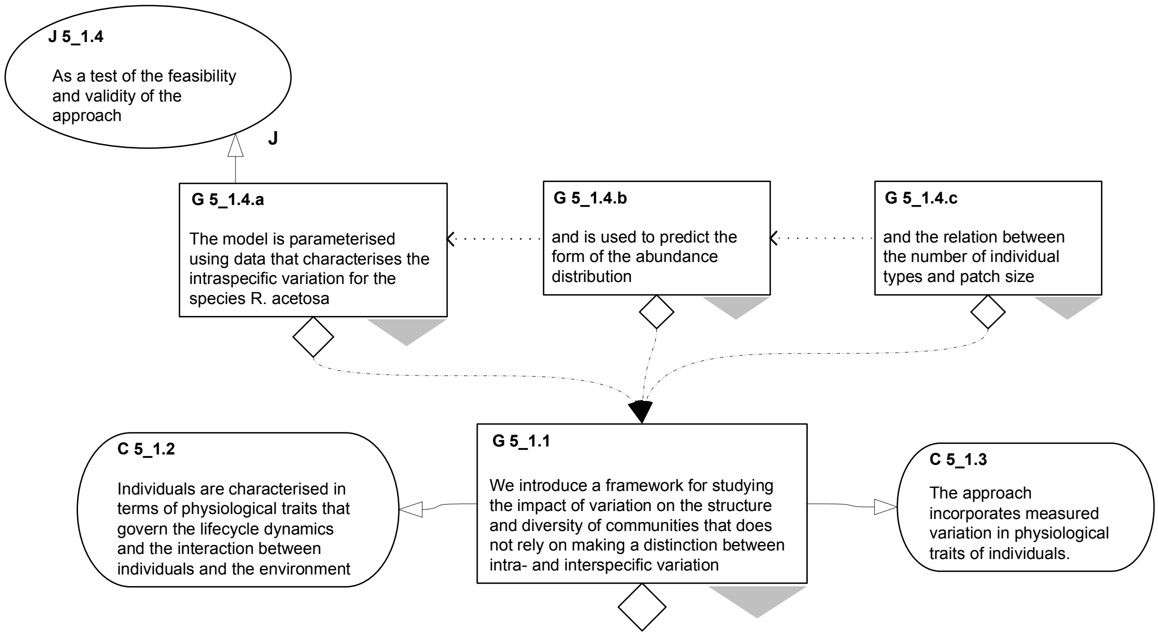


Figure 5.11: Reverse-engineering the first phrases in [23]'s Conclusion

5.3.2 Conclusions

Second in the reverse-engineering process is the Conclusions section. This section is composed of three paragraphs. We address them one at a time.

In Figure 5.11 we use activities 1 to 6 in order to assemble the first statements in paragraph 5_1. Because *goal* G 5_1.1 – the starting point – claims the introduction (hence the existence) of Bown et al's modelling framework, the three *goals* composing phrase 5_1.4 can all be partially substantiated by it; we can employ the property existence pattern described in Section 5.2.2.5, by considering that a claim for a property is substantiated firstly by a sub-argument claiming the existence of the object or concept to which the property belongs, then by sub-argument demonstrating the existence of the property. Note that J 5_1.4, albeit connected only to G 5_1.4.a, holds for the entire sequence of *goals* connected with *CoupledWith* (in this case, *goals* G 5_1.4.a to G 5_1.4.c).

At this stage in the reverse-engineering process, G 5_1.1 to g 5_1.4.c are undeveloped. The shaded triangle shapes located on the bottom left of each *goal* indicate the fact that these elements have been referred to in other arguments (as we shall show further on).

Figure 5.12 depicts a second set of claims put forward in paragraph 5_1. G 5_1.5 and G 5_1.8.b constitute, in fact, the two fundamental top claims of [23]; that is, the claims that relate directly to the scientific meaning of results published in [23]. Both G 5_1.5 and G 5_1.7 which represent the support for the two top claims, are result claims; in addition, *goal* G 5_1.5 represents the validity claim of [23].

Figure C.6 represents the GSN reverse-engineering of the entire first paragraph. The interconnection between *goals* represented in Figures 5.11 and 5.12 was done through *PartSupportedBy* relations e.g. a claim about predicted forms (that is, a claim about simulation results such as G 5_1.5) can be partially supported by the fact that the model was parametrised (G 5_1.4.a) and used to simulate such forms (G 5_1.4.b and G 5_1.4.c).

Performing this activity over the length of the case study paper yields cases that motivate further GSN extensions; such is the case of statement 5 1_9. Here Bown et al. [23] express an uncertainty regarding their results: they must be confirmed by field experiments. In other words, while not implying that there are problems in their work, the authors indicate areas (GSN *goals*) that should be accepted with a cautioned confidence, reminding their readers that the results have not been yet confirmed. Rather than leaving this reality implicit, the authors make it explicit, expressing part of their subjectivity and need for more confidence in their results.

Far from diminishing the importance or quality of their work, we consider that this single statement is a model for how SBR should be presented: an ensemble of claims and expressed subjectivity (including authors' uncertainties). Consequently, we suggest the introduction of another GSN element to express uncertainties, as in Figure 5.13; alternatively, any GSN element should have a decorator

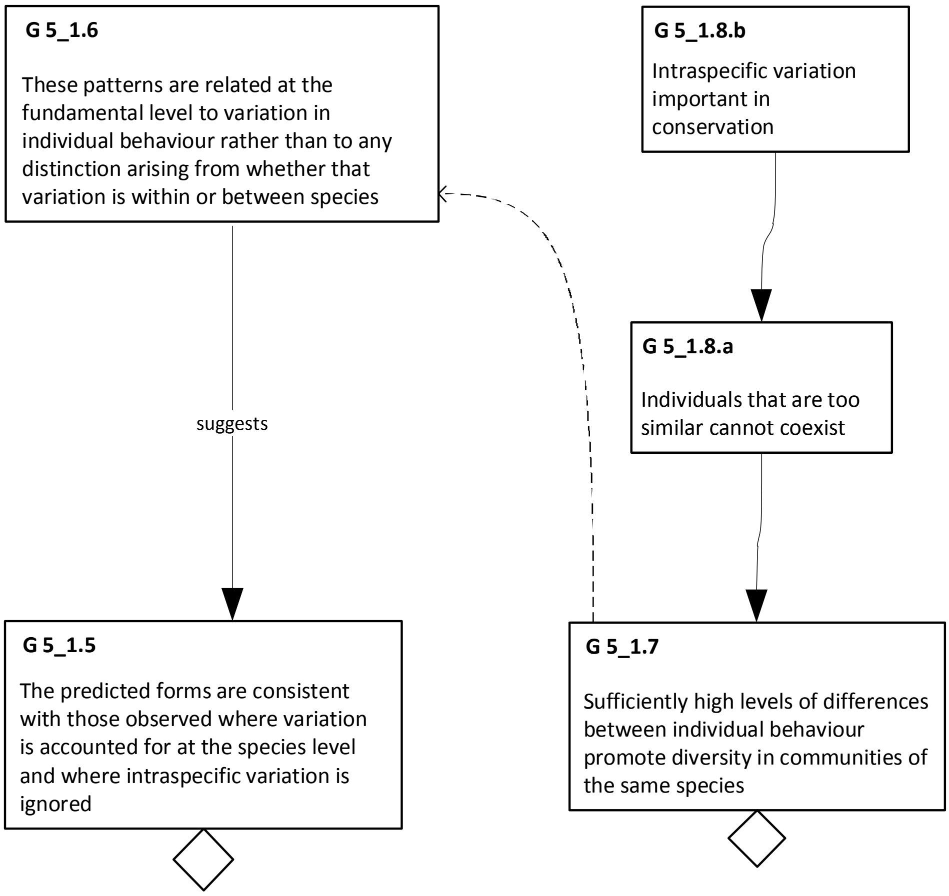


Figure 5.12: Core top claims in [23]'s Conclusion, paragraph 1

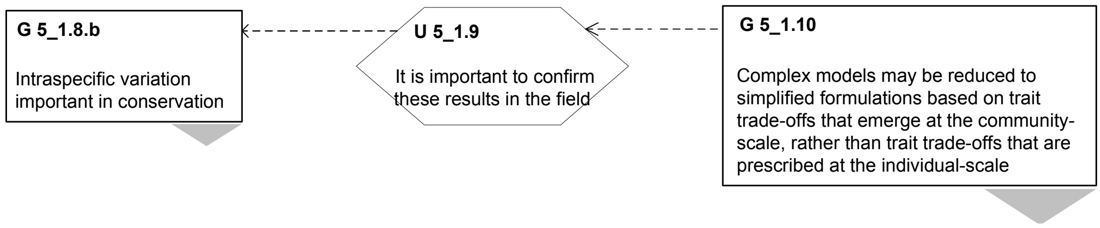


Figure 5.13: GSN *uncertainty* element

for expressing uncertainty.

If this new GSN element were to be adopted, its use within GSN would need to be regulated. For example, U 5_1.9 refers to "these results", making the statement vague unless the results that it is referring to are pinned down; a *context* element could be used in order to clarify this expression; another possibility would be to link element U 5_1.9 to those elements that stipulate the "results", through *InContextOf* relations, but considering the high probability that the results would be expressed as *goals*, which cannot be referred to through *InContextOf*, then the clarifying *context* element seems to be the solution.

Paragraphs 5_2 and 5_3 point towards work that does not fall within the confines of [23] and we shall not address them further here; representations of the reverse-engineered GSN arguments for the two can be found in Figures C.7 and C.8. Following the reverse-engineering of the Conclusion section, we are left with at least six unsubstantiated *goals* (belonging to paragraph 5_1). According the gradual nature of our reverse-engineering approach, we look into the adjacent, Predicted Patterns of Diversity section.

5.3.3 Predicted Patterns of Diversity

The Predicted Patterns of Diversity section provides information about the experimental setups and the results of [23]. The reverse-engineering process highlights that paragraph 4_1's top claims is that the model was used to predict. Figure 5.14 depicts the resulting paragraph argument. According to the GSN standard [1],

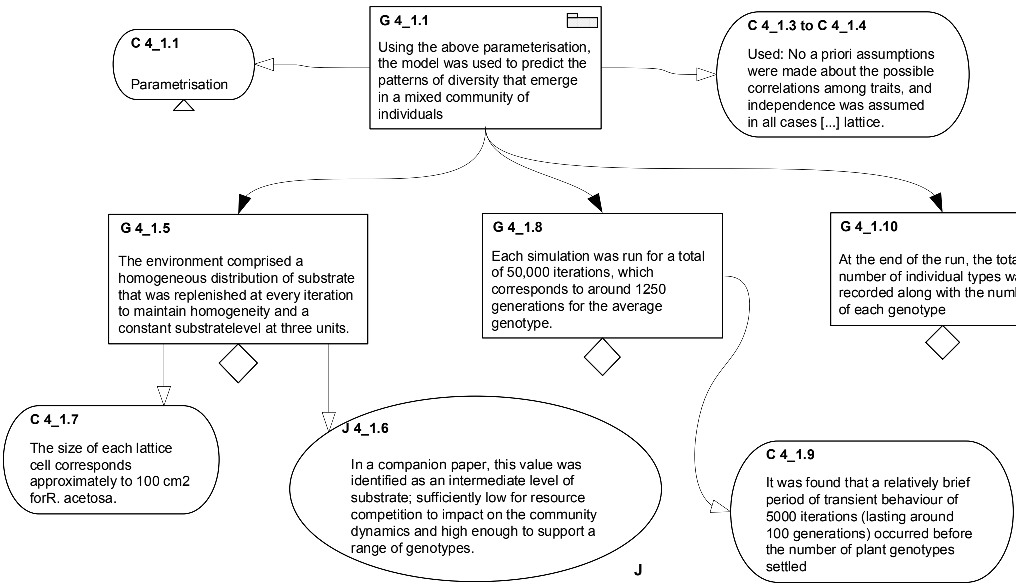


Figure 5.14: GSN representation of [23]'s Predicted Patterns of Diversity, paragraph 4_1

there are two ways of clarifying terms within a *goal*: through *context* elements or by direct substantiation. In our case, statements 4_1.5 to 4_1.10 clarify the way the model was used for predictions. The conversion of statements into GSN elements is constrained by the GSN standard e.g. since statement 4_1.6 represents clearly a *justification* for a term used in statement 4_1.5, this statement needs to be represented as a *goal*; *context* C 4_1.7 also reinforces the decision to leave statement 4_1.5 as a *goal*. At this point, the parametrisation used is represented as a *context* element that requires instantiation (C 4_1.1).

Note that the terminal *goals* are all unsolved; we do not have an item of evidence that can support them since we are not in possession[1] of the initial data files or simulator source code underlying the top claim G 4_1.1.

The section is continued with three subsections, each one paragraph long; each section describes the setup and results for a type of experiment.

[1]At least at this stage of the reverse-engineering process.

5.3.3.1 Diversity vs. Area Size

Let us look at the first subsection, reverse-engineered according to Figure 5.15. The Property existence pattern was employed for substantiating G 4.1_1.3: the consistency claim, concerning Bown et al.'s [23] simulation results can be substantiated firstly by a claim substantiating the experimentation indeed took place and a claim concerning the results. G 4.1_1.5 can be partially supported by G 4.1_1.3; this is not obvious for the text, domain expertise being required for knowing that power-law relations (claimed in G 4.1_1.3) are biologically relevant and found in studies of species richness (as claimed in G 4.1_1.5). G 4.1_1.5 cannot be fully supported as the term 'consistent' is not defined, while further *justifications* or supporting arguments or evidence are provided; the claim remains, however, the top claim of this paragraph.

Note that from the goal structure presented in Figure 5.15 it is not clear what the top claim refers too: "this value" was defined in the underlying G 4.1_1.4, "consistent", although substantiated in the underlying G 4.1_1.3, may not have the same meaning here. More importantly, the top claim does not clarify the experimental context to which it refers or its *purpose*; *context* C 4.1_1.2 provides this information (the fact that the experiments addressed the genotype diversity – area size relation). Reverse-engineering plain text into GSN arguments may not yield sufficiently meaningful arguments, especially in cases where the context of the top claim is misplaced at the level of sub-*goals*.

As observed in paragraph 4_1, the terminal *goals* of this GSN argument are unsubstantiated; to an extent, the fact that results have been provided in the shape of a plot (individualised in C 4.1_1.2.b) could be considered "soft" evidence for the fact that simulations were indeed performed, as claimed in G 4.1_1.2.a and G 4.1_1.1, but we do not have the "hard" evidence (original output files and simulator source code) to back these claims up. G 4.1_1.4 is also unsubstantiated, for the same reason. Accepting the top claim G 4.1_1.5 means accepting the adequate fulfilment of all unsubstantiated claims underlying it.

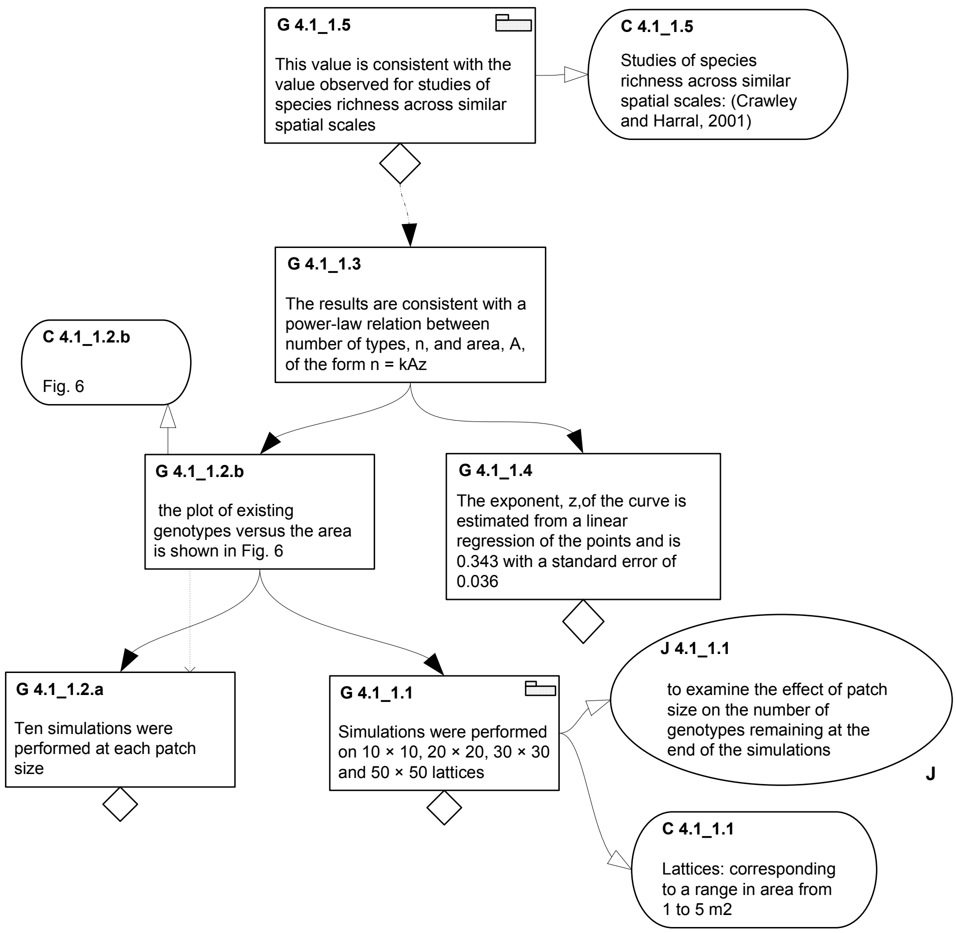


Figure 5.15: GSN representation of [23]'s Predicted Patterns of Diversity, paragraph 4.1_1

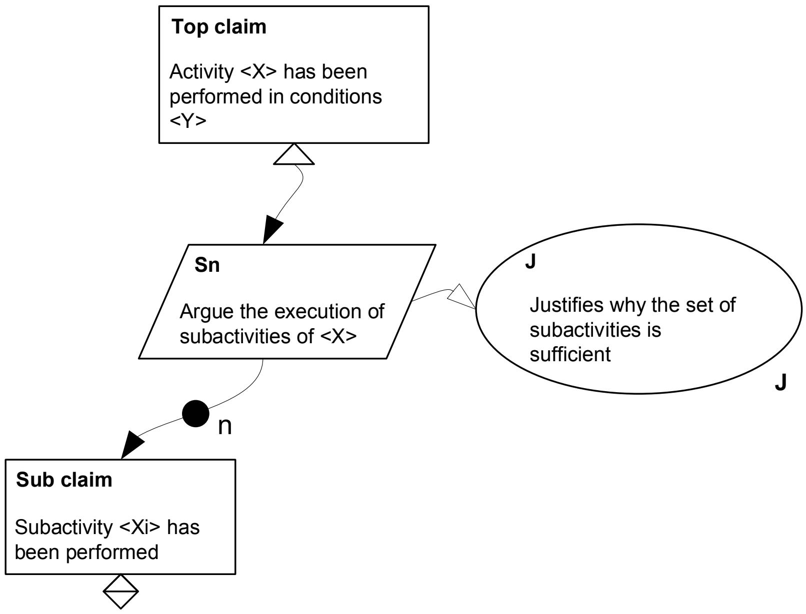


Figure 5.16: GSN pattern: Activity performed

5.3.3.2 Relative-Abundance Distribution

The following paragraph has been reverse-engineered into the GSN argument presented in Figure C.4. If the GSN argument for paragraph 4_2 culminates with *goal* G 4.2_1.6 (claiming the consistency of simulation results with empirical observations), it also reflects the authors' caution towards this claim through the uncertainty element U 4.2_1.7. Top claim G 4.2_1.6 is only partially supported by G 4.2_1.1 which, according to what the author proposes as a new GSN pattern called the activity pattern (Figure 5.16), is supported by *goals* substantiating the simulation experiments.

5.3.3.3 Impact of Similarity

Finally, the third paragraph, reverse-engineered according to Figure C.5, exposes a more elaborate argumentation structure. The top claim, G 4.3_1.19 integrates Bown et al.'s [23] study into the wider picture of ecological research; this *goal*, together with G 4.3_1.16, represent two lines of confirmation of Bown et al's [23] adequacy to the domain concepts and interests at that time. For reasons of clarity, we have introduced *strategy* Sn 4.3_1.16 (not found in the text) and encapsulated the description of the simplified model used by Bown et al. [23] into the GSN model element M 2 (spanning statements 4.3_1.7 to 4.3_1.11); had there been a need to address separately the statements encapsulated by M 2, these could have been expressed as individual GSN elements.

5.3.4 Assembling Section Arguments

Having reversed-engineered the Predicted Patterns of Diversity section, we can gradually substantiate the Conclusion arguments corresponding to the first paragraph (5_1). Leaving G 5_1.1 and G 5_1.4.a unsubstantiated as we have not processed the sections of Bown et al.'s paper [23] that deal with them, we focus on the remaining *goals*. Figure 5.17 presents the result of activity 6 applied to G 5_1.4.c; this *goal* was previously only partially supported by G 5_1.1 (as shown in Figure 5.11) whereas after reverse-engineering the Predicted Patterns of Diversity section, we can further substantiate that the model (or "framework", in G 5_1.1) was used to study the species – area relation by referencing as many *goals* (arguments) from the Predicted Patterns of Diversity section.

In the case of Figure 5.17, G 4_1.1 and G 4.1_1.1 were deemed sufficient for substantiating G 5_1.4.c; G 4_1.1 claims that the "model was used to predict the patterns of diversity" (and although the expression "patterns of diversity" is not clarified here, domain expertise allows us to deduce that it includes the species – area relation) and represents the *top-goal* of an argument that, although unsolved (its *sub-goals* are unsolved), provides a good amount of descriptive information; different from G 4_1.1, G 4.1_1.1 is a terminal *sub-goal*; although it is unsubstantiated at this stage, it is justified (J 4.1_1.1) and part of an argument that points towards simulation results: the reverse-engineering of paragraph 4.1_1 yielded G 4.1_1.5 as the *top-goal* as in order to claim something about simulation results,

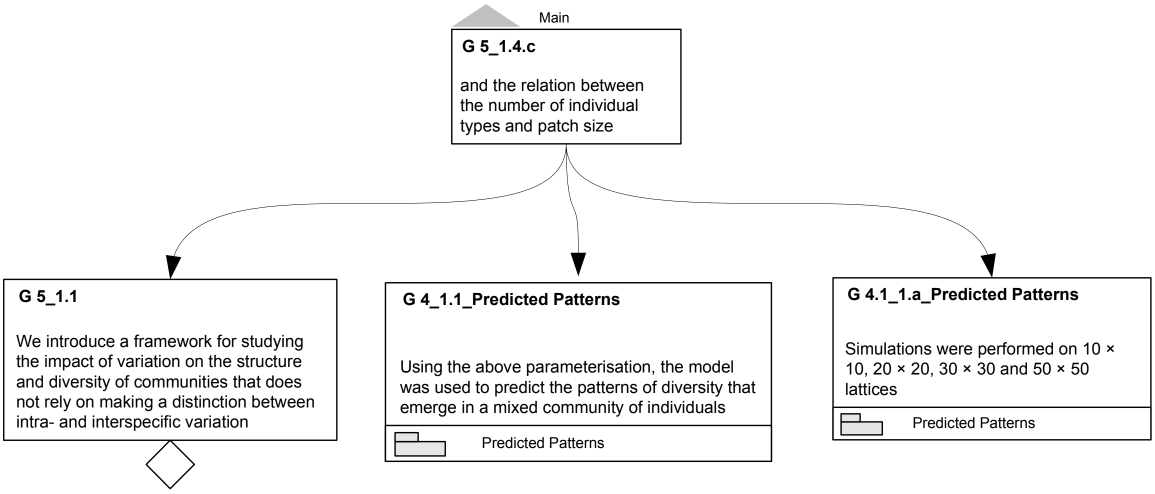


Figure 5.17: Activity 6: retrospectively substantiating claim G 5_1.4.c

one needs to establish the premises i.e. simulations took place[1],

The GSN arguments described in the previous section contribute also to retrospectively substantiated claim G 5_1.4.b, as shown in Figure C.9. Moving towards the higher (core) claims of paragraph 5_1 (Figure C.6), we address G 5_1.5 – one of the validity claims of Bown et al. [23] – according to Figure 5.18; apart from the set of claims belonging to statement 5_1.5, we can now substantiate why the simulation results are “consistent”, through the *away goals* G 4.1_1.5 and G 4.2_1.6; for convenience reasons, we introduced two *strategies* (Sn 5_1.5 Model and Sn 5_1.5 Results, not present in the original text).

The two remaining unsubstantiated claims have been addressed in a similar manner; G 5_1.7 is solved in Figure C.10, G 5_1.10 in Figure C.11.

[1]Albeit (and confusingly enough) a different view on the reverse-engineering process could have yielded an inverse of this argument: the fact that simulations took place can be supported by the existence of simulation results.

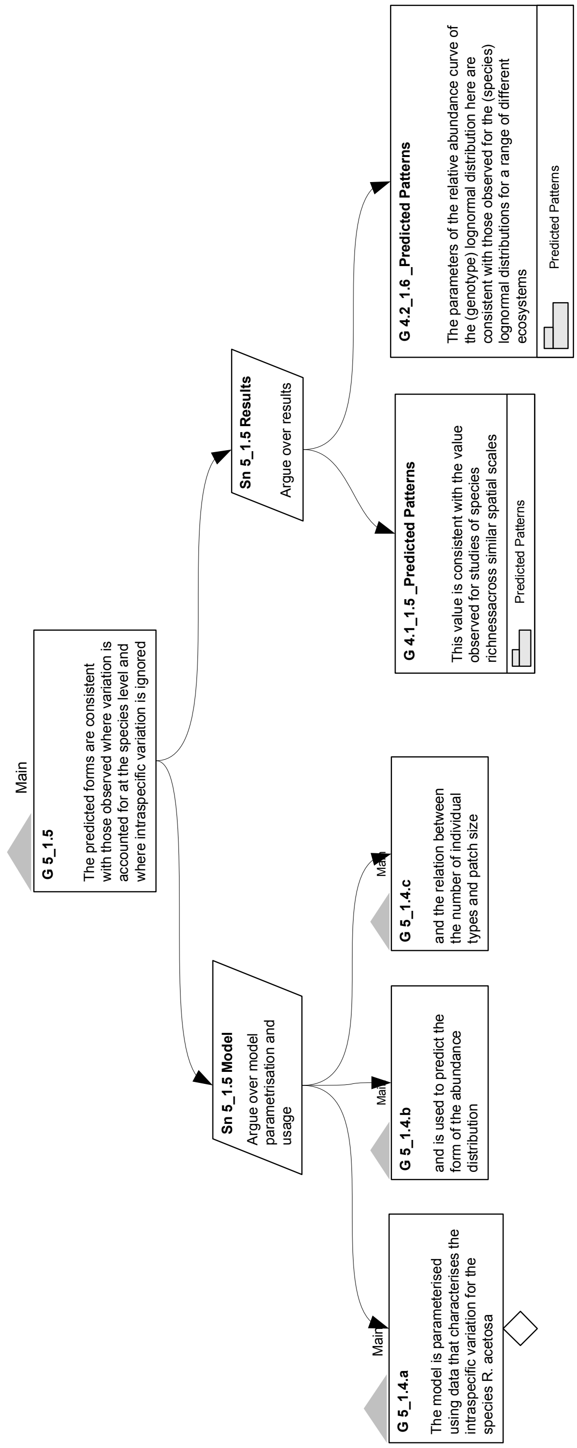


Figure 5.18: Activity 6: retrospectively substantiating claim G 5_1.5

5.3.5 Assessing the Argument Structures

The aim of this activity is to generate information concerning the GSN arguments obtained through the reverse-engineering activities (1 to 5), and consequently, concerning the SBR case study.

5.3.5.1 The Abstract

The first area to consider is the Abstract. Its corresponding GSN argument (Figure 5.10) suggests that Bown et al. [23] make seven "promises" (unsubstantiated Abstract claims). Statements A_1.1 to A_1.3.b point towards the Introduction section, emphasising the niche targeted by Bown et al. [23]; statements A_1.4 to A_1.7 point towards the Introduction and following two sections (Modelling approach and Sample Parametrisation respectively) and represent Bown et al.'s [23] description of their approach; finally, the remaining statements (A_1.8 to A_1.9.b) relate to the Predicted Patterns of Diversity section and express the outcomes of Bown et al.'s research [23].

The argument presented in Figure 5.10 represents one of the possible representations obtained through the reverse-engineering process. If Figure 5.10 abounds in *FollowsFrom* relations, which do not possess a precise logical function, the alternative representation of Figure C.2 replaces these with the *PartSupportedBy* relation, which is more committing. From this latter representation we can observe the importance of G A_1.4 and if this is not well substantiated, then the concluding *goals* G A_1.8 to G_1.9.b cannot to fully substantiated either. G A_1.4 does not depend on the "niche" *goals* G A_1.1 to G A_1.3.b.

Finally, we can evaluate the completeness and clarity of the formulated claims; we suggest marking areas of concern through uninstantiated *contexts*. Figure 5.19 exposes a fragment of the initial GSN argument, augmented through such GSN *contexts* that function as annotations for the GSN user; in total, we add five such elements to the GSN argument (Figure C.3). Provided these annotations are sound, they represent useful cues for further analysis of the text.

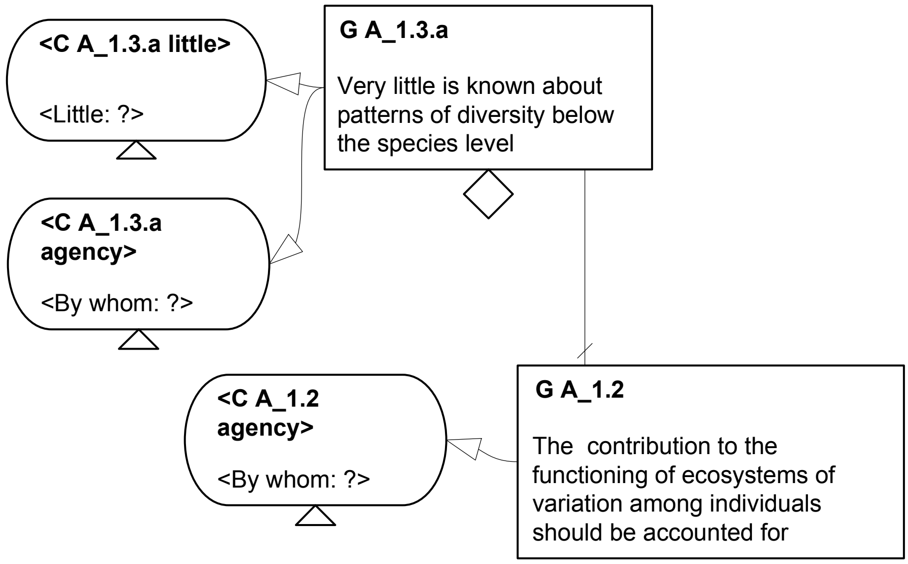


Figure 5.19: Fragment of [23]'s Abstract GSN argument, with annotations

5.3.5.2 Conclusions

Leaving aside paragraphs 5_2 and 5_3, which deal with Bown et al.'s [23] proposals for further work, paragraph 5_1 represents the sole concluding text to consider. The reverse-engineering process yielded three top claims: G 5_1.6, G 5_1.8.b and G 5_1.10. While the latter claim can be considered of secondary importance, in relation to the purpose of Bown et al.'s [23] research, it results that the first two *goals* (Figure 5.12) represent the main scientific claims put forward by [23].

It is useful to note firstly that both these claims are at least a hierarchical level away[1] from the main result claims; secondly, the claims are not addressing the level of the simulation but that of the ecological reality; thirdly, we note the absence of further grounds to inspire confidence in the adequate passage from the simulation domain to the ecological domain (none of the two claims are backed by *justifications* or further explained through *context* elements; no *assumptions* or *strategies* can be identified, between them and the supporting *goals*; they are supported by only one *sub-goal* each). These premises suggest the inductive na-

[1] G 5_1.8.b is two levels away.

ture of the concluding claims, and the leap of faith [1] implicit in their acceptance (that the reviewers of [23] have also made).

5.3.5.3 Abstract vs. Conclusions

In terms of GSN representations, we have already shown that more alternatives can be generated; Figures 5.10, C.1 and C.2 are only three examples corresponding to the Abstract of [23]; we could have pushed the reverse-engineering even further and considered all *context* elements as *goals*, such as presented in Figure C.3. For the purpose of this assessment we shall proceed with the GSN argument presented in Figure 5.10.

Table 5.6 provides a summary view of the Abstract claims and their substantiation. In addition, we differentiate between the Subject (or agency) of the claim and the Object (the noun referred to by the claim). We leave in italics the support elements that represent either unsolved *goals* e.g. G 1_2.4 or elements identified as providing partial support (a *SupportedBy* relation has not been established), whereas bold fonts are used for elements belonging to the Conclusion. We find a wide range of support in this table:

- at one extreme, we cannot find support for claim G A_1.3.b[1]; at the other extreme, G A_1.2 is solved directly in the Abstract;

- G A_1.1 can be matched with elements of the 1_2 paragraph and there is no reference in the Conclusions; G A_1.8 and G A_1.9.b are supported by Conclusion *goals*;

- G A_1.9.a cannot be directly matched with any of the Conclusion claims; the best approximation for a *SupportedBy* relation comes from G 4.3_1.1; together with G A_1.3.a, G A_1.3.b and G A_1.4, it forms a set of Abstract claims for which more investigations need to be done;

- finally, the first half of the Abstract claims lack an explicit Subject (they are

[1] At least not immediate support, for the exact formulation of claim G A_1.3.b.

Goal	Subject	Object	Support
G A_1.1	?	biodiversity	J 1_2.3.a, G 1_2.3.b
G A_1.2	?	contribution [...]	G A_1.1
G A_1.3.a	?	knowledge on [...]	*G 1_2.4*, G_1.9.10
G A_1.3.b	?	knowledge on [...]	*?*
G A_1.4	Bown et al.	model	G_1.9.7, ***G 5_1.1***
G A_1.8	Bown et al.	predicted forms [...]	**G 5_1.5**
G A_1.9.a	Bown et al.	physiological [...]	*G 4.3_1.1*
G A_1.9.b	Bown et al.	sensitivity of [...]	**G 5_1.7**

Table 5.6: Summary of [23]'s Abstract claims

expressed in general terms) whereas the second half claims are attributed to the authors (Bown et al.).

5.4 Discussion

This chapter presented an effort for reverse engineering a SBR publication into GSN arguments, in order to a) obtain a diagramatic, structured view of its main argument threads and b) facilitate the analysis and understanding of the research itself.

5.4.1 Bijective Transformation – An Ideal

An ideal concerning the former objective is that of achieving a bijective transformation: one publication transformed into one (composite) GSN argument. To clarify this idea even further, we could distinguish between a single (composite) GSN argument and a network of arguments. From the above exercise it is conclusive that it is difficult to obtain a unified argument beyound the realm of a paragraph, hence the possibility of obtaining a set of networked arguments is more practical. We were able to achieve this partially: by substantiating the Conclusion argument (paragraph 5_1), *goals* from the paragraph arguments of previous sections were brought in direct connectivity with it; this objective is only partially met as a) [23] has been reverse engineered only partially and b) not all top *goals* from the collection of resulted paragraph arguments were used in substantiating other sub-*goals*.

The clarity of the hierarchical arrangement of GSN *goals*, both at the level of paragraph arguments and of cross-section arguments, is affected by the multitude of permutations in which the *goals* can be arranged; the uncertainty resulting from these degrees of freedom is augmented by other fine-tunings that can be applied to the reverse engineering process (as described further down), which in turn would lead to the generation of different goal structures. In fact, starting from the Conclusion paragraph 5_1, the further down we go on the substantiation path, the more alternatives we find.

Converging to a stable argument structure represents a challenge; due to the lack of clear *SupportedBy* relations and to the fact that plain text is written in a persuasive rather than logical way, it is not realistic to achieve this convergence. Only through the collaboration of [23]'s authors could we agree on a set of representative GSN arguments.

5.4.2 Why reverse-engineering publications into GSN arguments is difficult

There are numerous aspects that render the reverse-engineering process difficult. Here are some of them:

1. a GSN argument substantiates a claim (top goal); a publication usually contains a set of claims, their substantiation ranging from rigorous to missing;

2. in a GSN argument, *goal* elements are organised in a hierarchical topology – they must be related through *SolvedBy* relations. In a publication, a claim can reference another claim, through a non-inferential or evidential relationship (as highlighted in Section 3.4.3): the *InContextOf* relation, with its restrictive

meaning, is also not allowed between *goal* elements;

3. scale: a publication spans several pages, up to tens and hundreds of pages. Due to the geometry of GSN elements, even a limited number of elements take up large proportions of page area, meaning that increasingly large page sizes need to be used;

4. software or paper: can we do the reverse engineering without a particular software? Is just pen and paper sufficient?

5. citations: if a phrase is composed of multiple claim statements, do terminal citations support the last claim, a subset of claims or all the claims in the phrase? In other words, in a phrase composed of multiple statements e.g. "SBR has an important potential, but also relevant problems and without addressing these, the state-of-the-art cannot advance [45]", should the final citation be considered as a *solution* for the last claim, or all three claims?

6. vague terms: e.g. "these findings" (in phrase 1_1.3) succeeds a series of divergent, composite claims. "these" is plural, but does it refer to the last composite claim or to all the previous composite claims in the paragraph?

5.4.3 Analysing the Resulting Goal Structures

At the end of the reverse engineering exercise, we have obtained a set of goal structures (GSN arguments) corresponding to a part[1] of [23]. We established that the best approach was to convert paragraphs into block-arguments, and then interconnect these paragraph-arguments where needed and possible, through *away* elements.

The resulting goal structures facilitated the analysis of [23]. If its Abstract put forward 7 main promises (claims), we then showed how these related to the Conclusions claims and that it was not straightforward to establish bijective corre-

[1] Its Abstract, Conclusions, Predicted Patterns sections, in full, and partially the remaining sections.

spondences; in fact, we were not able to do so in some cases, further support from [23]'s authors being required if intending to finish the one-to-one claim matching.

We also proceeded downstream, investigating the substantiation of the main scientific claims of the case study paper (identified in paragraph 5_1). The resulting goal structure showed that two out of the three top claims from paragraph 5_1 were inductive "leaps of faith". Furthermore, each of these two claims were substantiated by only one sub-*goal* and a justification for why they should be considered adequate was not provided. In fact, the issue of auxiliary information (e.g. *contexts*, *assumptions* and *justifications*) is insightful: due to the emerging hierarchy within paragraph patterns, the top *goal* is, most often, a *goal* lacking explanatory elements; these are defined in phrases that constitute sub-arguments. Consequently, resulting GSN arguments, although having a hierarchical structure, are not adequate according to the GSN standard and the six-step method; if clarity decreases as we go up the hierarchical ladder, then this goes against the top- down nature of GSN arguments.

5.4.4 Adequacy of GSN Standard

The above work suggests that the standard GSN [1] is insufficient for capturing the rich argument structured expressed in SBR. At the same time, the reverse-engineering exercise provided evidence in favour of claiming that plain text, irrespective of its scientific, peer-reviewed nature, cannot be fully translated into GSN arguments; even if the richer Debategraph notation would allow the capturing of more content, the results have a reduced utility. SBR publications can be too long, meaning that the effort required for reverse engineering would be too high and the resulting structured arguments would simply be excessively large. The reverse engineering process is too demanding, even/especially for journal papers.

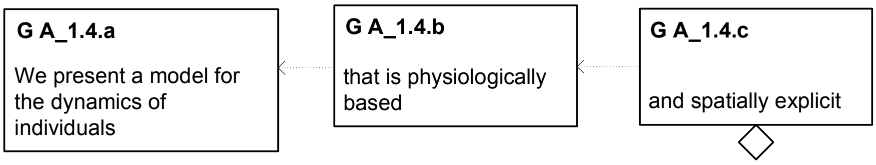


Figure 5.20: GSN representation of composite claim G A_1.4

5.4.5 Reverse Engineering Fine-tuning

The reverse engineering process can be fine-tuned on a number of dimensions. Firstly, the acceptable granularity (or atomicity) for the GSN elements. In activity 2 we proceeded with segmenting phrases into atomic statements; however, we have not provided a definition for the level of atomicity we find acceptable; in the GSN standard, it is said, for example, that a GSN *goal* should consist of one claim (multiple claims being represented through multiple GSN *goals*), however, when we are reverse engineering a complicated statement, we need to decide if to treat it as a single claim or as a set. To exemplify this situation, G A_1.4 (Figure 5.10) is considered a single claim, but we can equally well decide to divide it into three separate claims, as shown in Figure 5.20; in this case, the Abstract does not put forward 7 "promises", but 9 (and the number increases if we apply this viewpoint to the rest of the claims). It could be then considered ideal that the reverse engineering process starts from a coarse-grained representation of elements, then, according to additional analysis needs, these can be further broken down.

A second dimension that can be fine-tuned is that of the level of packaging that the GSN user applies to the paragraph arguments. If the initial translation of the plain text into GSN arguments delivers mainly linear goal structures (e.g. Figure C.1), the GSN user can seek to increase the degree of layering of these structures by replacing *FollowsFrom* (and *CoupledWith*) relations, with *PartSupportedBy* ones: there is a trade-off between these two types of relations.

In an ideal scenario, the GSN user would first obtain the quasi-linear goal-structures obtainable through the standard GSN, then would refactor them by using the non-standard *PartSupportedBy*. Note that *PartSupportedBy* is insufficient for considering a *goal* substantiated. In practice, through experience and through the use of argument patterns (e.g. the Existence Pattern – Figure 5.8), a refactored version of the arguments can be obtained from the first pass through the plain text.

The criteria for deciding what level of fine-tuning to apply should be made explicit and justified.

5.4.6 Insights into the Structure of Plain Text Arguments

One objective of this reverse engineering exercise was to gain insights into the structure of scientific arguments, as they are written in plain text format. We previously (Section 1.2.4) criticised the fact that arguments are not easily accessible; they are scattered throughout the length of a publication, they are not first-class elements and the persuasive rather than logical way in which scientific publications are written, affects the clarity of such information.

The GSN arguments obtain through the reverse engineering of [23] emphasise the fact that, even in the case of a well-written, peer-reviewed paper, published in a respectable journal, the path from Abstract to Conclusions is not a straightforward substantiation of a set of "promises"; vice-versa, going from Conclusions to Introduction is equally a difficult exercise. GSN helps, but also complicates the attempts to reverse-engineer. A first benefit of using GSN is that of using an argumentation framework, equipped with a syntax and semantic; while the process of constructing GSN arguments is not completely formal (and GSN arguments are not completely formal themselves, although attempts are being made into this direction [74]), the level of rigour and formalism it provides is still better than no formalism whatsoever.

5.4.7 The Argument Problem

The above reverse engineering exercise emphasises that there is indeed an argument problem (Section 1.2.4). In the case of this case study, the argument problem is describable both in terms of argument accessibility and argument constitution. We do not focus on the existence of first-class arguments for the fitness-for-purpose of Bown et al.'s simulation model [23], since mainstream ecology is not aware yet about such a type of knowledge representation – and consequently does not require it.

Chapter 6

Structured Arguments Supporting SBR Rigour

The previous chapter emphasised the existence of an argument problem (Section 1.2.4). SA can be used in addressing the publication phase of SBR (in achieving an adequate communication of SBR). However, deficiencies in the public arguments (or publication) layer cannot be considered in isolation[1]. The aim of this chapter is to substantiate the claim that, given knowledge and capability of using SA for their own purposes, researchers could develop and use structured arguments during research, this increasing the rigour of their research.

Expanding this topic, we need to clarify *when* SA is needed and how it can be used.

The chapter proceeds as follows: Section 6.1 presents theoretical considerations on the use of SA for supporting SBR rigour; this is followed by practical examples of retrospective uses of GSN (Section 6.2) as well as for ongoing research (Section 6.3). Section 6.4 details insights obtained from re-engineering Bown's simulator. The chapter concludes with a discussion in Section 6.5.

[1]The expression "a tree is recognized by its fruit" (Matthew 12:33) holds for more than its immediate meaning.

6.1 Theoretical considerations

6.1.1 When is SA needed

The discussion about rigour in SBR can be a far reaching effort. Looking at the literature, we find a range of criticism addressed to aspects of SBR: flaws can be limited to facets of an SBR study or be wide-reaching, they can have limited consequences (due to the limited use of the published SBR in time) or can become hubs of historical error (see [45]). Adopting a simpler approach for addressing the topic of rigour, we consider the following three levels:

1. complete lack of rigour: SBR of this type can be defined by a spurious, unvalidated simulation, and by unsubstantiated, unjustified claims put forward based on experimenting with this simulation;

2. partial rigour: here we can discuss of a systematic approach to SBR e.g. Stanislaw [83], Sargent [77], Pattern-Oriented Modelling [42]; issues of V&V and sensitivity analysis may be addressed, in a partial manner;

3. substantial rigour: this type of SBR provides *justifiable confidence* in its contributions; it is verified, validated, and there is an explicit level of assurance in the quality of the research e.g. the CoSMoS process [14].

SA is not really necessary for the first case, but it becomes more relevant as we climb the rigour ladder; it *is needed* for the latter case – in case a substantial level of rigour is aimed – as it allows structuring arguments that answer questions such as "why should one trust ⟨X⟩ is adequate?", while clarifying what the used terms e.g. "adequate", mean. From another perspective, the higher the rigour level required, the more assurance and validity information needs to be provided and, in order for this information to be effectively presented, structured arguments synthesising them are needed. The alternative, as so often noticed in scientific publications, is that of scattered pieces of information that, combined in a subjective manner, are expected to satisfy the requirements of readers and reviewers.

6.1.2 How can SA be used

6.1.2.1 Knowledge Repository

In SCS, a safety case consists of a safety argument and its accompanying evidence. The safety argument is essential, as it delivers the "core message" (why engineers consider a system to be safe) that can further be supported by tens or thousands of pages of detailed evidence [54]. The combination of structured arguments and evidence is then, an effective way of recording knowledge.

Structured argumentation, through GSN, has already been introduced to domains other than SCS. Ge et al. [35], for example, discuss about the use of GSN for explaining decisions in clinical practice. Our own research e.g. [36; 37] aims to popularise GSN within the SBR community. More generally, GSN can be used in recording in a structured manner the answer to any (relevant) question, whose answer brings together reasoning and evidence.

The author has used GSN as a way of recording literature reviews, for structuring the arguments of this thesis or as a collaboration tool. As the previous chapter has shown, if successful, the reverse-engineering of publications can also generate useful structured arguments; it may be more conclusive to remember or discuss about a publication, based on its (high-level) structured arguments rather than its (possibly lengthy) plain-text representation. In Polack et al. [68], GSN is used in establishing a baseline understanding between domain experts (cancer researchers) and computer scientists (handling the simulation development).

6.1.2.2 Argument Patterns

Argumentation can be useful not only through capturing a rich content of information, in a structured way, but also through the development of arguments patterns that have the potential of summarising existing standards (or defining new ones) [52]. An example is provided in Figure 6.1, which presents a GSN pattern for arguing validity, derived from Sargent's paradigm [77]. According to this pattern, six *goals* need to be substantiated in order to support the (top) validity claim; the relate to the four dimensions of validity identified by Sargent [77]. Each terminal *goal* will, in fact, represent the top claim of a different argument.

The argument presented in Figure 6.1 does not constitute a standard pattern in as much as it is too detailed; we can decompose it into a high-level pattern (the top claim, together with its *contexts* and the first level of sub-*goals*) and four separate sub-patterns, corresponding to the substantiation of each validity type. The high-level pattern is essential – to claim that the approach or the simulation model is valid, is to claim the four types of validity – but the way it is solved (or its leaf *goals* are solved) may vary.

Researchers can develop a multitude of high-level or supporting argument patterns. We can speak then of the CoSMoS [14] pattern(s) for fitness-for-purpose, the TRACE [80] pattern(s) for research adequacy, etc. The benefits (and downsides) of using argument patterns in the context of SBR are similar to those of using them in SCS [52].

6.2 Retrospective Use of SA

Moving on from theoretical aspects, we look first at retrospective uses for SA. It is natural to use our case study then, since this represents a piece of research that has been published and, to a large extent, concluded. In the previous chapter we have already used SA for extracting [23]'s argument threads, to the extent that we needed. From this point onwards, we are interested in addressing the V&V problem. We consequently look at Bown et al.'s [23] validity argument.

In [23], authors claim the validity of their approach in an indirect way: there is no explicit validity claim, formulated as a noun + verb phrase e.g. "Our approach is valid". The only reference to validity is made in J 5_1.4 (Figure 5.11).

In Figure 6.2, we sketch Bown et al.'s [23] potential argument of validity. There is a stark contrast with what would be considered sufficient in Sargent's view (Figure

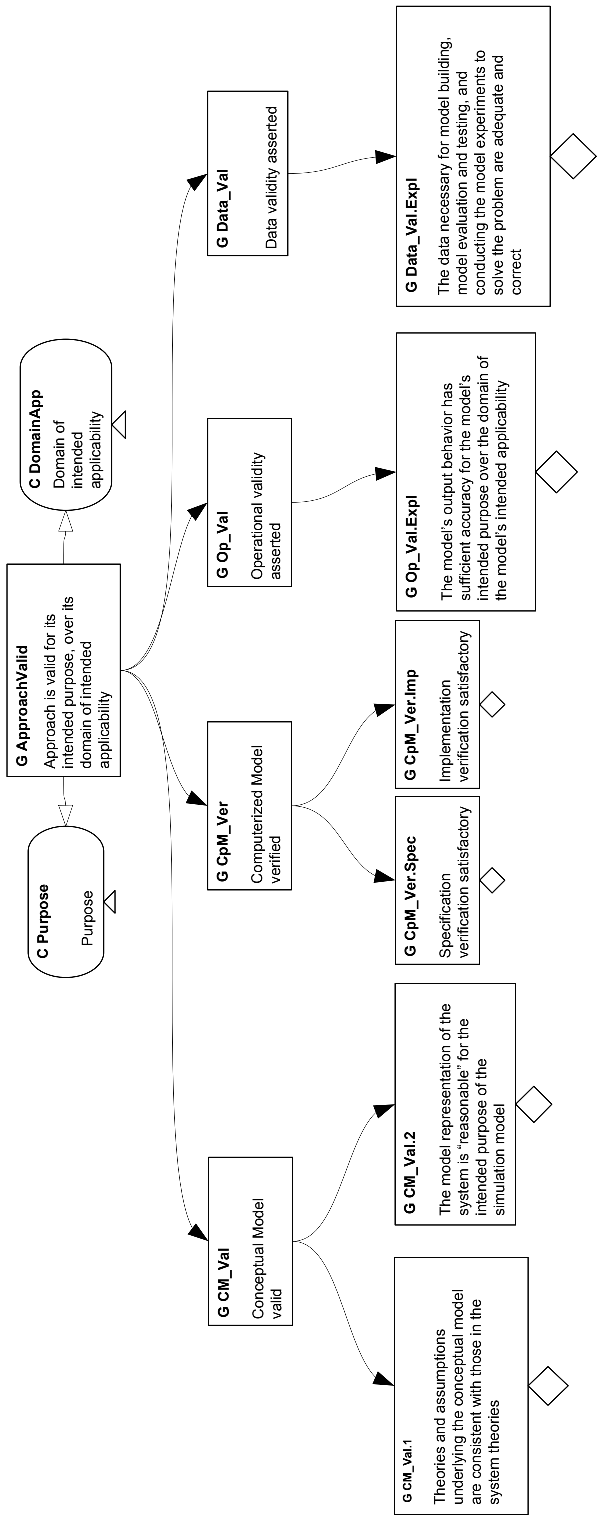


Figure 6.1: Validity argument pattern derived from Sargent's paradigm [77]

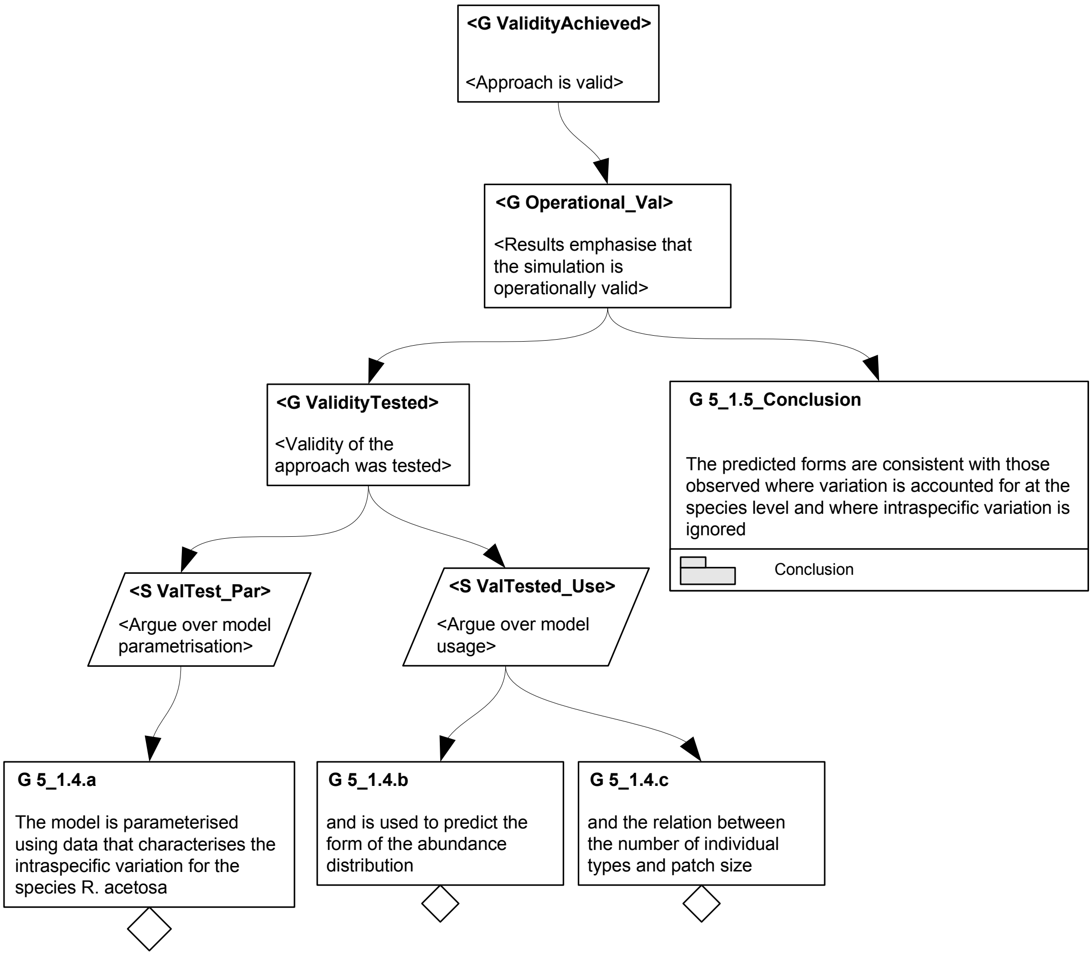


Figure 6.2: A potential structuring of [23]'s validity argument

6.1). Essentially, the *explicit* validity argument expressed in [23] is, in Sargent's terms, one of operational validity or, from Sales et al.'s perspective [61], we are dealing with black-box validation. It is then useful to note that, although aspects relating to the conceptual model and data validity are described throughout [23], they are not brought together in the only part of the text that mentions validity explicitly. *The reader needs to record and put all this information together*, in order to be assured that valid (consistent) simulation results are obtained from a valid conceptual model, simulated through a verified implementation that is instantiated with valid data.

Not making a direct validity claim implies that Bown et al. [23] do not provide a definition for "validity"; a validity test (as Bown et al. [23] put it "as a test of validity, we [...]") can only partially provide it; consequently, the narrower (but fundamental) understanding of operational validity is implied. Looking further into the topic of the top (validity) claim, we address the level of "the basis on which the goals are stated" [1, p.27]; in other words, how precise is the claim and to what extent are its terms defined. In Figure 6.2, we expressed the top validity claim as a rather absolute, unqualified "Approach is valid". The absence of an explicit validity claim from Bown et al. [23] can be interpreted as such a simple, unrefined claim. To contrast this approach, in Figure 6.1 we explicitly added the concepts of purpose and domain of intended applicability to the claim of validity, as required by the validity definitions provided in [77; 78]. These two factors, to be defined in their specific *contexts*, are then applicable to the entire sub-argument, and will be of use in *goals* such as G CM_Val.2 or G Op_Val.Expl. This argument itself is not ideal, but it is a step further in terms of rigour and clarity, from that expressed in Figure 6.2, and this emphasises the benefit of adopting an argument or claim-conscious approach to SBR.

With effort, [23] could be used for substantiating an argument of validity such as that proposed in Sargent's validity pattern (Figure 6.1), and this is testimony to the quality of the work of Bown et al. [23]. However, it is important to remember that unless Bown et al. themselves construct the structured argument, we are only going to obtain a *version*[1] of what may be such a validity argument.

Suppose we would want to reconstitute Sargent's validity pattern, according to the GSN arguments we have obtained in the RE process. We would need to first identify the purpose of Bown et al.'s [23] work and the domain of intended applicability (in Schlesinger et al. [78], this is defined as the "prescribed conditions for which the CONCEPTUAL MODEL is intended to match REALITY"). Albeit explicit for these aspects are provided by Bown et al. [23], we can only assume or deduce.

[1]In the previous chapter we showed that reverse-engineering a publication into GSN arguments does not lead to a single, cohesive argument.

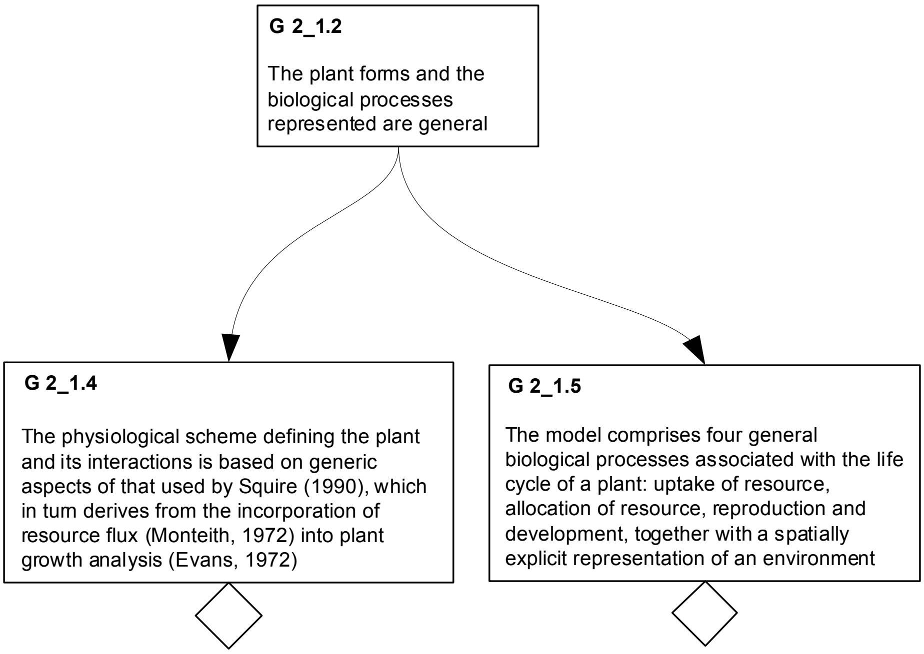


Figure 6.3: Snippet of GSN argument corresponding to [23]'s paragraph 2_1

Going further into the substantiation process, we would need to address the validity of Bown et al.'s [23] conceptual model; in doing so, we would seek *goals* that provide assurance with respect to the quality of the model; statements that describe, rather than justify, the conceptual model, are not evidential. Scanning through GSN arguments generated from Bown et al.'s [23] model description (Section 2 of [23]), we come across statements that seem to suit our assurance requirement; for example, *goal* G 2_1.4, represented in Figure 6.3. Because it refers to "generic aspects", we had considered it as supporting G 2_1.2 (claiming the generality of plant processes); G 2_1.5 was treated similarly. However, on a closer look, the relations between these *goals* are less clear than represented in Figure 6.3: G 2_1.4 refers to the physiological representation of modelled plants, making references to different articles with supporting information; it is not obvious if G 2_1.5 details content that could be found in these cited articles, or is independent from G 2_1.4.

Argumentation uncertainties, such as the ones described above, consolidate the case for the use of SA as a communication tool. This also substantiate the need for interacting with domain experts (in this case, Bown and the co-authors of [23]). Without a good knowledge of the referred domain, we cannot succeed in this argumentation exercise. The reverse casts an optimistic light over SBR: with good domain knowledge, or through efficient cross-domain collaborations, there is an important potential to harvest from the use of SA such as that expressed through GSN.

In addition to Sargent's pattern of validity, we can devise other GSN patterns, for the same or other purposes. We can speak of a CoSMoS [14] fitness-for-purpose pattern or of a TRACE [80] pattern, concerning the fulfilment of a documentation standard for SBR publications.

6.3 SA for Ongoing Research

While having shown that SA can be used for a suite of retrospective objectives, this section provides an insight into the use of SA for ongoing research purposes. Using the same case study, the section addresses one of the fundamental research activities: replicating the results of Bown et al. [23], by re-engineering their simulation. The motives behind this activity are multiple. The first reason is that of validation: replication enables the validation of published research. In addition, this effort allows us to push the boundaries of Bown et al.'s [23] validation, by performing wider-scoped experiments.

As shown in the previous chapter, Bown et al. themselves [23] acknowledge that, due to computational limitations at the time, they were not able to run a sufficiently large number of simulations (for each data point in their figures, they averaged from only 10 simulations), nor to perform large-scale experiments (in relation to real space, the largest simulation described in [23] is of 50×50 grid cells, meaning 25 m^2).

To increase the relevance of [23]'s validity, the model would need to be simulated over larger environments and results aggregated over a larger number of simulations (per parametrisation). Since the results published in [23] are positive, suggesting the adequacy of the model, it is an useful exercise to establish more clearly what is the "domain of applicability" of the model (in other terms, what are the parameter ranges over which the model remains valid).

6.3.1 Re-engineering the Case Study Simulator

In October 2008, the author obtained a version of Bown et al.'s [23] simulator (from Bown), written in the C programming language. In order to perform large-scale simulations, the author realised that a different simulator was needed – one which could be executed in parallel on a computer cluster. While performing more experiments per parametrisation was already possible using the original C simulator, large-scale simulations were not achievable. Consequently, the re-engineered simulator needed to scale the number of computer nodes used, in order to extend the spatial scale it simulated.

In addition to the above, the author observed that the C simulation (let us call it *cplants*) was unlikely to be the one used in performing [23] experiments; this was apparent as the source code did not contain specific functionality for outputting the data that was presented in [23], nor for accepting inputs (i.e. different parametrisations). Following further discussions with Bown, it was acknowledged that this was a different version of the code (probably one prior to Bown et al's research), and the exact version used in [23] was no longer available. Consequently, the C code was enhanced so that it could process inputs and output the type of required data.

The author also observed that the C code, albeit functional, was not optimised for high performance. It was obvious why, on slower computers, the simulation was only executed a limited number of times, on a limited scale. The architecture was preserved and the code was not further enhanced, however, in order to preserve its integrity.

To perform the re-engineering targeting large-scale simulations, we turned to the occam-π programming language, which allows the explicit parallelisation of execution code and, through the use of programming patterns [75], this can be further distributed across computer grids. At this stage, the author aimed only to re-engineer *cplants* into a single-machine occam-π simulation (*occplants*).

6.3.2 Arguing Equivalence of Simulators

If structured argumentation is useful and needed in the large e.g. in order to put forward fitness-for-purpose cases [68], and to structure arguments behind scientific claims, here the aim is to show that it is also useful and needed in the small i.e. in arguing lower-level, technical aspects of research. To do so, we focus on the concept of replication; it is generally accepted that replication of results is fundamental to the establishment of scientific truth or, to be less abstract, of *confidence* in scientific research; replication, together with other other quality assurance activities such as verification and validation, is a key factor in substantiating such confidence.

The literature of SBR emphasises, however, the fact that replication is not necessarily a straightforward exercise; works such as Wilensky and Rand's [89] suggest that it can represent a long, challenging process, relying more on human communication than on formalisms; failed replications can also be subject of long debates, such as that exemplified by Will vs. Macy and Sato [90; 91]. There is a need for *confidence* both in the original M&S and in the result of the replication process. Structured Argumentation plays a favourable role in this context.

In 2009, we published a first study on the use of structured argumentation for clarifying the result of a M&S replication process [36]. If Wilensky and Rand's approach is driven by the iterative testing and adaptation of a replicator (M&S replica) and Will's can be described as assumption driven (the model description was broken down into assumptions, from which the replica was generated), our approach consists of a three-fold strategy of arguing that the science, engineering and the outputs of each simulator are similar. Figure 6.4 presents a GSN

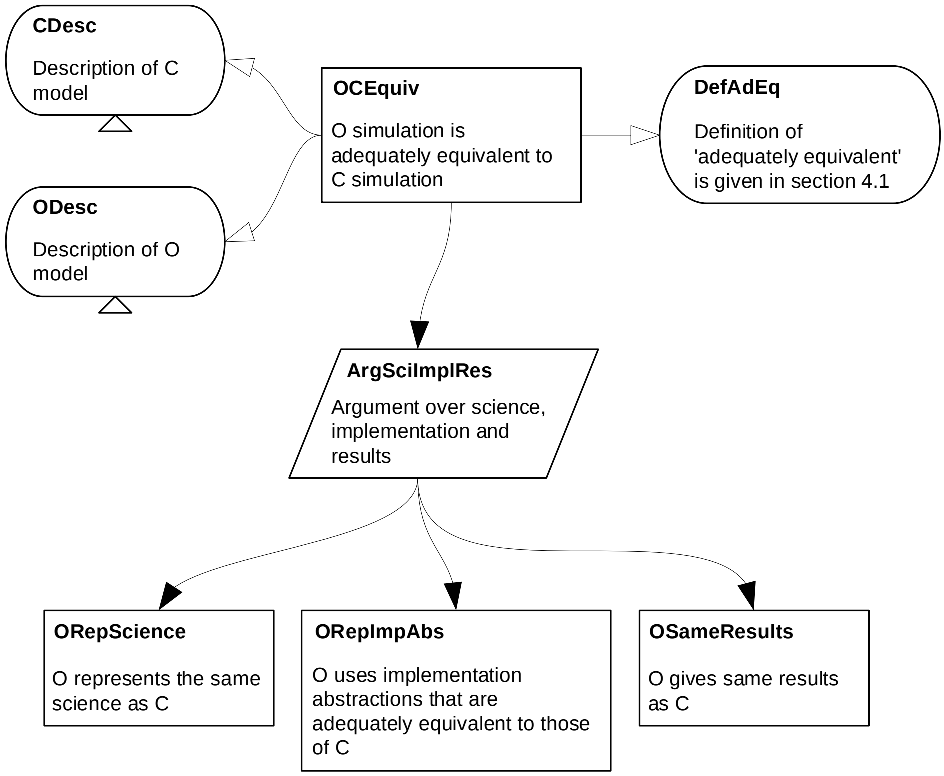


Figure 6.4: Top claim and strategy for arguing the equivalence of the two simulators

argument summarising it.

GSN practice requires us to clarify the basis of the top *goal* immediately after formulating it (it is the second step in Kelly's six-step argument construction method [54]). Appropriate *contexts* were added, accounting for the description of the two simulators and for *our* definition of adequate equivalence. GSN allows *context* and *solution* elements to point towards the exact location of the required information; in the case of [36], a definition for adequate equivalence of simulators was provided, but the *contexts* intended to describe the two simulators were left uninstantiated.

The question about how to effectively refer to a simulation, in a SBR argument, remained unanswered; was referencing the source code of each sufficient? Should we have pointed towards their binaries (executable files) or towards UML diagrams that described their implementation? From this perspective, this top *goal* needed to be revisited and its basis clarified. In the context of this thesis, Appendix D, sections D.1 and D.2, carry the details that describe the C and occam-π simulators respectively. Consequently, a revised GSN diagram would see *contexts* CDesc and ODesc being represented as supported (without the underlying triangle that mark the fact that they are undeveloped).

Looking at the strategy adopted for substantiating the adequate equivalence claim, the existence of the *OSameResults* branch is understandable: it represents Sargent's operational validity [77] (of the replica against the original simulator) or Sales et al.'s black-box validity [61]. The way this *goal* was substantiated (Figure 6.5) has been described in detail in [36]; the technical argument presented in Figure 6.5 corresponds to the realm of the computer scientist; if the *OCExperiments* is self-justifiable, the *OCBoundaryCases*, which can be complemented by other software testing techniques [19], suggests that a software engineering expertise was used in setting the path for substantiating *OSameResults*.

Having emphasised the importance of the results branch, it is important to explain why it was not considered sufficient to rely on the homomorphism of simulation outputs. Firstly, it is acknowledged that a given set of outputs can be reproduced by a multitude of models; this is the main distinction between definitional and descriptive models [69]: descriptive models are not bound to a given structure (they do not need to be "realistic"), their only aim being that of approximating or reproducing data. Consequently, if the two simulations yield the "same" results, this only tells us of a descriptive homomorphism between the two, not a definitional one. In addition, from a practical perspective, the two simulators cannot be tested over the same range of parameters, since the C implementation is not as computationally efficient as the occam-π one; large discrepancies in computing efficiency deter such comparisons. As such, even if the *OSameResults* branch would be the only one considered, it would be costly to obtain sufficient evidence for fully substantiating it.

To argue the adequate equivalence of two simulators is to use a more in-depth

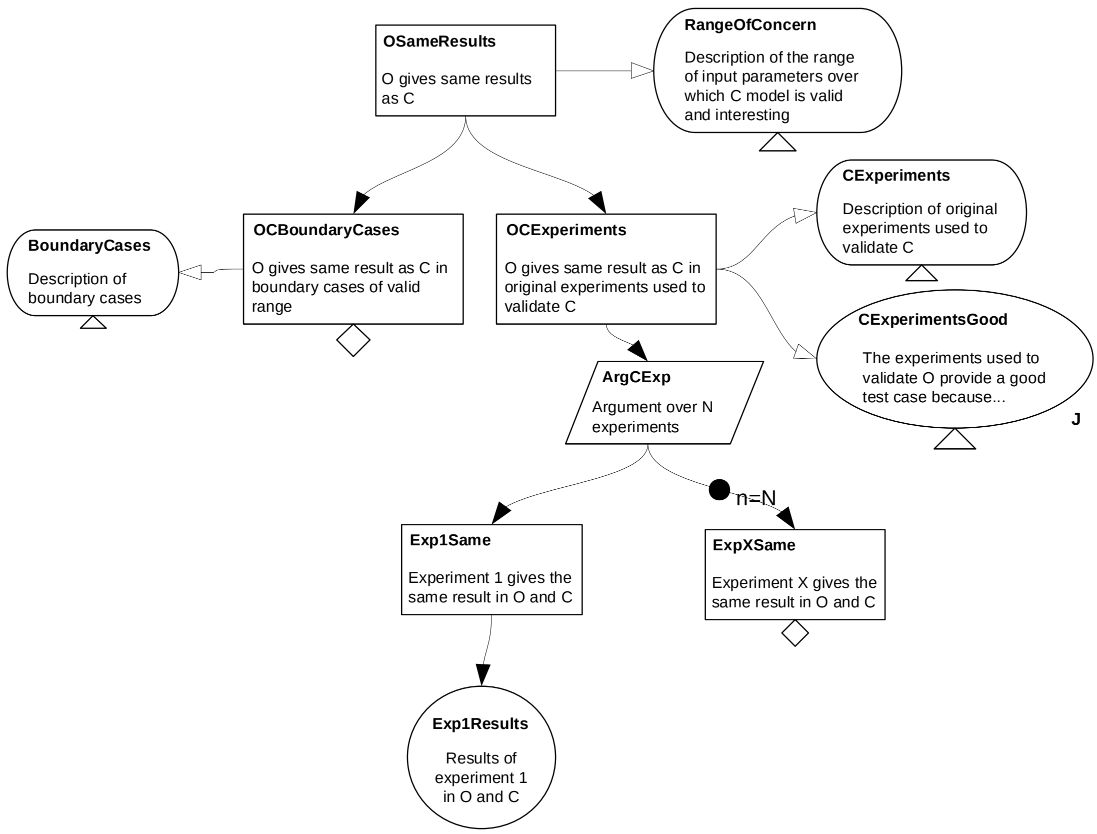


Figure 6.5: Argument for the equivalence of results

approach: that is, to address the engineering and conceptual levels. Figure 6.6 presents a GSN argument for the former purpose, while Figure 6.7 for the latter. On the engineering level, we seek to make a case for the structural similarity of the simulators; this implies that not only are the data structures and algorithms equivalent between the two implementations, but that also the two can be used in the same way (they accept the same parameter values); it came from our experience with the (what we believed to be the) original C implementation and its lack of input – output capability, that we realised the relevance of this input – output layer with respect to the equivalence of simulators.

Finally, arguing that the two implementations are based on the same science (conceptual model) is intended to counter-balance any remaining uncertainty generated by the incomplete solving of the engineering and results branches. In other words, if the results statistically match, and the implementations are structurally similar, remaining assurance deficits[1] can be outweighed by the adequate substantiation of this third and final argument strand.

Note that in solving the leaf *goals* of *ORepScience*, we head again into the domain of assumptions and implicitly into the assumptions problem (Section 1.2.3): to make a claim spanning "all" biological assumptions requires investing efforts into addressing the assumptions problem. Will and Hegselmann [91] provide a near-enough example of assumption awareness: before embarking of their replication exercise, they make a list of all the assumptions they could identify in the target model. Since their aim is not that of managing assumptions, the list they obtain is coarse-grained.The following chapter focuses more on this topic.

All the arguments presented in this section can be transformed in GSN argument patterns, through the removal of *solutions* and contextual information that is related to the specific case presented in [36].

[1]Hawkins and Kelly [53] define the concept of assurance deficit through "weaknesses or limitations in the sufficiency of the evidence". They provide three markers identifying an assurance deficit: 1) the type of evidence is incapable of supporting the safety claim, 2) an instance of that type of evidence is incapable of supporting a safety claim and 3) the instance of that type of evidence cannot be trusted to deliver the expected capability.

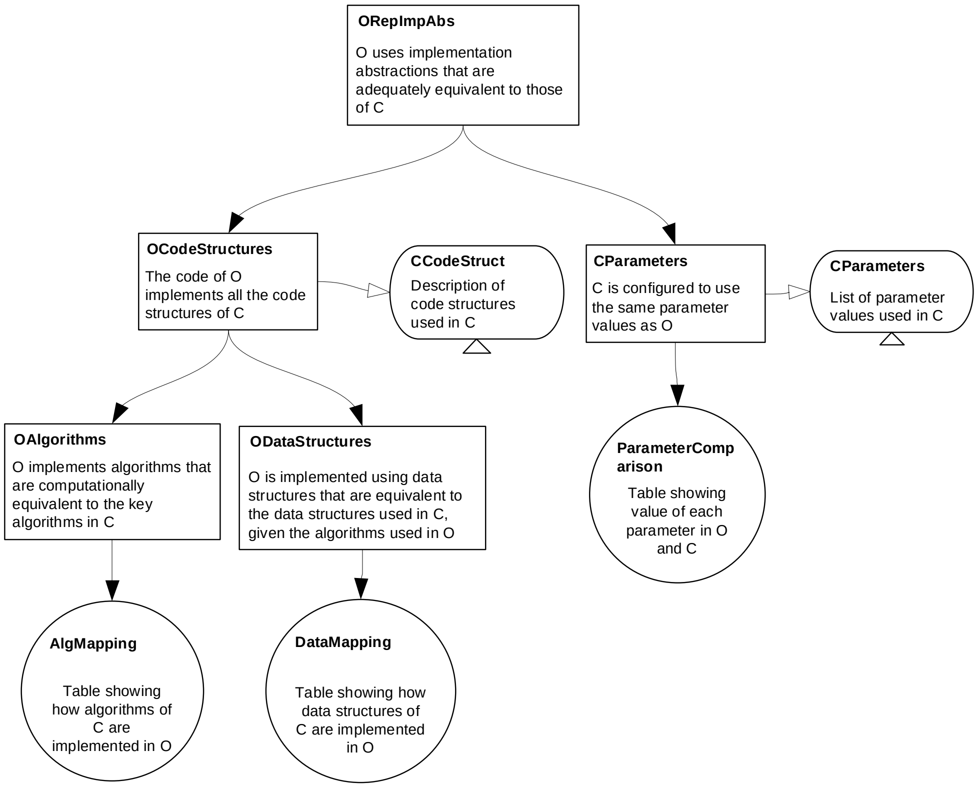


Figure 6.6: Arguing the conceptual similarity of the two implementations

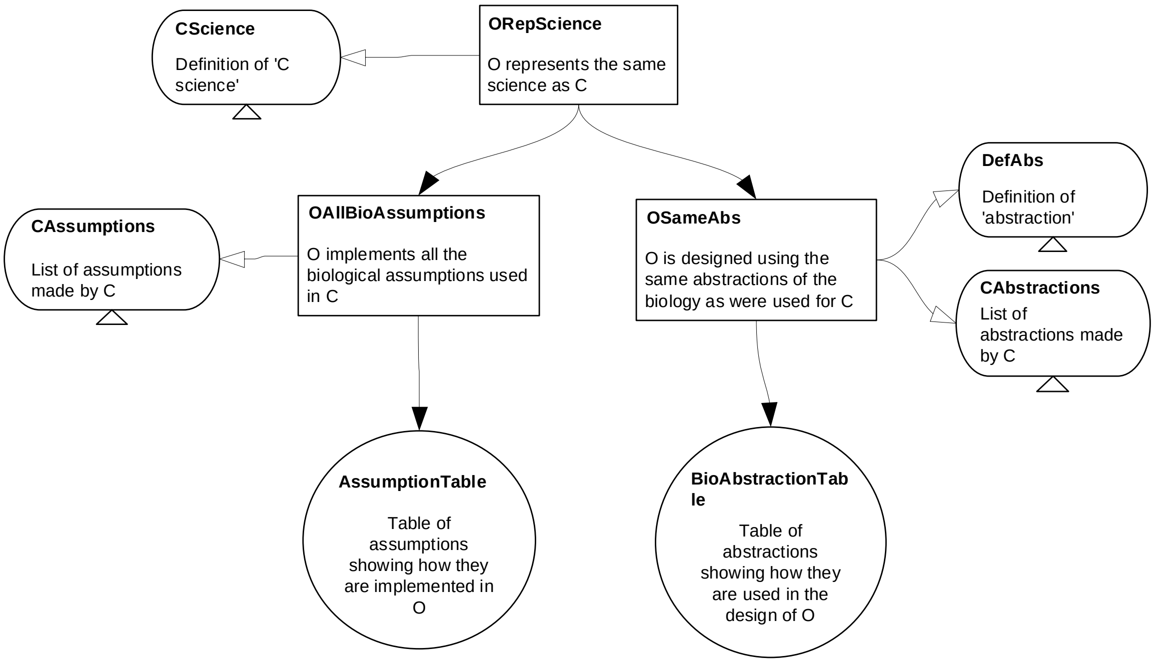


Figure 6.7: Argument for the equivalence of the science behind the two simulators

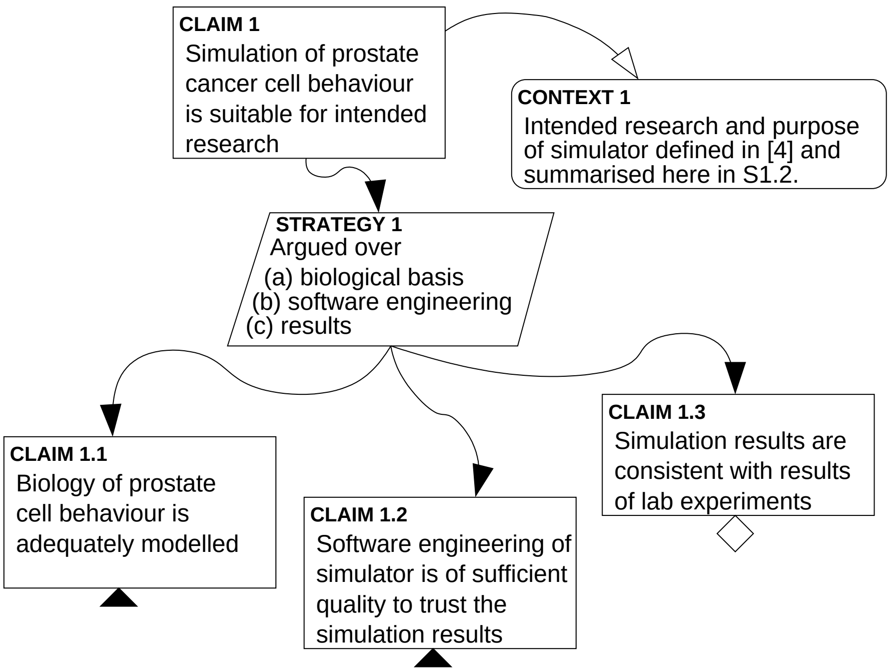


Figure 6.8: Top argument in the Alkan cancer model [68]

6.3.3 Fitness-For-Purpose Arguments in Cancer Model

More recently, Polack et al. [68] discuss the use of GSN for arguing the fitness-for-purpose of a complex system simulation used in cancer research. The approach adopted here is in the same philosophy as that presented in [36] – the top *goal* being substantiated through a three-fold science–engineering–results *strategy*, providing further basis for considering it a pattern.

The top argument provided by [68] is represented in Figure 6.8. The benefit of using GSN is again emphasised through this diagram: at a glance, we observe that albeit adequacy of simulation results is claimed, the *goal* is undeveloped – in [68] we shall not look for, nor expect information regarding simulations to be provided. We shall however expect to see how the biological basis and software engineering arguments are substantiated.

Use of GSN has also stimulated a better understanding of the domain of applicability of Polack et al.'s model. By recording what it means to claim that the scope and scale of the model are adequate (Strategy 1.1.2 in Figure 4 [68]), they bring into discussion the fact that the prostate was model as a closed system of cells. The consequence is *goal* 1.1.2.2, which exposes to scrutiny the fact that the computer scientists agreed with the cancer experts that such a modelling decision was acceptable; furthermore, as Polack et al. acknowledge, this exposure triggered a reassessment of the existing argument and model.

6.4 Simulator Re-engineering – Insights

In Section 6.3.2 we described theoretical work concerning the adequate equivalence of two simulators. In this section we describe insights obtained from performing the actual re-engineering. As mentioned in Section 6.3.1, the C source code we received from Bown et al. was soon identified as not being the one used in generating [23]'s results. We initially believed that this was only due to its lack of input – output capability. As the re-engineering process progressed, inspired both by the C code and the conceptual model description, we soon gained more insights into the reality of model.

6.4.1 Essential Uptake

The essential uptake function (U_e), describing a plant's resource requirements depending on its development stage, was defined by Bown et al. [23] as a sigmoidal curve [23, Table 2]. Two of the parameters of this function (α and β) were defined as distributions i.e. as means and standard deviations, their values being calculated at run-time in a stochastic manner[1]. In our implementation, we noticed some simulations taking much longer times to complete and in cases, we even had to abort runs as they were progressing too slowly. Further analysis of this situation allowed us to establish the cause: the way the β value was stochastically

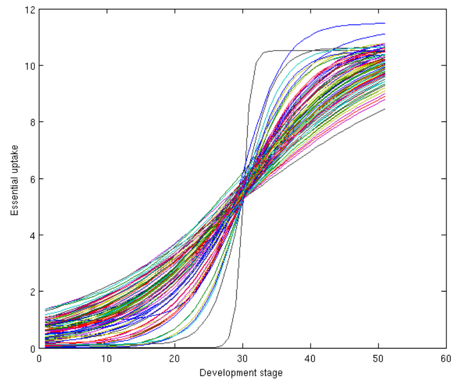
[1]The mean value was corroborated with the standard deviation multiplied with a Gaussian factor, obtained from a function using a random number generator (RNG)

sampled from its distribution meant that, on occasions, its resulting value was very low (i.e. lower than 2) compared with its mean (8.12).

The effect can be observed in Figures 6.9 and 6.10, generated from a Matlab simulation of the of the U_e function. In Figures 6.9, while U_e is monotonically increasing (in a sigmoid manner), as $\beta < 2$ and decreases towards 0, U_e becomes more rugged, approximating a step function rather than a sigmoid one. This implies that young, undeveloped plants, require little or no resource up to a given development stage, then suddenly transform into "devouring" individuals, requiring a maximum amount of resource.

Figure 6.10 presents the reverse: as β becomes negative,U_e becomes a monotonically decreasing function; in other words, seeds have higher resource demands then mature plants, which survive practically without resources. Clearly this was not a feature declared in [23], nor expected. In the end, this can be a problem with our interpretation of Bown et al.'s [23] parameter specification i.e. $\beta = 8.12 \pm 2.65$ perhaps implying that β cannot take values outside the 2.65 standard deviation; however, since "normality" was discussed, and the C implementation used a Gaussian function, this could imply that in Bown et al.'s experiments [23], they did not come across such $\beta < 2$ scenarios. This would be supported by the fact that, due to computationally prohibitive simulation times, Bown et al. [23] performed only a small number of experiments (10 simulation runs over 5 parametrisations) and as such, the $\beta < 2$ scenario may have not been encountered. In our case, benefiting from the use of a computer cluster and the execution of hundreds of simulations per parametrisation (so that we capture the significant behaviour of the simulation [72]), this scenario was encountered more often.

Since the C code the author possessed was not the one used in Bown et al.'s experiments [23], but a precursor of it, we did not have reasons to doubt the validity of Bown et al.'s results [23], but became more aware of the need for transparency, source code scrutiny and for evidence, from SBR authors in general, that such "small" details have been identified and addressed.



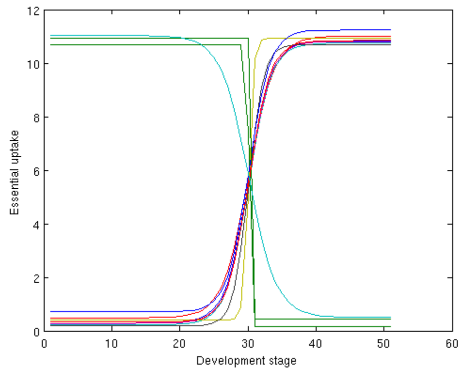


Figure 6.9: Examples of acceptable essential uptake function (U_e) for a pool of simulated plants

Figure 6.10: Examples of unexpected essential uptake function (U_e) for a pool of simulated plants

6.4.2 Reproductive Frequency

Another set of aspects that were observed, concerned the topic of plant fecundity. Firstly, Bown et al.'s model [23] speaks of time dependent reproduction (R_t): plants become fecund (they reproduce) in a cyclic manner; for each plant, a value representing the number of days between reproductive events, is sampled from its species distribution and, throughout its life time, the plant will reproduce with this precise periodicity (as long as it has sufficient resources).

In the C code we obtained, however, it emerged that plants were indeed using this time dependent reproduction, but the generation of a reproductive event was not based on the plant's own age; in theory, after each R_t days in a plant's life, the plant would reproduce; in practice, after each R_t in the global time of the simulation, the plant would reproduce. As such, if a plant's $R_t = 50$, and the global simulation time $t = 50$, irrespective of its age, the plant would generate a reproduction event; *a seed could reproduce*, provided it had sufficient resources.

Although such extreme cases were improbable, and as stated before, we could not infer that the same implementation had been used in Bown et al.'s [23] experiments, this discovery warranted the creation of a separate branch of the

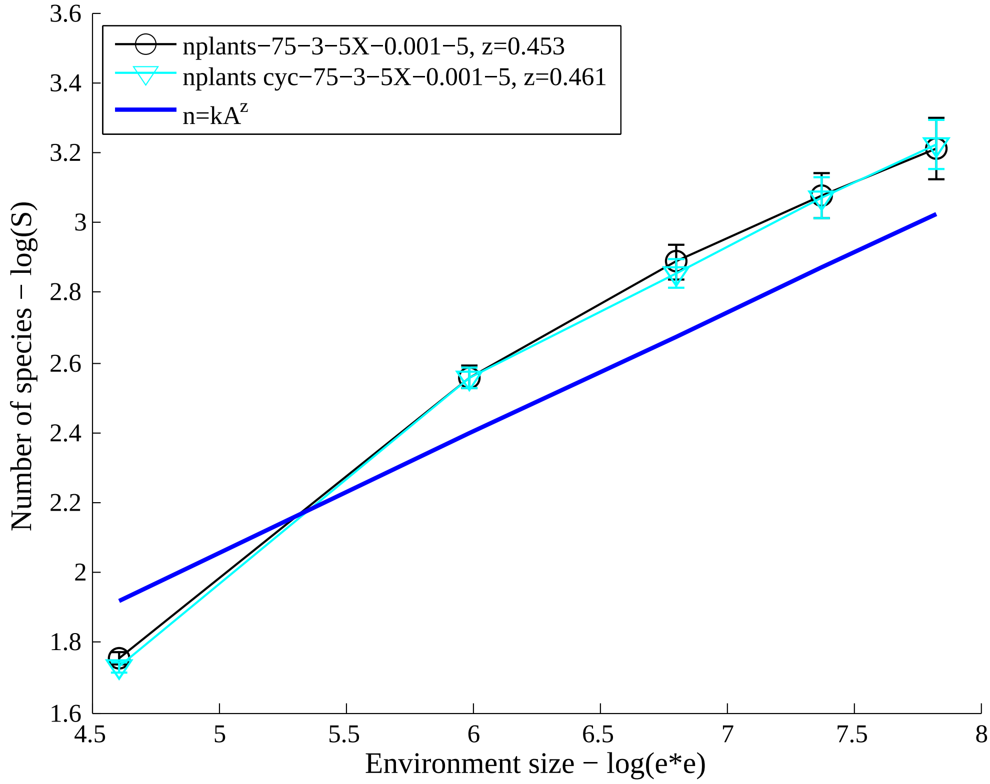


Figure 6.11: Comparison of the species-area curves generated by the *cplants* and *cplants_cyc* simulators

C simulator – *cplants_cyc* – to represent this version of the code (see Chapter 7) and the modification of *cplants* so that R_t would be associated with a plant's age. In Figure 6.11 we present a comparison of the species-area curves obtained from running the two simulators; note that for more confidence, we would need to compare also the species-abundance distributions of the two, as experimented by Bown et al. [23].

6.4.3 Calculating Fecundity

Calculating the number of propagules (seeds) that a plant generated during its reproduction yielded further unexpected information. In [23], this number is calculated based on a storage/fecundity relation (R_f), dictating how much resource a propagule will need; if, for example, a plant disposes of $S_r = 100$ units of resources available for reproduction, and the quantity of resource required by a propagule is $R_{min} = 10$, then it will generate 10 propagules. In Bown et al.'s model, R_{min} depends on the plant's essential uptake, $R_{min} = U_e(1)$. This way, there are two degrees of freedom in the way the number of propagules, per plant, is calculated.

The C code that we obtained contained a different implementation: $R_{min} = 0.6$. This constant value has the (beneficial) side effect of limiting the essential uptake problem (Section 6.4.1). In the eventual case of a $0 < \beta < 2$, $U_e(1)$ is a very small number, tending towards 0; consequently, R_f takes a very high value, tending towards... infinity. Each plant found in this situation, is trying to disperse millions of (almost resource-empty) seeds. The effect was observed due to high simulation times in various experiments: if one such mutant plant exists in the simulation, then each reproductive event it will execute a *for loop* of 10^5 or 10^6 order; provided it successfully disperses some of its seeds, the effect will be multiplied.

The above situation prompted the creation of another simulator branch – *cplants_ct_seed* – maintaining this $R_{min} = 0.6$ implementation, while *cplants* was corrected so that $R_{min} = U_e(1)$. Figure 6.12 details simulation results for these two simulators, emphasising a quantitative difference in results; from a qualitative perspective, the differences are smaller e.g. $z = 0.458$ compared with $z = 0.549$.

6.5 Discussion

This chapter has provided a series of examples for the use of SA (GSN in particular) for supporting rigour in SBR. In the first part (Section 6.1) we stated the obvious: if rigour is not a requirement, then SA can be ignored; we can say that the need for rigour modulates the need for SA. Continuing the theoretical approach, we suggested how SA could be used in SBR. This topic is similar to what was expressed in SCS; for example, in the GSN standard [1, par 0.4.13] it is mentioned that GSN can "improve comprehension amongst the key stake-

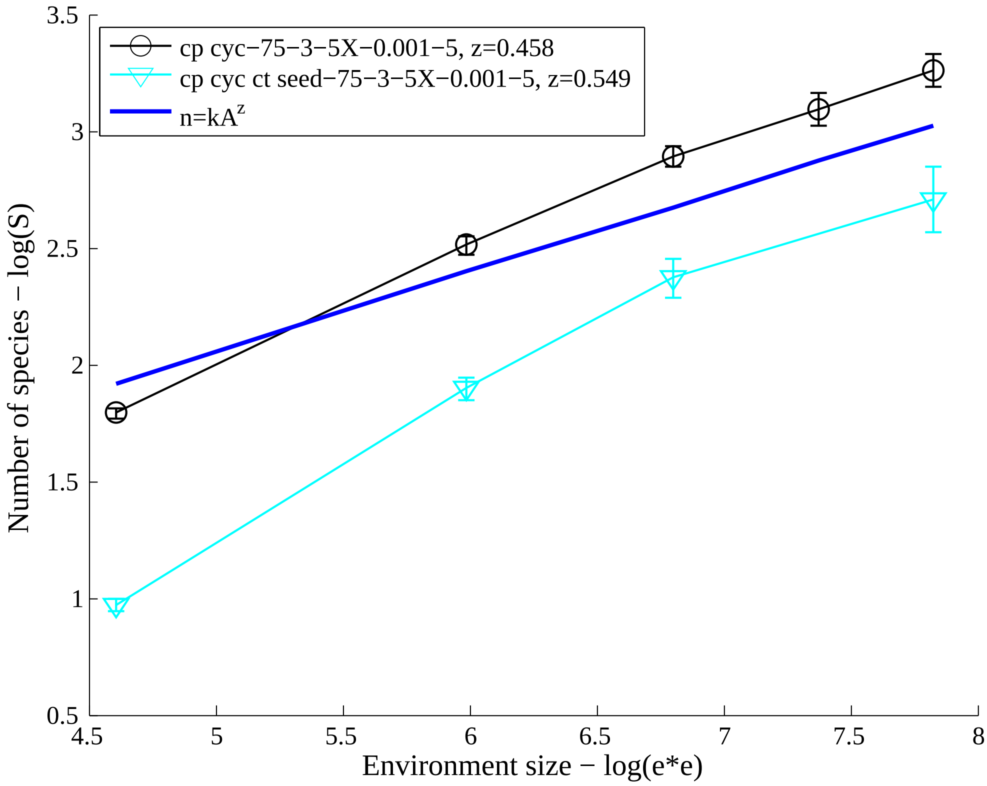


Figure 6.12: Comparison of the species-area curves generated by the *cplants_cyc* and *cplants_cyc_ct_seed* simulators

holders", and we confirmed this in an SBR setting (see Section 6.3.3. The GSN standard continues by expecting GSN to improve "the quality of the debate and the time taken to reach agreement on the argument approaches being adopted". Since this is early days of GSN use in SBR, we can only confirm through the examples detailed above, that GSN use has expanded our understanding of the problems addressed.

In this chapter, we continued the line of retrospective GSN usage initiated in Chapter 5. Having converted Sargent's approach to assuring the validity of a simulation effort, into a GSN pattern, we contrast this with the validity argument obtained from reverse-engineering [23]. Even if sufficient information is present in [23] in order to substantiate a validity argument following Sargent's validity pattern, it is difficult to reuse the arguments obtained in the reverse-engineering process. Even if we were to complete such a task, the result would still represent a version of what Bown et al. [23] actually meant.

In terms of using GSN for ongoing research, we described two case studies: one is the use of GSN argument patterns as a high-level guidance for achieving fitness for purpose (Section 6.3.3), while the second refers to the more technical level of assuring the equivalence of two simulators (Section 6.3.2. Beyound the theory lie practical insights; in Section 6.4 we described unexpected observations concerning Bown et al.'s model and simulation. To finish the argument for adequate equivalence, in practice, is outside our possibilities; only through further interaction with Bown et al. [23] can sufficient confidence be gained both in their simulation and in our replica, as exemplified by the "public correspondence" initiated by Will and Hegselmann [91].

Chapter 7

Assumption Deviation Analysis

"It has long been appreciated that designers make assumptions when developing systems. It is also known that assumptions are not always recognised as such, or are rapidly and casually considered then forgotten. There have been highly catastrophic consequences of this casualness" [81]. This statement epitomises a fundamental difference between SCS and complex systems research: casualness in SCS may have direct catastrophic consequences, whereas a similar conclusion is not drawn with respect to complex systems research – "casualness" may continue unsanctioned, as long as its consequences are not deemed relevant.

This chapter expands on the topic of assumption management in the context of SBR, and introduces a technique for performing deviation analysis over research assumptions. Inspired by work in the field of SCS, where deviation analysis is used for obtaining insights into the possible safety implications [32], the assumption deviation analysis (ADA) adapts the hazard and operability study (HAZOP) [56] to the analysis of SBR assumptions. The author views this as complementary to existing sensitivity or uncertainty analysis techniques.

The chapter addresses the thesis statement by focusing on the relation between the way (degree) in which a model has been studied, the interpretations of its results and the conclusions and contributions being claimed. Having acknowledged the existence of validity evidence, confidence building requires of us to proceed to the following phase: performing a holistic, albeit systematic, assessment of the model uncertainty, in comparison with its interpretation and claims.

The chapter proceeds as follows: Section 7.1 expands on the need for adopting assumption management and focusing on assumption analysis, as prerequisites for rigorous SBR; Section 7.2 details related work; Section 7.3 describes the assumption deviation analysis method that the author proposes, while Section 7.4 exemplifies it through the use of our case study. Section 7.5 concludes the chapter with a discussion.

7.1 Need for Assumption Management and Analysis

Let us assume that a SBR publication describes how a model was validated, through plots comparing empirical data with simulation outputs (the classical operational or black-box validation). Results are visually suggestive of a good match. We may actually be supplied with a more conclusive, statistical analysis of results. Having converged on a *positive* result of this comparison (and provided no unexpected findings, such as those described in Section 6.4, emerge) we are left wondering what the statistical and visual match actually imply: what do they say about the system being modelled? This wondering is both necessary and motivated by the fact that the model belongs to a conceptual world and is only an analogy for the studied system (Section 1.2.1). If we know little about the studied system, how much more do we actually know about our study tools (models, simulations)?

One has to accept that, even after validation, **the model is largely unknown** and to a similar extent, **the simulator is largely unknown** (since the conceptual model is different from the computer simulation [77; 83]), both to the authors and the other stakeholders or readers; the results have been obtained, but not the understanding. Comparing simulators to physical, engineered systems, after a successful validation one can observe that the physical system is behaving as expected, but he could not be able to say exactly why, nor list the components that are critical to its functioning.

This is where methods such as sensitivity and uncertainty analysis [47] come into place. A better insight into the model and simulator's workings may be obtained as the simulator's sensitivity to inputs and outputs is being studied: any test enhances ones understanding of the simulator and, provided the relation between it and the conceptual model has been verified as adequate [83], it furthers ones understanding of the model. These traditional methods, however, determine a holistic, high-level and coarse-grained understanding of the model and do not fully reach the level of individual assumptions and abstractions which compose the model. The sensitivity of the model to its fundamental assumptions should not be treated as of less importance, to its input – output sensitivity.

To further a model's understanding, literature suggests one should consider testing the robustness of model results against competing assumptions (or submodels) [80]; other authors speak of white-box validation [51], which is supposed to test each sub-model at a time. Albeit we find these examples in the literature, there is still one aspect that is unsatisfactory: *in SBR, there is no assumption management*. We can perform a white-box validation, and this will provide insights beyond (or at least different from) those obtained through the classical sensitivity or uncertainty analysis, however, this is still a coarse-grained approach, not dealing with individual assumptions per-se (but with sub-models). If assumptions are not managed, they are not explicitly, nor exhaustively accounted for, consequently how can a finer-grained understanding of assumptions be obtained?

Drawing a parallel from Shore's work [81] in the domain of SCS, concerning the management of safety-related assumptions in safety critical projects, a management of SBR assumptions implies:

1. recognising assumptions;

2. capturing assumptions;

3. classifying assumptions as validity-related or not;

4. validating those that are validity-related;

5. justify that the assumptions management has been performed adequately.

Since there is no explicit management of assumptions in SBR, each of the phases suggested above may be carried out with a degree of subjectivity e.g. "common sense" may dictate which assumptions are deemed relevant for further analysis and which are simply not taken into count. Grimm and Schmolke [38, p. 21] provide an overview into this issue as observed in ecological modelling:

> "most sensitivity analyses presented in the modeling literature only include a subset of all model parameters because modelers tend to 'hide' parameters they never considered for being varied. There can be good reasons for this, for example because the modeler wants to focus on certain scenarios, but these reasons, and all 'hidden' parameters should be documented nevertheless, preferably in a separate table or paragraph."

It is the author's experience that, without performing an analysis, researchers have no more than the 50% chance of predicting accurately the effects and implications of most assumptions; even with respect to their own models, there are no guarantees that authors will have developed an in-depth intuition about each and every non-analysed assumption. How can we classify an assumption as validity related, if we have not tested it?

There is a certain *preference* of researchers, to study a subset of relevant assumptions, while leaving the others for "further work" (if they are declared at all). It is understandable that one research effort may not have the resources for probing *all* assumptions, and it has become the norm that such an objective is not even desirable. Before a model is fully assessed, it has already been modified and used in publishing a different set of experiments; the modified model can be considered a new model altogether and, it can (subjectively) capitalise on the credibility of the original model, it will need to be validated again [77].

7.2 Related Work

Relevant to the topic of assumption analysis are a number of proposals, both from the domain of computer science and ecology. In terms of computer science, Sargent and others have already discussed about conceptual model validation [61; 77]. In the case of a non-complex domain, we can discuss of accurate conceptual models, that can be validated to a large extent e.g. if we are simulating a computer system, all the technical details concerning the functioning of this system are available and the model encapsulates as much information as necessary for its purpose.

However, when researching complex systems, the domain is hardly in the grasp of the modeller: scientists discuss (and debate) about emergence and downward causation [84], two processes describing the interaction between lower and higher level of organisations within complex systems; systems (e.g. natural organisms) are not isolated, but exist and act within an environment; drawing a line between what is relevant or not, between what is interacting with and influencing a given system or not, is a choice rather than a certainty. In such a context, the concept of (hard) validation is elusive. Jakeman et al. [51] prefer to avoid the use of this term, in favour of that of model *performance*.

Looking back at Figure 6.1 describing, in a GSN format, Sargent's view on what constitutes a valid simulation effort, we observe that the conceptual model validation is supported by a claim that "theories and assumptions underlying the conceptual model are consistent with those in the system theories" (G CM_Val.1). Such an unqualified claim implies that *all* theories and assumptions should have the desired consistency, but then, **no SBR effort has kept track of *all* its assumptions**, nor is it usually possible to build a model out of only "consistent" assumptions. Incomplete knowledge, together with the fact that "all models are wrong, but some are useful" [24], imply that not everything in a model should or can be consistent with an existing theory.

Turning towards the ecological domain, there are several techniques of relevance here: pattern-oriented modelling (POM) [42], Pareto Optimal Model Assessment Cycle (POMAC) [73], Van Nes and Scheffer's strategy to improve the contribution of complex simulation models to ecological theory [86], etc. While the first two techniques allow the refinement of a model based on assessing it against multiple patterns (POM) or criteria (POMAC), Van Nes and Scheffer's method, aimed at improving the understanding of the way in which complex models generate their results, relies on the analysis of simplified variants of a model.

All of the above techniques, together with sensitivity and uncertainty analysis methods customised for complex systems models [47], allow us to obtain a better understanding of a model. What is missing though is that, even if we have simplified our model through an approach like Van Nes and Scheffer's, or we have obtained an added confidence in the model's realism through a process like POM or POMAC, **we still have not obtained an in-depth understanding of each individual assumption, without justifying why this level of understanding is not needed**. In SCS terms, even if our engineered system has passed its certification tests, we have not investigated the potential hazards that might affect it; we have not performed a hazard analysis in a systematic and sustained manner.

Consequently, we investigate the use of deviation analysis for obtaining insights into model assumptions.

7.3 Assumption Deviation Analysis (ADA)

This assumption analysis approach makes use of deviation analysis in general and HAZOP in particular. HAZOP builds on critical examination – "a formal technique for examining an activity and generating alternatives by asking 'What is achieved?', 'What else could be achieved?', 'What should be achieved?', 'How is it achieved?',... 'When is it achieved?', 'Where is it achieved?'... 'Who achieves it?', and so on." [56]. Instead of maintaining the focus on generating alternatives, HAZOP generates deviations. In practice, this implies analysing "what if" scenarios (deviation analysis), in corroboration with a number of keywords: no or none, more, less, as well as, part of, other than, reverse. For each component of a critical system, a number of at least 7 scenarios (one for each question – keyword combination) can be *analysed* by a HAZOP team, with the aim of identifying potential hazards.

In the context of SBR, the author proposes a customized approach for performing ADA, comprising of the following steps:

1. capture model assumptions, from high-level to low-level;
2. formulate deviation keywords;
3. create deviation tables comprising all components;
4. evaluate deviations and identify the ones that:
    (a) render the model unusable;
    (b) require further investigation;
5. generate simulator variants for assessing deviations;
6. complete the deviation tables with results obtained from simulator variants.

ADA can be applied to low-level assumptions or to high-level design details. In the latter case, one could assess the eventual outcome of not using a spatial representation at all e.g. [13] and [21], or of using a different type of spatial layout e.g. a tile layout instead of the more popular grid-based layout [20]. In this study, the focus is placed on low-level assumptions – the ones that usually are left unanalysed.

7.3.1 Capture Model Assumptions

To start with, we draw a parallel between the *components* of a safety-critical system and a model's *assumption*; engineered systems are built with components, whereas models are built from assumptions. Although assumptions are of different types e.g. simplifications, distortions, hypotheses, "facts", we generalise them to: all (critical) pieces of information defining a model, without which the model would not operate. Such a component-oriented view on assumptions makes way also for a better accountability of assumptions and, in turn, for converging towards a type of *assumption management*.

To obtain a thorough list of model assumptions, we can use ODD [40]. If we possess an ODD description which is sufficient in enabling us to reimplement a simulator, this contains the information needed for ADA.

In line with Shore's proposal for the use of an Assumptions Log [81], the author proposes here an adaptation of this tool, to be used in SBR. Table 7.1 details the information this Assumption Log for SBR should capture, whereas Table 7.3 provides an example of filled in Assumption Log for SBR. Each entry (assumtion) consists of four types of information: assumption statement, its impact, validity and scope. The (expected or evaluated) impact of an assumption is a synonym for how sensitive the model is to it: the greater the sensitivity, the greater the impact[1]. The Validity field captures the expected or evaluated validity of the assumption. In relation to domain knowledge, researchers know in advance if some assumptions are grounded (supported by existing evidence) or if they contradict commonly accepted knowledge (hence are invalid). The Scope field records how the assumption influences the scope of the model.

[1]By altering a critical assumption, one is sure to expect a significant change in simulation results and vice-versa.

	Information Field	Description
1	Assumption	statement of the assumption
2	Impact	assessment of the assumption's impact over the model's outputs e.g. Low, Critical
3	Validity	assessment of the assumption's validity e.g. Grounded, Invalid, N/A (not to be assessed)
4	Scope	the effect of the assumption over the model's scope e.g. (-) model not relevant for studying

Table 7.1: "Assumptions Log" fields

7.3.2 Formulate Deviation Keywords

Following HAZOP, ADA requires a set of keywords to guide the analysis. To begin with, it is satisfactory to use the standard HAZOP terms listed above, whereas in the future, experience might allow us to propose other, SBR specific terms.

7.3.3 Create Deviation Tables

The next stage is to arrange the assumptions and keywords into a HAZOP-like table and begin assessing the deviations. The author calls this type of table, "Deviation table" and its information fields are described in Table 7.2.

The impact of some deviations, on the overall model, may be obvious and hence immediately written in the appropriate table cell. Deviations that are considered critical should be highlited e.g. using red text colour. In the remaining cases, however, further simulation is required and this can be marked with a '?' symbol. Tables 7.4 and 7.5 provide an example of filled in Deviation Table.

	Field	**Description**
1	Assumption	assumption being analysed
2	No/None	impact if assumption is removed from model
3	More	impact if assumption is quantitatively increased
4	Less	impact if assumption is quantitatively decreased
5	As well as	impact if assumption is qualitatively increased
6	Part of	impact if assumption is qualitatively decreased
7	Other than	assumption substituted

Table 7.2: “Deviation Table” fields

7.3.4 Generate Simulator Variants

Irrespective of the keyword, there may be ‘assumption + keyword’ combinations for which no pertinent, simulation-free answer can be given. We need then to develop a systematic framework for analysing these combinations. The author proposes the use of a version control system e.g. Subversion (SVN), Git with which to track the development of simulator variants. These are branched from the main repository and differ from the origin by implementing a single deviation. Outputs collected from the simulator variants can be stored in a database, where each result entry can be related to the simulator variant (branch) and a commit identifier – this way we assure the fact that both source code and outputs match the investigated deviation.

Through experimentation, we will be able to fill the missing gaps in the HAZOP table. By making the entire source code repository (with its original simulator and variants) available, together with the results database, we open our work to external scrutiny.

At the end of this analysis, we will have obtained a thorough understanding of the areas of vulnerability in the model (the hazards), in addition to having sufficient evidence for substantiating that we *know* the model. To know a model means to be able to say what is the particular effect (or influence, role) of each of its constituting assumptions, similar to knowing an engineered system means to be able to describe the role of each of its components.

7.4 Case Study

According to the above method, the first stage is to list Bown et al.'s [23] modelling assumptions, followed by the execution of the deviational analysis.

7.4.1 Capture Model Assumptions

Listing assumptions is not a straightforward task. Firstly, because in a publication it is often the case that not all assumptions are described, whereas from a theoretical perspective, the computer simulation contains more assumptions than the conceptual model, because it represents a reification of it (to be executable, it requires concrete information not described within the conceptual model [83]).

Consequently, the author used the structured information presented in Section 4.2 – which approximately follows the ODD protocol. In order to fill the assumptions log, an information elicitation session was performed with Bown and Eberst (first and co-author on the case study paper [23]).

The information elicitation session yielded information that was not entirely presented in [23]; this because [23] was not written according to a documentation standard (such as ODD), and because the author specifically asked questions about the justifiability of different modelling decisions (questions that perhaps the reviewers of [23] had not asked). At the end of this effort, we are in possession of two types of information: Bown et al.'s model description and the elicited information presented above.

This information, together with what was published in [23], is then "compiled" into a unique set of assumptions are recorded in the Assumptions Log (Table 7.3).

	Assumption	Impact	Validity	Scope
1	competition model	?	Assumed	(-) model not relevant for settings in which plant facilitation takes place
2	only one plant per location	?	?	(-) model not relevant for plants that share resources across cells
3	clonal reproduction	?	Invalid	(-) model contradicts the biology
4	no seed banking	?	Invalid	(-) model contradicts the biology
5	plant composed of three resource compartments	?	Grounded	(+) consistent with prevailing knowledge

Table 7.3: Fragment of the Assumptions Log for Bown et al.'s [23] model

7.4.2 Create Deviation Tables and Perform Initial Evaluation

We devise different HAZOP-like tables for different types of information. We can have a table for each type of entity (in this case, plants and spatial locations) or one for an aspect of the model's description (e.g. one for the plant genotype, including the plant traits and another for its phenotype).

As we have mentioned before, at the end of this phase, each "component" – assumption without which simulations cannot be performed – needs to appear in a HAZOP table.

The first phase of the analysis sees deviations for which an immediate answer is readily available, have it written in their table cell; Tables 7.4 and 7.5 present an example of a deviation table, after this first phase. Table 7.6 captures the deviations that may render the model unusable (recognised hazards).

Deviations that cannot be readily addressed (marked with "?" in the tables), will be evaluated through simulation.

7.4.3 Generate Simulator Variants

A list of simulator branches to be developed results, as shown in Table 7.7. Here each simulator branch receives a unique (encoded) name, and a description of the difference between it and the main simulator.

Through experimentation with each simulator branch, we obtain answers for the HAZOP questions (and insights into the original assumptions forming the model). In Figures 7.1 and 7.2 we present simulation results concerning two *cplants* branches. The first figure suggests that the way plant age is defined in Bown et al.'s model [23], does not have a qualitative or quantitative influence over results – the model is not sensitive to it. The model is, however, sensitive to the assumption that plants have a development stage and to the way the number

	Trait	No/None	More	Less
1	$U_e(s)$	1) fixed uptake 2) *no uptake*	greater resource need	lower resource need
2	$U_r(s)$	loss of surplus resources	excessive request	*request below need*
3	$D_u(s)$	1) fixed distribution 2) *no uptake*	more types of uptake	a single type of uptake
4	P_s	1) constant ratio 2) *no allocation*	*no reproduction*	underdeveloped plants
5	r_t	less resources for reproduction	more resources for reproduction	less resources for reproduction
6	r_u	no surplus	greater release	smaller release
7	R_t	no time dependent fecundity	less frequent reproduction	frequenter reproduction
8	R_d	no development dependent fecundity	?	?
9	R_f	?	?	?
10	D_p	*no seed dispersal*	more elaborate model	simpler model
11	V_t	no survival threshold	*quicker death*	*immortal plants*
12	V_p	no survival assessment	*immortal plant*	*quicker death*
13	P_d	no random death	high r.d. probability	low r.d. probability

Table 7.4: Deviation table, addressing the plant traits

	Trait	As well as	Part of	Other than
1	$U_e(s)$	?	?	?
2	r_u	?	?	?
3	$D_u(s)$	? ?	? ?	gradient update
4	P_s	?	?	?
5	r_t	?	?	?
6	r_s	?	?	?
7	R_t	?	?	?
8	R_d	?	?	?
9	R_f	?	?	?
10	D_p		?	?
11	V_t	?	?	?
12	V_p	?	?	?
13	P_d	more elaborate model	simpler model	? ?

Table 7.5: Deviation table, addressing the plant traits (continuation)

	Deviation	**Effect**
1	No $U_e(s)$	plants die without resources
2	Less $U_r s$	plants would starve to death
3	No $D_u(s)$	plants die without an uptake area
4	No P_s	plants die without resources
5	More P_s	plants allocate no resources for reproduction
6	No D_p	no new plants get created without seed dispersal
7	More V_t	plants need more resources for survival
8	Less V_t	plants need extremely few resources for survival
9	More V_p	resource shortfall may be assessed too late
10	Less V_p	resource shortfall assessed instantly

Table 7.6: Deviations that affect the model's usability

No	Simulator Branch	Description
-	cplants	main trunk
1	cplants_cyc	reproduction based on global timer
1.1	cplants_cyc_ct_seed	constant seed production
1.2	cplants_cyc_no_age	no plant ageing
1.3	cplants_cyc_no_dvs	no development stage
1.4	cplants_cyc_no_indvar	no individual variation
1.5	cplants_cyc_no_res	no resources
1.6	cplants_cyc_var_seed	variable seed production
1.7	cplants_cyc_diff_alloc	different allocation algorithm
-	*cplants_no*	features switched off
2.1	cplants_no_age	no plant ageing
2.2	cplants_no_alloc	no resource allocation
2.3	cplants_no_death	no plant deaths
2.4	cplants_no_dvs	no development stage
2.5	cplants_no_indvar	no individual variation
2.6	cplants_no_res	no resources
2.7	cplants_no_shortf	no resource shortfall
3	cplants_seedb	with seed bank
4	cplants_seqord	sequential ordering of plant events
-	*cplants_ue*	essential uptake variations
5.1	cplants_ue_ct	constant essential uptake
5.2	cplants_ue_lin	linear essential uptake
5.3	cplants_ue_rev	reversed essential uptake

Table 7.7: Example of assumption variations tested on the cplants simulator

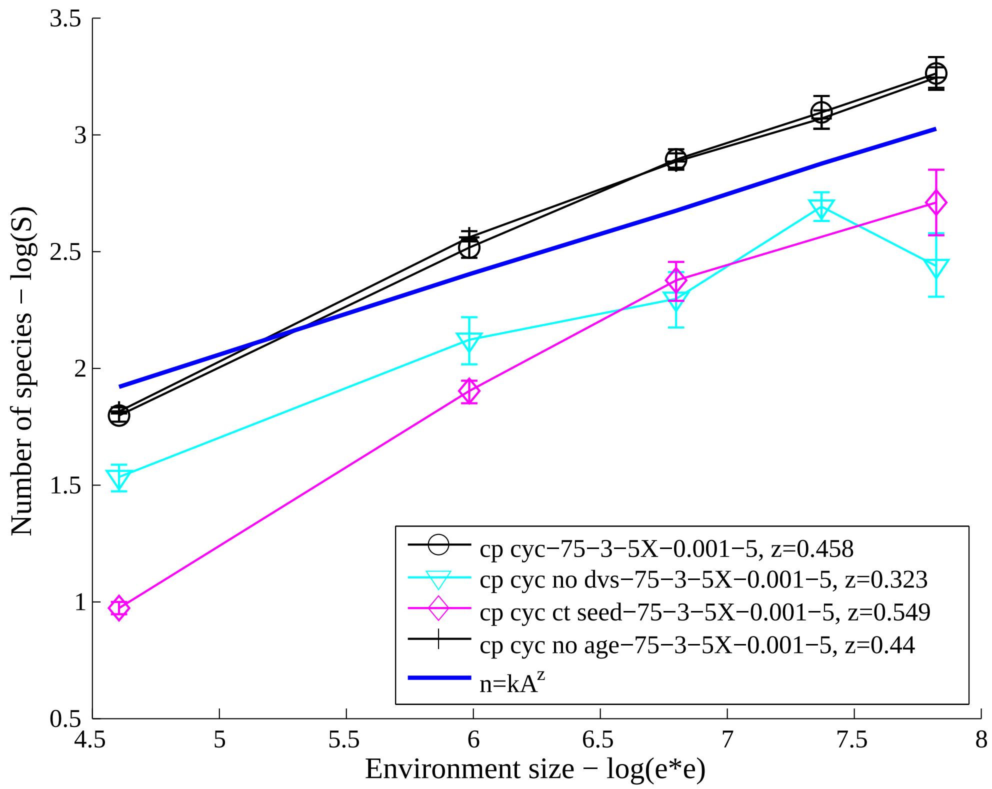


Figure 7.1: Comparison of *cplants_cyc* branches, at default parametrisation

of propagules is calculated (see Section 6.4.3).

In Figure 7.2, we observe the role of individual variation (another assumption of Bown et al.'s model [23]). If it is switched off, the resulting species-area curve is qualitatively and quantitatively different from the expected one (corresponding to the mathematical model $n = kA^z$). We confirmed this result by using a secondary simulator – *nplants* – developed in NetLogo, hence both results are depicted.

More results generated from this assumption deviation analysis are provided in Appendix E. We observe there that assumptions that were initially assumed as invalid e.g. no seed banking, do not have a significant impact over the simulation results. If an invalid assumption does not alter simulation results once it is removed or changed, then the model is robust to it and the assumption can be dropped or the model needs to be further investigated in order to understand the

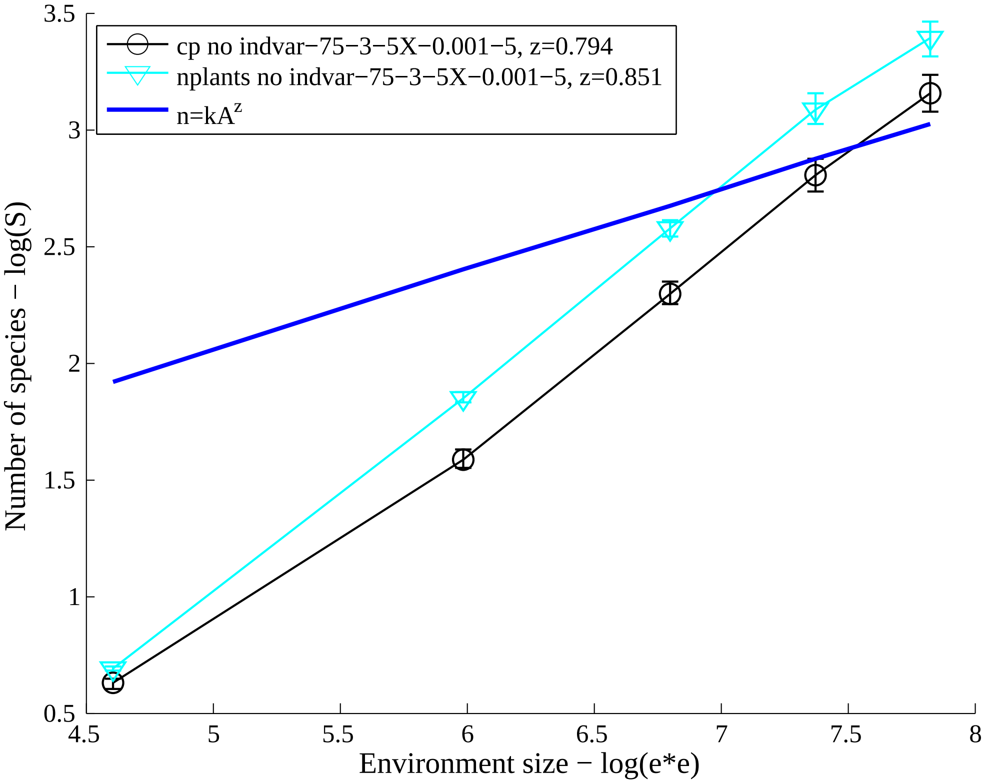


Figure 7.2: Comparison between the *no_indvar* branch (2.5 in Table 7.7) of the *cplants* and *nplants* simulators

reason why it was insensitive to the assumption.

7.5 Discussion

This chapter introduce the ADA technique to the domain of SBR. While we have not provided here extensive results, spanning the entire description of Bown et al.'s [23] model, we expect the method to be scalable and effective in expanding our understanding of simulation models.

The chapter has addressed related work, from the domain of computer science and ecology. There exists numerous sensitivity and uncertainty analysis techniques, as well as methods for converging on minimal model definitions that maintain validity. None of this methods, however, delivers the same insight as ADA. Through systematic listing of assumptions we take the first steps towards assumption management. Through the keyword-based deviational analysis, we gain extensive insights into the role, influence or importance of each assumption.

We can also calculate the coverage of this assumption analysis. Provided we identify A assumptions and we use a set of K keywords, we have $N = AK$ cases to analyse. Out of this, a percentage P_i will be answered immediately, another percentage P_s will be addressed through simulation, whereas the remainder $P_?$ will not be addressed. At the end of the exercise we can quantify the unknowns (the uncharted assumptions), through $P_?$. There is a different confidence resulting from a scientific claim made in the context of a model that has been analysed in a "ADA proportion" of 90%, out of which $P_s = 85\%$.

Chapter 8

Conclusions

8.1 Methodological Advancements

This thesis mentions three methodological advancements:

1. Reverse-Engineering Publications into GSN Arguments

Chapter 5 describes a literal (brute force) but not-biased reverse-engineering method through which argumentation threads from published research can be captured into GSN arguments. This method is not seen as optimal, nor as the most efficient in extracting structured arguments from literal text, but it serves the purpose of showing as objectively as possible, that unless a research article is written with structured argumentation in mind (and in this case, there should be no reason why the underlying structured arguments cannot be delivered as supplementary information, in order to facilitate review and understanding of the work), researchers will continue complaining about the quality of published work, of peer reviewing or of the lack of assurance giving information.

2. Using Structured Argumentation for SBR Purposes

Chapter 6 details a number of uses for SA in the context of SBR; this applies both to ongoing research and to already published research. SA can be used for assuring the adequacy of SBR, at finer-grained or higher levels of description.

3. Applying Deviation Analysis to SBR Assumption

Chapter 7 proposes a method for performing deviation analysis on modelling assumptions. Derived from HAZOP and taking into account the concept of assumption management and the way it is performed in SCS, the method may be used in yielding a better, more in depth understanding into the criticality, validity and effect of each model assumption, irrespective of its granularity.

8.2 Contributions

The first contribution of this thesis is its actual focus: that of increasing the rigour level of SBR, through the use of proven techniques from the domain of SCS. A successful uptake of the research provided in this thesis can make an impact on a wide scale, as SBR is not limited to a domain, but it is inter- and cross-disciplinary.

Secondly, the thesis provides a detailed case for the use of structured argumentation in scientific (and simulation-based) research. In line with "evidence-based" interests, such as those taking place in engineering (through evidence-based engineering), medicine (evidence-based medicine), the use of GSN for SBR purposes has the potential of consolidating the evidence-based approach that should constitute the very foundation of research. Evidence alone is insufficient, however, as it has been shown in SCS. Either due to sheer volume, or due to its opacity, those that want to make use of it need to provide also clear arguments using this evidence.

The thesis has addressed this topic from two perspectives: a retrospective and a dynamic one, GSN can be used for retrospective purposes such as capturing the argument threads of different publications; eventually, arguments constitute an essential layer that is not explicit in an abstract, nor in a full paper format. This type of information is inaccessible, it is not indexed in data bases (although attempts to maintain argument data bases have been made, for other domains) and consequently the research community cannot benefit from them.

We have shown why GSN is not perfectly suited for capturing scientific arguments; in comparison with debate-mapping tools such as Debategraph, GSN is powerful but also limited in the type of relations between information elements it allows. Extracting GSN arguments from plain-text papers has also proved to be a difficult and only partially successful exercise. Without the support of the original authors, it is simply not possible to achieve this in full; the plain text format allows for the use of a persuasive rather than a logical language; ambiguities prevent converging towards a single GSN argument for a given piece of text.

We have also shown how GSN has been used in ongoing research. It is important to have both these types of uses well represented. GSN allows researchers to establish the basis of their confidence, "there and then"; it also facilitates retrospection and reaching agreements in research teams. What GSN cannot do is solve any modelling or research problem. It is still the researchers that, through the careful use of structured argumentation, can perform research of higher rigour.

8.3 Further Work

8.3.1 Better Reverse-Engineering of Articles into GSN Arguments

In Chapter 5, the author used a "brute force" reverse-engineering method, in order to convert a research article [23] into structured arguments. A principal purpose for attempting to put the entire content of the article, sentence by sentence, into GSN format, was to emphasize that when one attempts to lose no content from a research paper (to apply no heuristics, to make no subjective decisions, etc.), the resulting structured arguments are far from the traditional tree-structured GSN arguments as seen in Safety Critical Systems [54]. In other words, the value generated by the researchers (authors) themselves creating and publishing their own structured arguments, is more or less irreplaceable.

To go beyond the above effort, a more refined method of extracting arguments out of plain text would be needed. That's what we normally do when we read an article. The questions that can be asked here are: can we develop a reliable method of extracting readable (rather than comprehensive, like those generated in Chapter 5) structured arguments out of research articles and could this method be fully automated? The benefits of a positive answer are worth considering: the science community could develop argument databases, to complement the Abstracts databases, so that anyone could have a "quick look" through the core argument of a research paper, rather than just its abstract; complicated or just not easy to read papers would still be summarised in a structured argument that would clearly show what is claimed and how are the claims supported.

8.3.2 ODD, TRACE and SA

Following ODD, another documentation standard was developed in the field of ecology: transparent and comprehensive ecological modeling (TRACE) [80]. While ODD is limited to structuring a model's description, TRACE is aimed at structuring the modeller's notebook, hence encompassing the model development phase, its testing and analysis, together with considerations on its applicability for its intended purpose. Together with ODD, TRACE is considered to provide a complete set of templates for organising research data, as a complement[1] to plain text research articles.

It is the author's opinion that, without SA, the package composed of ODD and TRACE is highly useful, but still incomplete. While TRACE records information concerning a model's verification and validation and requires (through the "Rec-

[1] Or supplementary data.

ommendation" section) the "description of how initial question(s) could be answered" based on the obtained results, there is still a missing element: how the researcher bridges its conclusions with all the other aspects documented in TRACE (problem formulation, design and formulation, model description, parametrization, calibration, verification, sensitivity analysis, validation, results, uncertainty analysis). This is exactly where structured argumentation is needed, as through SA the researcher provides justifications for its conclusions.

Consequently, a line of research should be dedicated to studying the use of ODD, TRACE and SA (through GSN) together, for achieving a higher level of rigour in research.

8.3.3 Tools and Notations for SA

This thesis has emphasized the utility of GSN for scientific purposes; it has also suggested that the GSN standard is insufficiently adequate to SBR. The *Part-SupportedBy* relation we proposed, for example, finds echos in Ge et al.'s effort of proposing a looser meaning for *SupportedBy* [35]. The thesis proposes a set of other relations, but we need to establish a semantic for each of these in order to increase their applicability.

In addition, the topic of tool support is also relevant. For generating diagrams, the author used Microsoft Visio, with a free plugin developed at the University of York installed. There are other commercial tools available, but these either come at a prohibitive price, or do not allow the flexibility we would require in a research environment (there where GSN can be used beyound its standard). Other free tools, such as NASA's CertWare (based on the Eclipse IDE), are still steps behind even complying with the GSN standard, although being open-source, software engineers could modify it in order to achieve the flexibility we have discussed above.

Paul Andrews, researcher on the R-Futures project[1], within the University of

[1]http://r-futures.ecs.soton.ac.uk/

York, is working on an online GSN tool. While this is still under development, it offers the benefits of platform independence, lack of costs and, of course, wide accessibility. To use it for research purposes, however, issues related to the security and usability of this tool need to be addressed.

8.3.4 Model arithmetic

The path from general to specific is paved with requirements for gaining complexity (or "realism") through replacement of simplifying assumptions with modelling details; there exists a trade-off between generality and specificity. Each removal or replacement of a model assumption leads to the creation of a model variant, such that we can even speak of an arithmetic for models. If $M = A + P$ represents the necessary formula for obtaining a model, where M represents a model, $A = \{a_i | i \in N\}$ is a set of assumptions and $P = \{p_i | i \in N\}$ is a set of parameters, we can then define classes of model variants:

$$
\begin{aligned}
M_1 &= M + (\overline{a_i} - a_i) \\
M_2 &= M + (\overline{a_i} - a_i) + (\overline{a_j} - a_j) \\
&\vdots \\
M_n &= M + (\overline{a_i} - a_i) + \cdots + (\overline{a_t} - a_t)
\end{aligned}
$$

where $\overline{a_i} = \{null, \neg a_i, \text{etc.}\}$ represents the substitute of an assumption a_i. The difference between two adjacent model classes should be one assumption: $M_n - M_{n-1} = 1$. An instantiation of a class M_n model is n assumptions away from an instantiation of an initial model M.

There are three cases of modifications: a) assumption removal (when $\overline{a_i} = \text{null}$, b) assumption replacement, when $\overline{a_i}$ is instantiated e.g. the negation of a_i and c) assumption addition, when no assumption is removed $a_i = \text{null}$ but a new assumption $\overline{a}$ is added to the model. From this perspective, practically any two

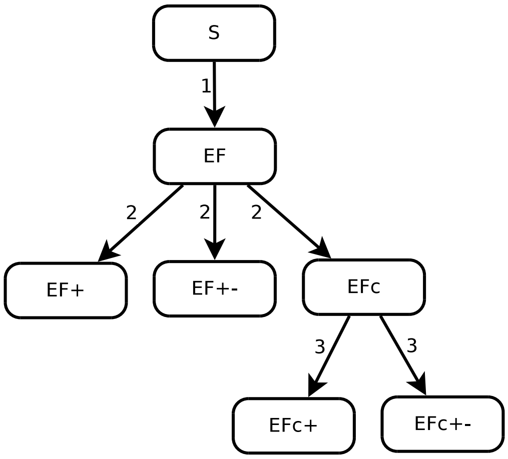


Figure 8.1: Model variant hierarchy, starting from Schelling's initial model (S), continuing with the three variants presented in [30] (we shall name them EF, EF+ and EF-) and finishing with three variants defined in this paper (EFc, EFc+ and EFc+-). The numbers attached to the edges represent the total assumption distance from the root model.

models are located at a discrete distance from one another.

Varying the parameter set scopes the thoroughness with which a model can be explored, more parameters enabling a more thorougher exploration. While assumptions represent the invariant part of a model, parameters open up a model to external variance. This because they are in direct relation with the model's assumptions.

8.3.5 A bio-inspired view on simulation-based research

A model can be seen as a genotype composed of assumptions and parameters. This genotype is at the basis of a *simulator* – the phenotype, formed according to the model, the engineering technology adopted and the type of data it needs to deliver. This organism-like entity acts in an informational environment: it consumes data i.e. its parametrisation and additional data inputs, and outputs data i.e. result sets. Figure 8.2 further depicts this image.

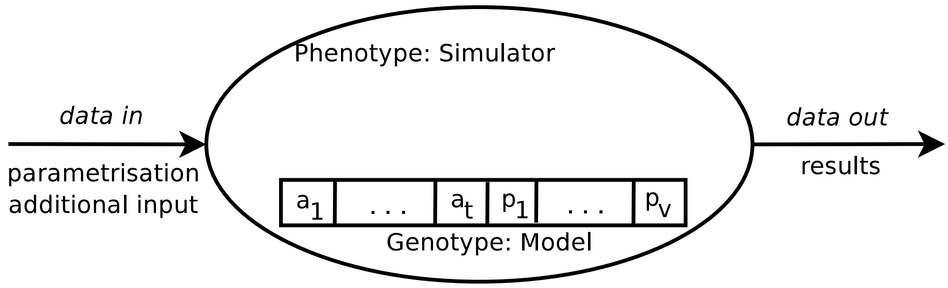


Figure 8.2: Bio-inspired view on simulation-based research.

We can imagine now populations of these genotypes (models) being evaluated on behalf of their phenotypes (simulators). This is, to an extent, what happens in a simulation-based research project, where a family[1] of models is usually developed, but only a selected set deliver the required solution. We can also consider the type of fitness that such an individual has: traditionally, this is an objective function i.e. $f(P, V_c, V_d)$, where P represents the purpose of the model, V_c represents a set of validation criteria and V_d represents validation data; alternatively, it may depend on other individuals e.g. co-evolution of models, model cross-comparison.

In a population of models, each individual genotype is unique. It does not make sense to have two identical genotypes in the population, albeit having more phenotypes of the same genotype is useful in gaining more confidence in the model's features (through result replication).

Scientific literature abounds in research performed through two types of models: equation-based and agent-based. These two types of (meta) models differ through the nature of assumptions they are usually built from, but they still conform to the image depicted in Figure 8.2. While agent-based models (ABMs) are directly associated with computer simulators, equation-based models (EBMs) may also be embodied in computational phenotypes (solvers) facilitating their evaluation.

If this bio-inspired view is accepted, then we can take a step forward and focus on the importance of each element in the genotype. Biological entities have a significant degree of redundancy in their genomes and the actual mechanisms and

[1] A set of model variants

pathways of gene expression are still a topic of study. Techniques such as DNA sequencing and gene knockouts have revolutionised our understanding of living organisms. Albeit simpler and "wrong", models (genotypes) also deserve thorough understanding. To this date, we are not aware of a simulation study that goes all way to fully analysing a model (by analysing its constituting assumptions).

Appendix A

ODD Specifications

At the moment there exist two specifications of ODD, introduced in two publications: the first one, dating from 2006 [40] and the second one from 2010 [41]. The newer specification, making little structural changes, was introduced by its authors as a replacement for the 2006 version.

In our view, the second publication, while bringing more explanations as to how the ODD format should be filled, does not completely superseed the first publication. In cases, the new format simplifies some elements, explicitly narrowing down their scope e.g. the Purpose is required to list only the modelling objectives, the Process overview and scheduling should only define the process names and not go into their details. Other elements are extended e.g. more standard complex adaptive systems (CAS) concepts are added to the Design Concepts element.

Both versions of ODD combine rigorous specifications with looser, vaguer ones. Rigour can be found in the prescribed sequence of ODD elements (their order and labels should be preserved), whereas looser specifications can be found in elements such as the Submodels, which should provide all modelling details, including assumptions; ODD does not provide support, however, on how to efficiently deal with the problem of assumptions (Section 1.2.3).

To elucidate ODD specifications, we use tables A.1, A.2 and A.3, where we detail its content and structure specifications, as described in the 2006 and 2010 versions. Table A.1 reflects the core information for building an ODD model description, while Table A.2 lists what we consider to be the assurance requirements, explicitly included in the ODD specifications. In addition, Table A.3 details the structure specifications of ODD. A good ODD model description should contain all information required by tables A.1 and A.2, according to the structure specified by table A.3.

From the content specification in Table A.1 we observe that the 2010 format:

- simplifies the Purpose element: now it is only required to summarise the model's objectives;
- asks explicitly that ODD description relates model units to real ones (if possible);
- introduces three new design concepts, while removing one.

Table A.3 shows that the 2010 format:

- renames two ODD elements;
- clarifies the scope of the Purpose element, reducing the amount of information it should contain;
- clarifies and scope of the Process overview and scheduling element, reducing the amount of information it should contain;
- describes how new concepts can be included in the Design concepts element;
- clarifies what Input data should contain;
- explicitly asks for explanation and justification of design decisions.

ODD format	**Content Specification**
1. Purpose	
2006	a) *why* you need to build the model? b) *what* will it be used for?
2010	a) summary of *objectives* for which the model was developed
2. (Entities) State variables and scales	
2006	a) higher-level *entities* b) *full* set of state variables and their *units*[1] c) temporal *scale* i.e. length of time steps and time horizon d) spatial *scale*[2] and *extent* of the model world
2010	+ what the model *units* represent in reality
3. Process overview and scheduling	
2006	a) environmental and individual *processes* and their b) *effects* c) type of state variable *update* and d) *time* representation e) *scheduling* order of/within model processes
4. Design concepts	
2006	emergence, adaptation, fitness, prediction, sensing, interaction, stochasticity, collectives, observation
2010	+ basic principles, objectives, learning - fitness
5. Initialization	
2006	a) *creation* of environment and initial population b) *initial values* of state variables and c) their *source* d) *variation* of initialisation across simulations
6. Input (data)	
2006	a) imposed dynamics of certain state variables b) input data files and c) seeds of RNGs
7. Submodels	
2006	a) detailed *presentation and explanation* of each process b) mathematical "skeleton" of the model c) full model description

Table A.1: ODD content specifications. The table collates specifications from the 2006 and 2010 ODD formats, according to each of the 7 ODD elements. For the 2010 format, a '+' symbolises that the specification came in addition to the 2006 one, whereas a '-' symbolises that the 2010 specification does not contain that 2006 item.

ODD format	Assurance Specifications
(Entities) State variables and scales	
2006	reason for choosing *scales* should be explained
Process overview and scheduling	
2006	basis for *scheduling decisions* should be described
Initialization	
2006	references to initialisation data should be provided
Input	
2006	[optional] provide input files, including the RNG seed
Submodels	
2006	each parameter and equation is verbally explained, in full detail what *assumptions* underlie equations and rules how were submodels *tested* and *calibrated*

Table A.2: Explicit assurance specifications in ODD

ODD format	Structure Specifications and Validation Criteria
2006	order of the 7 elements must be maintained all 7 elements must be documented
2010	+ names of the 7 elements must be maintained + ODD description referred via "the model description follows [...]"
1. Purpose	
2010	+ do not describe anything about how the model works + should be complete and understandable by itself
2. (Entities) State variables and scales	
2006	state variables are not auxiliary[1] or aggregated variables if many, variables can be listed in tables or UML class diagrams
2010	+ state variables should not be computable from other state variables
3. Process overview and scheduling	
2006	visualisation via flow chart diagrams is allowed
2010	list only the (self-explanatory) names of the model processes *scheduling* details the order in which processes are executed and in which each process is executed by individuals
4. Design concepts	
2006	concepts that do not apply can be omitted order of design concepts is not compulsory
2010	+ new design concepts can be included; they should have a short name, be presented at the end and be marked as non-standard ODD
5. Initialization	
6. Input (data)	
2010	+ environmental variables should not be affected by model variables + should not define parameter or state variable values
7. Submodels	
2006	if too large, can be included in online appendices or published separately
2010	+ design decisions should be appropriately explained and justified

Table A.3: ODD structure specifications (rules)

Appendix B

Surveyed SBR Articles using ODD

Bibliography

[1] Chlo Deygout, Annick Lesne, Fabien Campillo, Alain Rapaport, *A multiscale approach for biofilm modelling: application for growth in a plug flow reactor*. http://hal.archives-ouvertes.fr/hal-00604377/, 2011.

[2] Els Ducheyne, Martin Lange, Yves Van der Stede, Estelle Meroc, Benoit Durand, Guy Hendrickx, *A stochastic predictive model for the natural spread of bluetongue*, Journal of Preventive Veterinary Medicine, Volume 99, Issue 1, 1 April 2011, Pages 48-59,

[3] Sprouffske K., Pepper J.W., Maley C.C, *Accurate Reconstruction of the Temporal Order of Mutations in Neoplastic Progression*, Cancer Prev Res (Phila). 2011 Jul;4(7):1135–44.

[4] Rebaudo F., Crespo-Prez V., Silvain J.F., Dangles O., *Agent-Based Modeling of Human-Induced Spread of Invasive Species in Agricultural Landscapes: Insights from the Potato Moth in Ecuador*, http://jasss.soc.surrey.ac.uk/14/3/7.html

[5] Xiaodong Chen, Frank Lupi, Li An, Ryan Sheely, Andrs Via, Jianguo Liu, *Agent-based modeling of the effects of social norms on enrollment in payments for ecosystem services*, Journal of Ecological Modelling, Volume 229, 24 March 2012, Pages 16-24

[6] Bing Zhang, Yongliang Zhang, Jun Bi, *An adaptive agent-based modeling approach for analyzing the influence of transaction costs on emissions trading markets*, Journal of Environmental Modelling & Software, Volume 26, Issue 4, April 2011, Pages 482-491

[7] F.E. Bert, G.P. Podest, S.L. Rovere, A.N. Menndez, M. North, E. Tatara, C.E. Laciana, E. Weber, F.R. Toranzo, *An agent based model to simulate structural and land use changes in agricultural systems of the argentine pampas* Journal of Ecological Modelling, Volume 222, Issue 19, 10 October 2011, Pages 3486-3499

[8] Pepijn Schreinemachers, Thomas Berger, *An agent-based simulation model of humanenvironment interactions in agricultural systems*, Journal of Environmental Modelling & Software, Volume 26, Issue 7, July 2011, Pages 845-859

[9] Lei Gao, Bohdan Durnota, Yongsheng Ding, Hua Dai, *An agent-based simulation system for evaluating gridding urban management strategies*, Journal of Knowledge-Based Systems, Volume 26, February 2012, Pages 174-184

[10] Thomas G. Preuss, Markus Brinkmann, Elin Lundstrm, Bengt-Erik Bengtsson, Magnus Breitholtz, *An individual-based modeling approach for evaluation of endpoint sensitivity in harpacticoid copepod life-cycle tests and optimization of test design*, Journal of Environmental Toxicology and Chemistry, Volume 30, Issue 10, pages 2353-2362, October 2011

[11] M. Ginovart, C. Prats, X. Portell, M. Silbert, *Analysis of the effect of inoculum characteristics on the first stages of a growing yeast population in beer fermentations by means of an individual-based model*, Journal of Industrial Microbiology & Biotechnology, January 2011, Volume 38, Issue 1, pp 153–165

[12] A.J. Verhulst, A.M. Cappuyns, E. Van Derlinden, K. Bernaerts, J.F. Van Impe, *Analysis of the lag phase to exponential growth transition by incorporating inoculum characteristics*, Journal of Food Microbiology, Volume 28, Issue 4, June 2011, Pages 656-666

[13] Pieter R. van Oel, Maarten S. Krol, Arjen Y. Hoekstra, *Application of multi-agent simulation to evaluate the influence of reservoir operation strategies on the distribution of water availability in the semi-arid Jaguaribe basin, Brazil*, Journal of Physics and Chemistry of the Earth, Parts A/B/C, Volumes 47-48, 2012, Pages 173-181

[14] A. Smajgl, E. Bohensky, *Behaviour and space in agent-based modelling: Poverty patterns in East Kalimantan, Indonesia*, Journal of Environmental Modelling & Software, Volume 45, July 2013, Pages 8-14

[15] Guy Peer, Klaus Henle, Claudia Dislich, Karin Frank, *Breaking Functional Connectivity into Components: A Novel Approach Using an Individual-Based Model, and First Outcomes*, DOI: 10.1371/journal.pone.0022355

[16] Guy Peer, Klaus Henle, Claudia Dislich, Karin Frank, *Connected fragmented habitats facilitate stable coexistence dynamics*, Journal of Ecological Modelling, Volume 222, Issue 3, 10 February 2011, Pages 447-455

[17] Guy Beauchamp, Peter Alexander, Roger Jovani, *Consistent waves of collective vigilance in groups using public information about predation risk* Journal of Behavioral Ecology, DOI: 10.1093/beheco/arr194

[18] Markus Kautz, Uta Berger, Dietrich Stoyan, Juliane Vogt, Nabiul Islam Khan, Karen Diele, Ulrich Saint-Paul, Tran Triete, Vien Ngoc Nam, *Desynchronizing effects of lightning strike disturbances on cyclic forest dynamics in mangrove plantations* Journal of Aquatic Botany, Volume 95, Issue 3, October 2011, Pages 173-181

[19] Isabel Martnez, Thorsten Wiegand, J. Julio Camarero, Enric Batllori, Emilia Gutirrez, *Disentangling the Formation of Contrasting Tree-Line Physiognomies Combining Model Selection and Bayesian Parameterization for Simulation Models*, American Naturalistm, Volume 177, Issue 5, May 2011, Pages 136–152

[20] Pavel Fibich, Jan Lep, *Do biodiversity indices behave as expected from traits of constituent species in simulated scenarios?* Journal of Ecological Modelling, Volume 222, Issue 13, 10 July 2011, Pages 2049-2058

[21] Benjamin T. Martin, Elke I. Zimmer, Volker Grimm, Tjalling Jager, *Dynamic Energy Budget theory meets individual-based modelling: a generic and accessible implementation* Journal of Methods in Ecology and Evolution, Volume 3, Issue 2, pages 445-449, April 2012

[22] B. C. Harvey, S. F. Railsback, *Effects of Passage Barriers on Demographics and Stability Properties of a Virtual Trout Population*, Journal of River Research and Applications, Special Issue: Special Issue on Fish Passage: An Ecohydraulics Approach, Volume 28, Issue 4, pages 479-489, August 2011

[23] Clment Chion, P. Lamontagne, S. Turgeon, L. Parrott, J.-A. Landry, D.J. Marceau, C.C.A. Martins, R. Michaud, N. Mnard, G. Cantin, S. Dionne, *Eliciting cognitive processes underlying patterns of human-wildlife interactions for agent-based modelling* , Journal of Ecological Modelling, Volume 222, Issue 14, 24 July 2011, Pages 2213-2226

[24] Alexander Peringer, Gert Rosenthal, *Establishment patterns in a secondary tree line ecotone*, Journal of Ecological Modelling, Volume 222, Issue 17, 10 September 2011, Pages 3120-3131

[25] Innocent Bakam, Guillaume Pajot & Robin B. Matthews, *Estimating carbon price from a closed emissions trading scheme in the agricultural sector*, Journal of Land Use Science, Volume 7, Issue 2, Pages 221–238, Published online: 2011

[26] Michal Arbilly, Uzi Motro, Marcus W. Feldman, Arnon Lotem, *Evolution of social learning when high expected payoffs are associated with high risk of failure*, Journal of The Royal Society, Interface 7, Volume 8, no. 64, Pages 1604–1615, November 2011

[27] Simon McDonnell, Moira Zellner, *Exploring the effectiveness of bus rapid transit a prototype agent-based model of commuting behavior* , Journal of Transport Policy, Volume 18, Issue 6, November 2011, Pages 825-835

[28] Kachel AF, Premo LS, Hublin JJ, *Grandmothering and natural selection* , Proceedings of The Royal Society B, Volume 278, no. 1704, Pages 384–391, February 2011

[29] Cdric Sueur, Jean-Louis Deneubourg, Odile Petit, Iain D. Couzin, *Group size, grooming and fission in primates: A modeling approach based on group structure* Journal of Theoretical Biology, Volume 273, Issue 1, 21 March 2011, Pages 156-166

[30] Brian V. Merkey, Laurent A. Lardon, Jose M. Seoane, Jan-Ulrich Kreft, Barth F. Smets, *Growth dependence of conjugation explains limited plasmid invasion in biofilms: an individual-based modelling study* , Journal of Environmental Microbiology, Volume 13, Issue 9, pages 2435-2452, September 2011

[31] Laurent A. Lardon, Brian V. Merkey, Snia Martins, Andreas Dtsch, Cristian Picioreanu, Jan-Ulrich Kreft, Barth F. Smets, *iDynoMiCS: next-generation individual-based modelling of biofilms* , Journal of Environmental Microbiology, Volume 13, Issue 9, pages 2416-2434, September 2011

[32] Thomas Mueller, William F. Fagan, Volker Grimm, *Integrating individual search and navigation behaviors in mechanistic movement models*, Journal of Theoretical Ecology, August 2011, Volume 4, Issue 3, pp 341–355

[33] C. Athena Aktipis, *Is cooperation viable in mobile organisms? Simple Walk Away rule favors the evolution of cooperation in groups* Journal of Evolution and Human Behavior, Volume 32, Issue 4, July 2011, Pages 263-276

[34] Tatiana Filatova, Alexey Voinov, Anne van der Veen, *Land market mechanisms for preservation of space for coastal ecosystems: An agent-based analysis* , Journal of Environmental Modelling & Software, Volume 26, Issue 2, February 2011, Pages 179-190

[35] Kelly Elizabeth Lane, *Landscape Dynamics: Genetics and Parasitism in Balinese Long-tailed Macaques (Macaca fascicularis)*, PhD Thesis, URL: http://etd.nd.edu/ETD-db/theses/available/etd-07202011-143422/

[36] Jolyon Alderman, Shelley A. Hinsley, Richard K. Broughton, Paul E. Bellamy, *Local Settlement in Woodland Birds in Fragmented Habitat: Effects of Natal Territory Location and*

Timing of Fledging Journal of Landscape Research, Volume 36, Issue 5, Pages 553–571, 2011

[37] Arne Pommerening, Valerie LeMay, Dietrich Stoyan, *Model-based analysis of the influence of ecological processes on forest point pattern formation-A case study* , Journal of Ecological Modelling, Volume 222, Issue 3, 10 February 2011, Pages 666-678

[38] L.S. Premo, A. Friederike Kachel, Jean-Jacques Hublin, *Modeling the Effects of Weaning Age on Length of Female Reproductive Period: Implications for the Evolution of Human Life History*, American Journal of Human Biology, Volume 23, Issue 4, pages 479-487, July/August 2011

[39] Geoffrey Caron-Lormier, David A. Bohan, Richard Dye, Cathy Hawes, Roger W. Humphry, Alan Raybould, *Modelling an ecosystem: The example of agro-ecosystems* , Journal of Ecological Modelling, Volume 222, Issue 5, 10 March 2011, Pages 1163-1173

[40] H. Dupont, J.B. Mihoub, N. Becu, F. Sarrazin, *Modelling interactions between scavenger behaviour and farming practices: Impacts on scavenger population and ecosystem service efficiency* Journal of Ecological Modelling, Volume 222, Issue 4, 24 February 2011, Pages 982-992

[41] Patrick M. A. James, M.-J. Fortin, B. R. Sturtevant, A. Fall, D. Kneeshaw, *Modelling Spatial Interactions Among Fire, Spruce Budworm, and Logging in the Boreal Forest* , Journal of Ecosystems, January 2011, Volume 14, Issue 1, pp 60–75

[42] M. Burkitt, D. Walker, D. M. Romano, A. Fazeli, *Modelling sperm behaviour in a 3D environment*, CMSB '11 Proceedings of the 9th International Conference on Computational Methods in Systems Biology, Pages 141–149

[43] Yan Liu, *Modelling sustainable urban growth in a rapidly urbanising region using a fuzzy-constrained cellular automata approach*, International Journal of Geographical Information Science, Volume 26, Issue 1, pages 151–167, Published online: 24 Oct 2011

[44] Francisca Ana Soares dos Santos, Karin Johst, Volker Grimm, *Neutral communities may lead to decreasing diversity-disturbance relationships: insights from a generic simulation model*, Ecology Letters, Volume 14, Issue 7, pages 653-660, July 2011

[45] C. Athena Aktipis, Eduardo Fernandez-Duque, *Parental investment without kin recognition: simple conditional rules for parent-offspring behavior*, Behavioral Ecology and Sociobiology, May 2011, Volume 65, Issue 5, pp 1079–1091

[46] Steven F. Railsback, Matthew D. Johnson, *Pattern-oriented modeling of bird foraging and pest control in coffee farms*, Ecological Modelling, Volume 222, Issue 18, 24 September 2011, Pages 3305-3319

[47] Brd-Jrgen Brdsen, John-Andr Henden, Per Fauchald, Torkild Tveraa, Audun Stien, *Plastic reproductive allocation as a buffer against environmental stochasticity - linking life history and population dynamics to climate* , Oikos, Volume 120, Issue 2, pages 245-257, February 2011

[48] Alberto Acerbi, *Predation and the phasing of sleep: an evolutionary individual-based model*, Animal Behaviour, Volume 81, Issue 4, April 2011, Pages 801-811

[49] Holly Gaff, *Preliminary Analysis of an Agent-Based Model for a Tick-Borne Disease*, American Institute of Mathematical Sciences, Volume 8, Issue 2, Pages 463–473, April 2011

[50] Silvana Siehoff, Gottfried Lennartz, Ira C. Heilburg, Martina Ro-Nickoll, Hans Toni Ratte, Thomas G. Preuss, *Process-based modeling of grassland dynamics built on ecological indicator values for land use*, Ecological Modelling, Volume 222, Issues 23-24, Pages 3854-3868, 10 December 2011

[51] Trond Kristiansen, Kenneth F. Drinkwater, R. Gregory Lough, Svein Sundby, *Recruitment Variability in North Atlantic Cod and Match-Mismatch Dynamics*, Plos One, DOI 10.1371/journal.pone.0017456

[52] C. Athena Aktipis, Lee Cronk, Rolando de Aguiar, *Risk-Pooling and Herd Survival: An Agent-Based Model of a Maasai Gift-Giving System* Human Ecology, April 2011, Volume 39, Issue 2, Pages 131–140

[53] Simone Vincenzi, Alain J. Crivelli, Jarl Giske, William H. Satterthwaite, Marc Mangel, *Selective consequences of catastrophes for growth rates in a stream-dwelling salmonid*, Oecologia, February 2012, Volume 168, Issue 2, Pages 393–404

[54] Lorenz Fahse, Marco Heurich, *Simulation and analysis of outbreaks of bark beetle infestations and their management at the stand level*, Ecological Modelling, Volume 222, Issue 11, Pages 1833-1846, 10 June 2011

[55] Jiang Jiang, Donald L. DeAngelis, Thomas J. Smith, Su Yean Teh and Hock-Lye Koh, *Spatial pattern formation of coastal vegetation in response to external gradients and positive feedbacks affecting soil porewater salinity: a model study* , Landscape Ecology, Volume 27, Issue 1, Pages 109–119, 2011

[56] Hughes, Elinor Jane, *The effect of sex ratio on male reproductive success in painted turtles (Chrysemys picta)*, PhD Thesis, http://hdl.handle.net/10214/2871

[57] Huber, Markus O., *The evidence of stand development stages in simulations by an individual-tree growth simulator* , Canadian Journal of Forest Research, Volume 41, Number 1, Pages 124-138(15), January 2011

[58] Mathias Franz, Daniel van der Post, Oliver Schlke and Julia Ostner, *The evolution of cooperative turn-taking in animal conflict*, BMC Evolutionary Biology 2011, 11:323 doi:10.1186/1471-2148-11-323

[59] Muhammad Ali Imron, Sven Herzog, Uta Berger, *The Influence of Agroforestry and Other Land-Use Types on the Persistence of a Sumatran Tiger (Panthera tigris sumatrae) Population: An Individual-Based Model Approach*, Environmental Management, Volume 48, Issue 2, Pages 276–288, August 2011

[60] Elise Dion, Louis VanSchalkwyk, Eric F. Lambin, *The landscape epidemiology of foot-and-mouth disease in South Africa: A spatially explicit multi-agent simulation*, Ecological Modelling, Volume 222, Issue 13, Pages 2059-2072, 10 July 2011

[61] Woon-Seok Cho, Sven Erik Jorgensen, Tae-Soo Chon, *The role of conjugation in the gene-individual-population relationships in increasing eco-exergy* , Ecological Modelling, Volume 222, Issue 3, Pages 407-418, 10 February 2011

[62] Paul D. Grosman, Jochen A.G. Jaeger, Pascale M. Biron, Christian Dussault, Jean-Pierre Ouellet, *Trade-off between road avoidance and attraction by roadside salt pools in moose: An agent-based model to assess measures for reducing moose-vehicle collisions*, Ecological Modelling, Volume 222, Issue 8, Pages 1423-1435, 24 April 2011

[63] Damaris Zurell, Volker Grimm, Eva Rossmanith, Niklaus Zbinden, Niklaus E. Zimmermann, Boris Schrder, *Uncertainty in predictions of range dynamics: black grouse climbing the Swiss Alps*, Ecography, Volume 35, Issue 7, pages 590603, Published online: November 2011

[64] E.E. van Loon, J. Shamoun-Baranes, W. Bouten, S.L. Davies , S.L. Davis, *Understanding soaring bird migration through interactions and decisions at the individual level* , Journal of Theoretical Biology, Volume 270, Issue 1, Pages 112-126, 7 February 2011

[65] Boone, R. B., K. A. Galvin, S. B. BurnSilver, P. K. Thornton, D. S. Ojima, J. R. Jawson, *Using Coupled Simulation Models to Link Pastoral Decision Making and Ecosystem Services* , Ecology and Society 16(2): 6. [online], URL: http://www.ecologyandsociety.org/vol16/iss2/art6/

Appendix C

Reverse-Engineering of Case Study Publication

In the following, excerpts from [23] will be analysed, statement by statement. The sections in this appendix represent the sections of Bown et al.'s [23] article that are addressed.

C.1 Abstract

A_1.1 "Biodiversity is generally accepted to include both within species and between species variation."

This phrase constitutes a claim: a claim about the existence of a particular state of facts (a general acceptance, in this case). Note: the expression "generally accepted" needs to be clarified, either through a *context* element, or through a supporting argument.

A_1.2 "Consequently, the contribution to the functioning of ecosystems of variation among individuals should be accounted for."

The phrase can be classed as a claim. The term "consequently" points towards a *SupportedBy* relation (statement A_1.1 can either be a justification or substantiation, but since GSN does not allow a claim to be justified by another, it must be "supported" by it. As such, statement A_1.2 is solved by claim A_1.1. Note that the scope of the claim is not explicit: who should account for this contribution?

A_1.3.a "However, very little is known about patterns of diversity below the species level"

Statement A_1.3.a represents a claim about the absence or the lack of a particular state of facts (lack of knowledge, in this case). The term "however" does not point towards a *SupportedBy* or *InContextOf* relation, but rather to a *InconsistentWith*[1] relation (the statement does not oppose the preceding claim A_1.2, but points towards a situation inconsistent with the one in which A_1.2 is true). Note that the agency of the statement is missing; a *context* is needed to clarify who is referred to by the claim.

A_1.3.b "and less still about the patterns of individual diversity and their relation to ecosystem context and community function"

This is the second statement in phrase A_1.3 and represents a claim. The term "and" points towards a relation: the two claims are part of the same phrase; they work together and there is a certain relation between them; however they can and should still be evaluated independently.

A_1.4 "We present a model for the dynamics of individuals that is physiologically based and spatially explicit"

This phrase constitues a claim; although it can be considered "editorial" or "narrative", "descriptive" phrase, it describes the authors' work and it can however be a claim since it is a noun + verb phrase and can be substantiated.

[1]Similar to Debategraph's Inconsistency cross-link type.

A_1.5 "Individuals are defined in terms of measurable parameters that relate environmental context to phenotype and in this sense define the genotype"

• context to claim A.1.4, describes the individuals, to an extent (although arguably it is not a full context)

A_1.6 "Estimates for the variation in the parameter values are obtained from experiments conducted on the species Rumex acetosa."

• descriptive

• context to claim A.1.4 (physiologically based)

A_1.7 "Simulations are performed to predict the form of the relative abundance distribution, and the dependence of the predicted number of coexisting genotypes on patch area (the genotypearea curve)."

• descriptive

• can be used as context if there's any element referring to simulations

A_1.8 "We find that the predicted forms of the abundance distribution and genotypearea curve are indistinguishable from those measured at the species level"

• claim

• adding "We find ¡through simulation¿, that" makes descriptive a context

A_1.9.a "Furthermore, we identify the importance of physiological trade-offs at the individual level in promoting diversity"

• claim

- “furthermore” places claim 7 on the same level as claim 6

A_1.9.b “and the sensitivity of genotype richness to the degree of similarity of individuals in the community”

- claim

- Composite relation with claim A.1.9.a

- “and” should be replaced by “We identify”

Figure C.2 presents a GSN diagram for Bown et al.’s [23] Abstract, using a linear arrangement of GSN elements.

Figure C.3 presents a GSN diagram for Bown et al.’s [23] Abstract, using a compact arrangement of GSN elements, with annotations and enforced *goal* elements).

C.2 Introduction

1.1.1.a “There exists a considerable literature based on both the structure and function of biological communities”

- claim

- the “considerable literature...” can’t be defined as a context, as it is neither a definition, nor can it be confined to a set of references (it is an unquantifiable, hence open statement). Therefore it should be substantiated.

1.1.1.b “with a particular focus on exploring the link between the two”

- claim

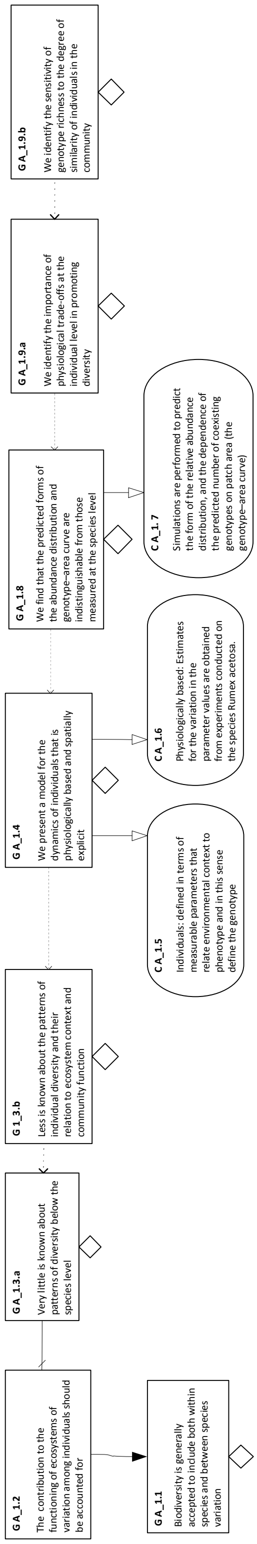


Figure C.1: GSN representation of [23]'s Abstract (linear arrangement)

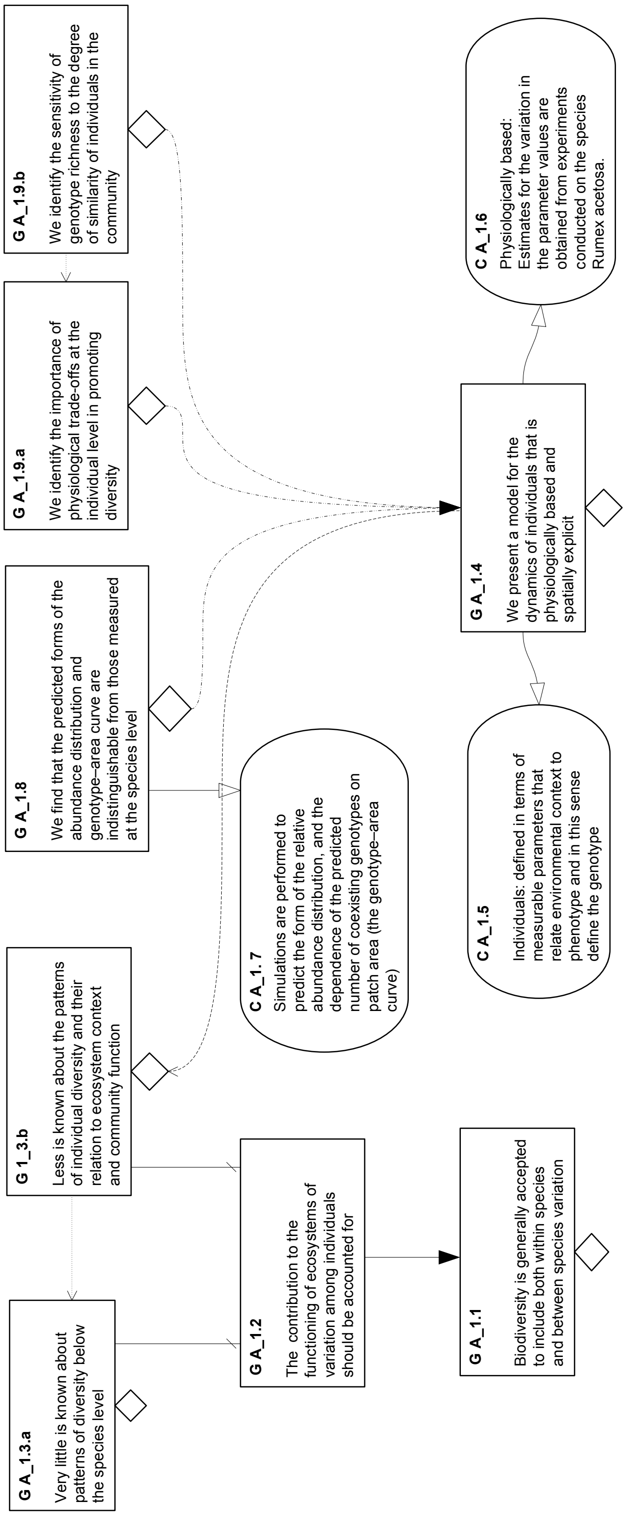


Figure C.2: GSN representation of [23]'s Abstract (compact arrangement)

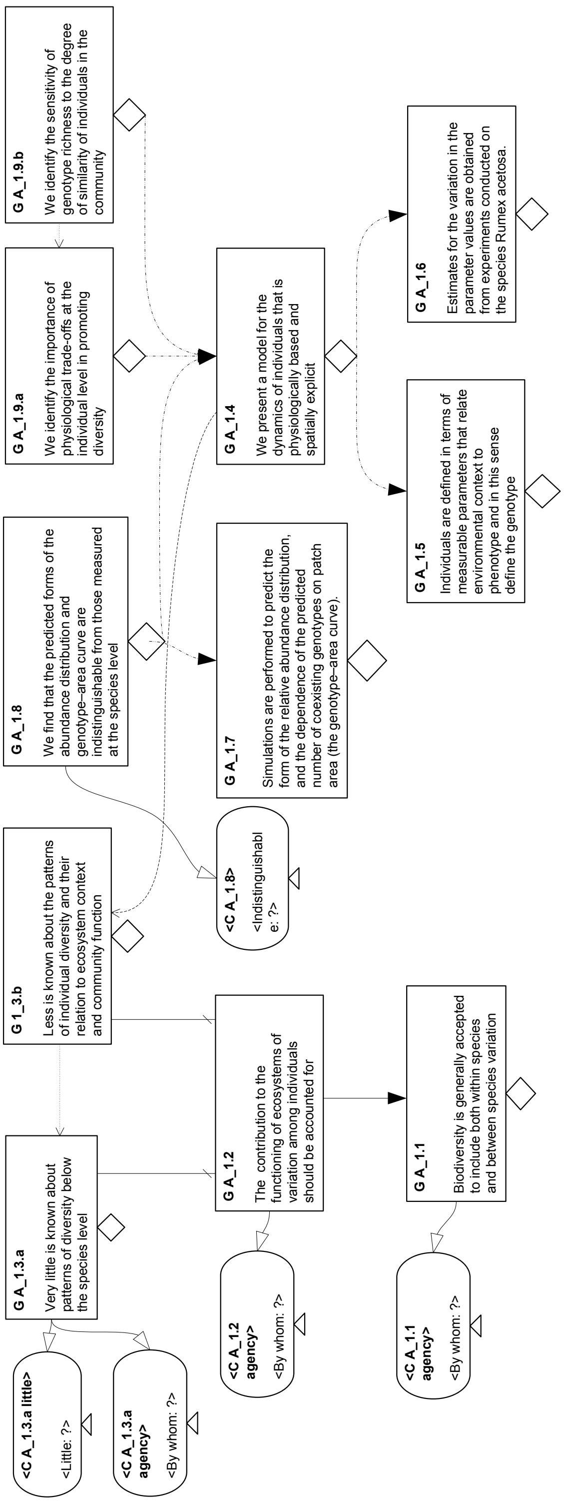


Figure C.3: GSN representation of [23]'s Abstract (compact arrangement, with annotations and enforced *goal* elements)

- "with" implies a dependence relationship with claim 1.1.1.a (if 1.1.1.a fails, 1.1.1.b can't logically be true, as it depends on the truth of 1.1.1.a). Consequently, 1.1.1.a is an Assumption to 1.1.1.b; 1.1.1.b can be true only in the context of 1.1.1.a.

- Also, should the content of claim 1.1.1.b be the exact statement, or should the "with" be replaced with "The considerable literature... has a particular focus..."?

1.1.1.b ", e.g. Naeem and Li (1997), Yachi and Loreau (1999), Schwartz et al. (2000) and Mouquet et al. (2002)"

- the supporting examples "e.g. Naeem...", forming Sn 1.1.1.b, apply only to 1.1.1.b or to 1.1.1.a also? Since they represent examples (e.g.) they certainly support 1.1.1.b and to a lower extent 1.1.1.a. Can we say 1.1.1.a is satisfied by Sn 1.1.1.b? No, but we can say it is justified, to an extent. But the authors consider it solved by Sn 1.1.1.b, since they find the e.g. sufficient.

- a new type of relation should exist: PartiallySupportedBy. In a phrase, claims will be unsupported and linked with a PartiallySupportedBy relation to the last block of citations (evidence), but the claim preceeding block of citations, which will be SupportedBy them and will be substantiated.

1.1.2.a "In spite of a considerable effort to seek general relations between species richness and function, e.g. productivity"

- claim

- "in spite of" points to a ContraryTo relation, not between claim 1.1.2.a and claims 1.1.1 but between claim 1.1.2.a and those following from i t(1.1.2.b and 1.1.2.c)

- "e.g." points to a context clairifying the type of general relations sought: C 1.1.2.a

- agency missing. + context: Who made this considerable effort?

1.1.2.b “no such generalisations have been found”

- claim

- agency missing. + context: by whom, just by the author or by anyone else?

1.1.2.c “and it is now believed that such relations, if they exist, depend on the phenotypic response of species to different environmental contexts (Mouquet et al., 2002)”

- “and” points towards a Composite relation with claim 1.1.2.b, but which can be evaluated independently of it.

1.1.3 “These findings acknowledge that in both structure, and structurefunction relations, variation in behavioural responses to environment is important”

- claim

- “these” does not clarify which findings are referred to. Is it all claims preceeding it, or just the subset of claims in the preceeding phrase (1.1.2)?

- the verb “suggest” points towards a weaker causality between claim 1.1.3 and “these findings”. Consequently, should we use SupportedBy or PartiallySupportedBy? We choose SupportedBy to relate with the composite claim 1.1.2 and PartiallySupportedBy to the composite claim 1.1.1;

1.1.4 “It is natural to consider how we might extend these ideas to study the implications of the full range of variation on community structure and function”

- claim

1.1.5.a "Not only will this inform the management of diversity by avoiding simplistic strategies that conserve the average habitat for the population and exploit the potential stabilising effects of intraspecific variation (Lomnicki, 1988)"

• claim. Initially considered a *justification*, but because it is supported by a citation (and

1.1.5.b "but it allows a clearer understanding of the relative importance of genetic isolation for the structure and functioning of communities"

• justification. It can also be treated as a unsupported claim.

C.3 Predicted Patterns of Diversity

Figure C.4 depicts the GSN representation of paragraph 4.2_1.

Figure C.5 exposes the GSN representation of paragraph 4.3_1.

C.4 Conclusions

Figure C.9 presents the result of activity 6 applied to G 5_1.4.b.

Figure C.10 presents the result of activity 6 applied to G 5_1.7.

Figure C.11 presents the result of activity 6 applied to G 5_1.10.

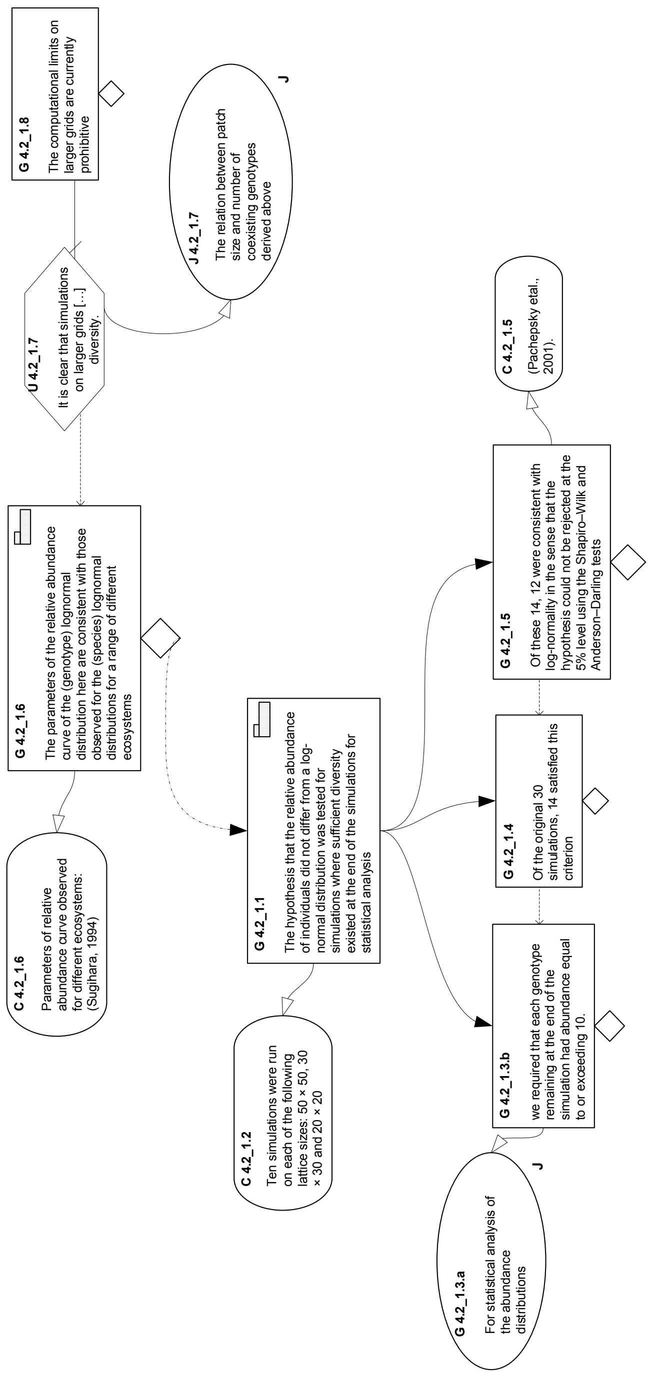


Figure C.4: GSN representation of [23]'s Predicted Patterns of Diversity, paragraph 4.2_1

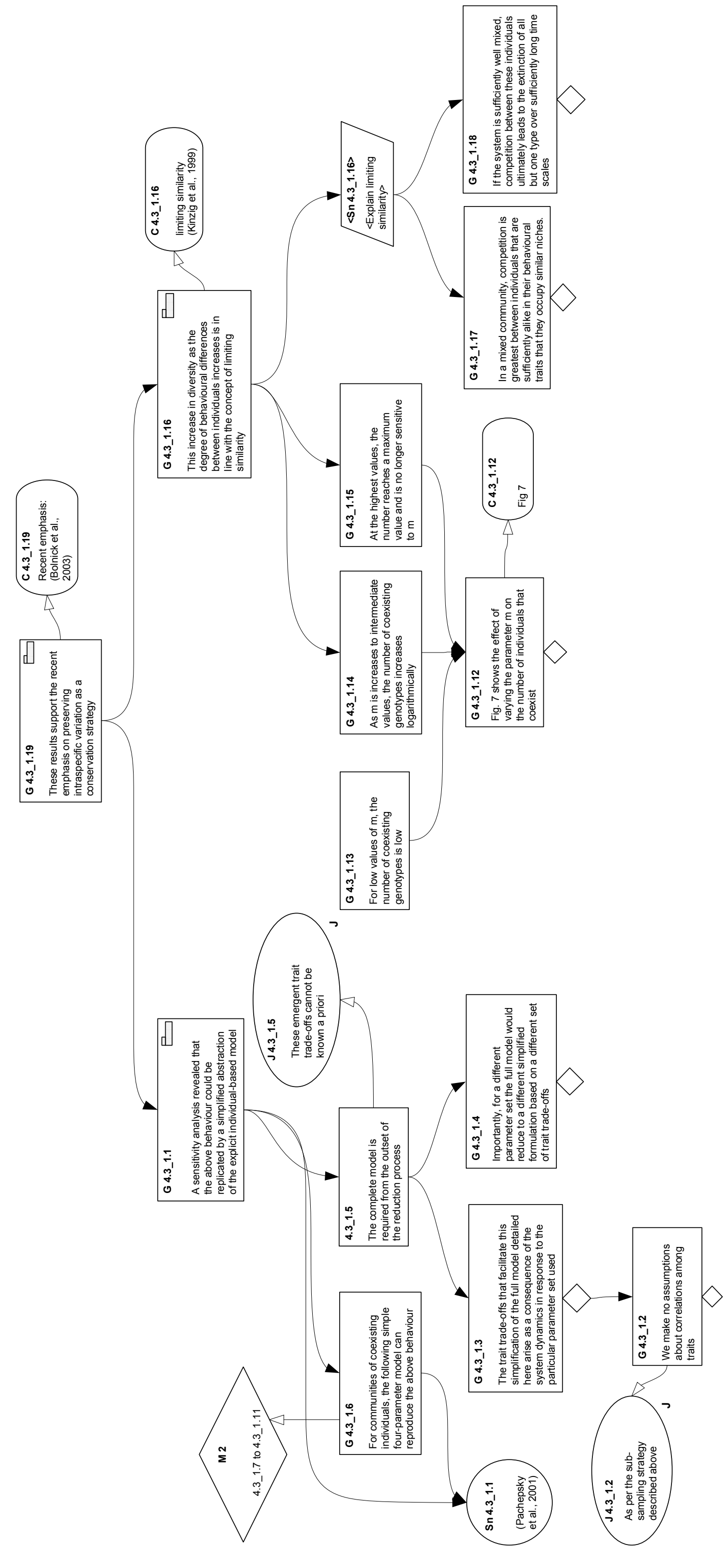


Figure C.5: GSN representation of [23]'s Predicted Patterns of Diversity, paragraph 4.3_1

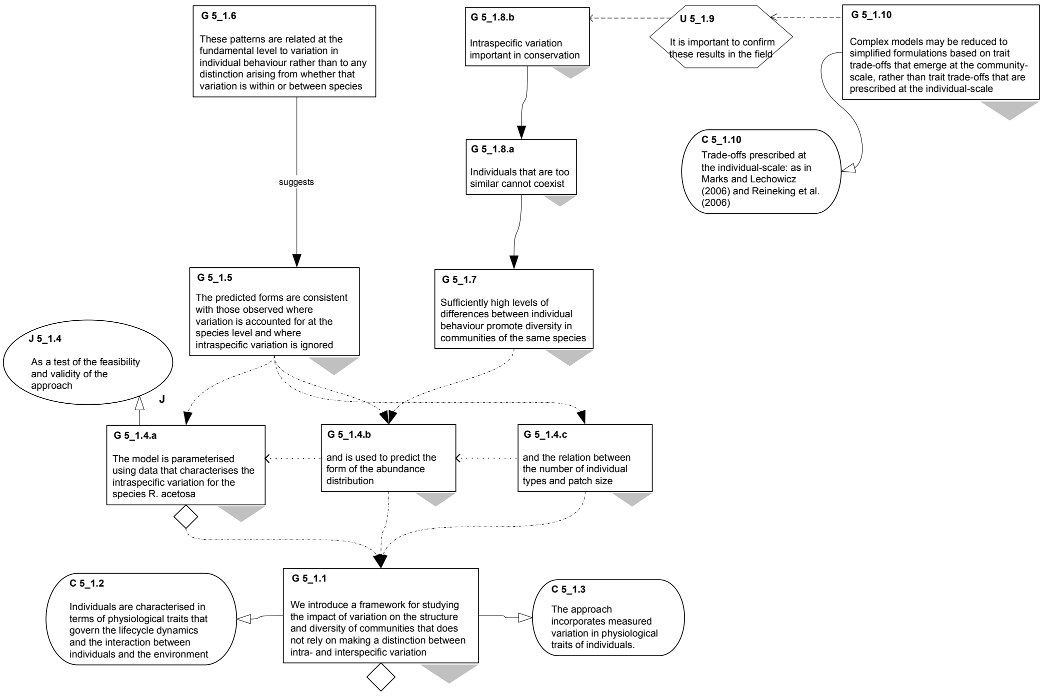


Figure C.6: GSN representation of [23]'s Conclusion, first paragraph

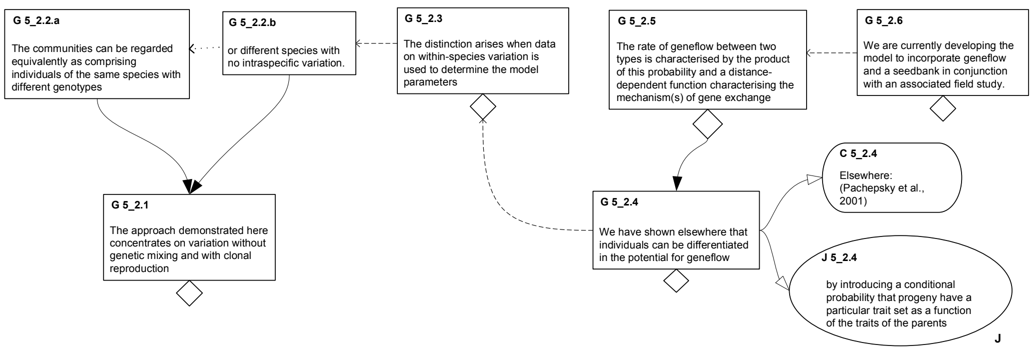


Figure C.7: GSN representation of [23]'s Conclusion, second paragraph

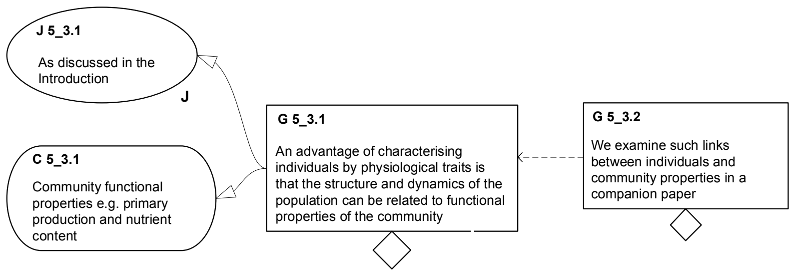


Figure C.8: GSN representation of [23]'s Conclusion, third paragraph

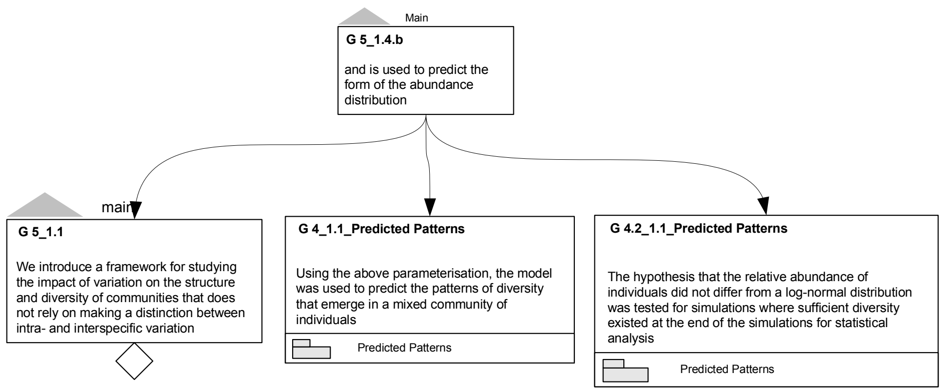


Figure C.9: Activity 6: retrospectively substantiating claim G 5_1.4.b

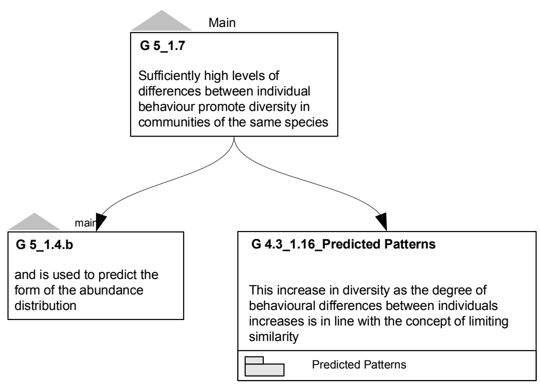


Figure C.10: Activity 6: retrospectively substantiating claim G 5_1.7

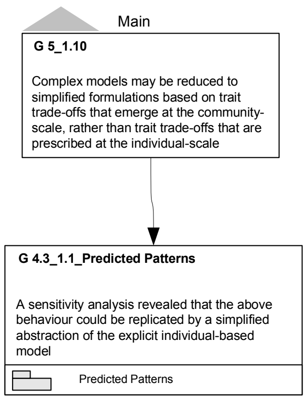


Figure C.11: Activity 6: retrospectively substantiating claim G 5_1.10

Appendix D

Equivalence Argument Details

D.1 CDesc – The C Simulator

D.1.1 Data structures

The C simulator makes use of four data structures, as reflected in Figure D.1. The spatial layout of the environment is a 2D grid, composed of *Location*[1] objects. Each spatial cell contains a resource substrate (the *Substrate* class) and an instance of a *Plant* class.

D.1.2 Pseudocode

D.1.2.1 High-level pseudocode

A model run represents the execution of one experiment, lasting for a number of time steps.

[1]Note: The names of the data structure in the original source code are different from the ones used in this description; the author decided to use different names, for readability purposes.

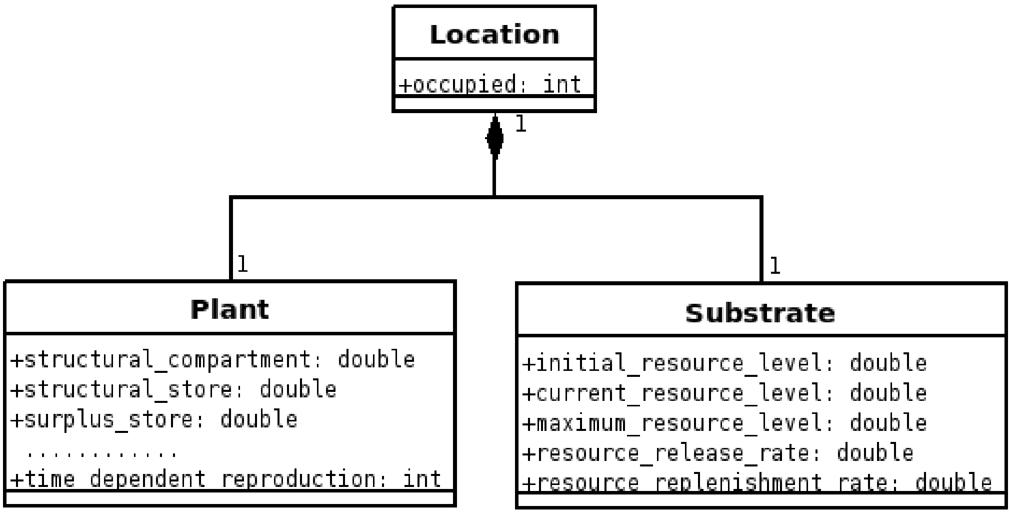


Figure D.1: Class diagram representing data structures in the C simulator

Algorithm 1 Model run

```
setup environment
for each time step do
    process environment
end for
```

Setting up the environment consists of two phases:

Algorithm 2 setup environment

```
instantiate 2D grid
instantiate individuals
```

Instantiate 2D grid a bi-dimensional array of lattice cells is created, each containing an instance of a substrate and of an individual. The substrate defines the resource characteristics of the cell, while the individual is relevant when the lattice cell is occupied.

Instantiate individuals The plant population is created. Firstly, an empty lattice cell, where the plant will be instantiated, is chosen randomply. The instantiation consists of setting the plant's trait values, by randomly sampling the trait distributions, and by initialising its other state variables e.g. age.

Processing the environment consists of an iterative execution of the following two phases:

Algorithm 3 process environment

```
for each time step do
  resource acquisition
  resource usage
end for
```

where resource acquisition and resource usage are defined as:

Algorithm 4 resource acquisition

```
for each grid cell do
  assess resource demand
  process resource demand
  replenish substrate
end for
```

Algorithm 5 resource usage

```
for each individual do
  allocate uptake
  assess death
  if not dead then
    assess development
    assess reproduction
  end if
end for
```

D.1.3 Low-level pseudocode

The details of the submodels are as follows:

Algorithm 6 assess resource demand

```
select neighbourhood of focal cell
create empty demand vector
for each lattice cell in neighbourhood do
  if lattice cell occupied then
    calculate demand of occupying plant on focal cell
    add demand to demand vector
  end if
end for
```

D.2 ODesc – The Occam-π Simulator

The occam-π language is a process-oriented, rather than an object-oriented language. Instead of C classes, data is contained in records; instead of functions, execution is performed by processes (*PROC*); communication among processes is performed, in a client-server manner, via channels (*CHAN*) while synchronization via *BARRIER*s.

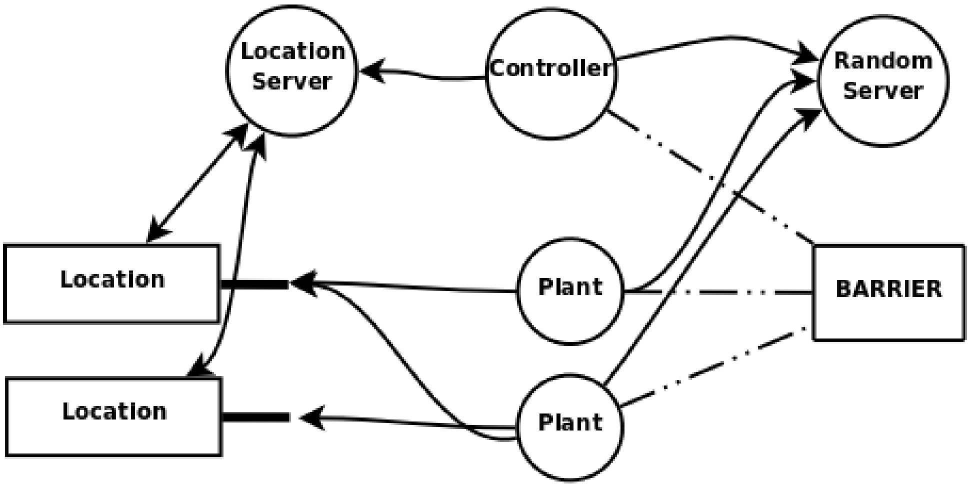


Figure D.2: Occam-π simulator architecture

D.2.1 Architecture

In order for an occam-π simulation to work, a client-server approach is used. Server processes always answer requests from clients; depending on the design, some processes may only act as servers, while others can switch between client and server roles. Topping this hierarchical architecture is a *Controller* process; this is the first process started by the occam-π runtime, which spawns the , controls and stops the simulation, collecting global information from and about the underlying processes, which is then delivered as simulation output to the simulator user.

In the case of this plant ecology occam-π simulation, the *Controller* process

The standard UML class diagram needs to be adapted to the process-oriented paradigm of occam-π in order to express the workings of the occam-π simulation; this was achieved by allowing both processes, records and channels to be expressed using the standard UML class representation, while prefixing processes and channels with their respective occam-π keywords. Figure D.3 provides the result.

Reflecting to the C simulation architecture (Section D.1), there is a *Location* process, which contains a resource *Substrate* record. Independent from the *Location*

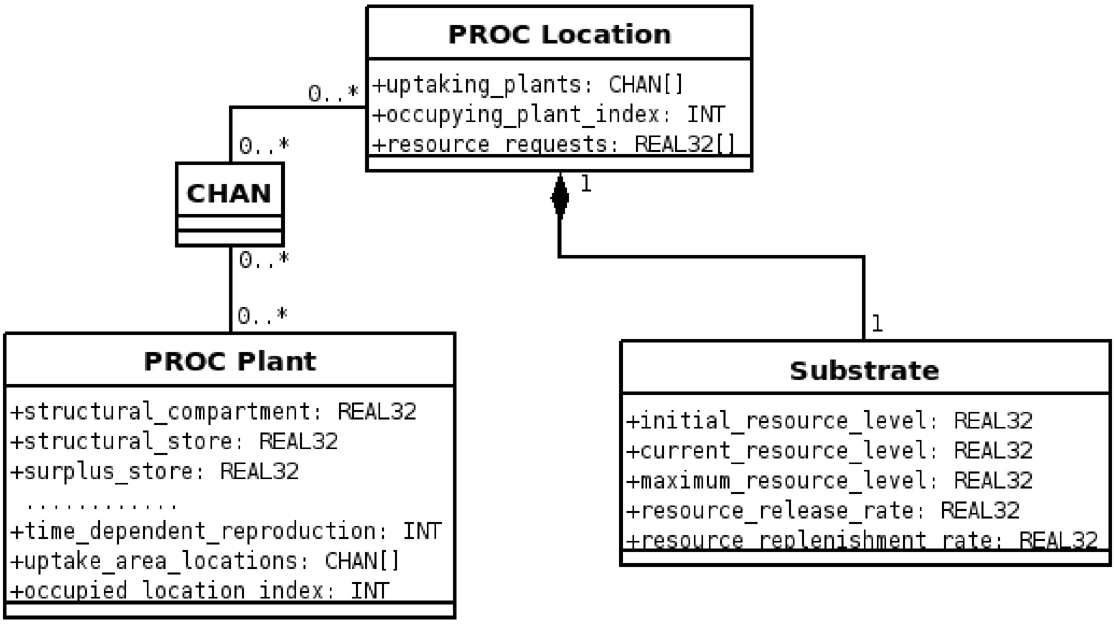


Figure D.3: Diagram representing data structures in the Occam-π simulator

process is the *Plant* process; the two types of processes interact through channels (*CHAN*) in a many-to-many relationship.

D.2.2 Pseudocode

As mentioned above, the occam-π is executed by a Controller process. This spawns the necessary processes (Location and Random server, Location and Plant processes), tracks the passage of "time" and, at the end of the simulation, terminates the spawned processes.

The spawned processes execute independently and in parallel. Because the conceptual model requires the execution of plant life-cycle activities in a sequential manner (e.g. resource uptake is followed by resource usage), barrier synchronization is used: the barrier blocks the execution of processes that are registered with it, until all of them issue a *SYNC* message (see Algorithm 9 for the pseudocode).

Location processes wait until they receive resource demands from Plant processes registered with them. They subsequently "deliver" (communicate) back the amount of resource apportioned to each Plant (see Algorithm 10). Locations then synchronise with a different barrier, marking the end of the resource uptake phase.

Plant processes then execute the resource usage phase, by allocating the received resources throughout their resource stores, increasing their development stage and then reproducing (if conditions are met). At the end of these activities, Plant processes synchronise, this final synchronization marking the end of a time step. Note: to detect the passage of time, the Controller also synchronises with the barriers.

Algorithm 7 Controller: Model run

```
Controller spawns servers
Controller spawns Location processes
Controller spawns Plant processes
for each time step, in parallel do
  Plants submit resource demands
  Plants undergo barrier synchronization

  Locations release resources
  Locations undergo barrier synchronization

  Plants allocate resources
  Plants assess development
  Plants assess reproduction
  Plants and Controller undergo barrier synchronization
end for
Controller terminates spawned processes
```

Algorithm 8 Plant: submit resource demand

```
calculate total resource demand
for each connected Location do
  send proportional resource demand
end for
synchronise with barrier
```

Algorithm 9 BARRIER: synchronization

```
flag = False
while flag == False do
  flag = True
  for all registered processes do
    if process not synchronized then
      flag = False
      break
    end if
  end for
end while
for all registered processes do
  release process from synchronization
end for
```

Algorithm 10 Location: release resources

```
repeat
  wait for communication from a connected Plant process
  if communication is a resource demand submission then
    store demand
  end if
until all connected plants have submitted their resource demands

calculate resources to be released to each Plant
for each connected Plant do
  send proportional resources
end for
synchronise with barrier
```

C Function	Occamp-π PROC
int model_run(void);	Controller
int initialise_simulation(void);	
void set_environment(void);	
void set_substrate(void);	
void set_distribution(void);	
double essential_uptake_trait(individual ind, int dvs);	Plant
void individual_development(int x, int y);	
double resource_shortfall_threshold_trait(individual ind);	
void assess_resource_shortfall(int x, int y);	
double requested_uptake_trait(individual ind);	
void distribute_resource(a_resource_demand dem[mdm]);	Location
void deplete_substrate(int x, int y, double tdm);	
void replenish_substrate(int x, int y);	
void process_environment(void);	None
void resource_acquisition(void);	

Table D.1: Parallel between functions from the C simulation and processes that in the occam-π one

D.3 AlgMapping

D.4 DataMapping

The following code extracts emphasise the way data structures are represented in the C simulation. As an example, the Plant structure is used.

```
typedef struct {
  //-- TRAITS
  // resource capture development type
  int rcdt;
  // essential uptake
  double eu_y0;
  // essential uptake
  double eu_b;
  // requested/essential uptake ratio
```

<table>
<tr><th>C simulation</th></tr>
<tr><td><pre>void replenish_substrate(int x, int y) {
  env[x][y].sub.cur+=env[x][y].sub.rep;
  if (env[x][y].sub.cur>env[x][y].sub.sat)
    env[x][y].sub.cur=env[x][y].sub.sat;
}</pre></td></tr>
<tr><th>Occamp-π simulation, Location process</th></tr>
<tr><td><pre>PROC replenish.subs (SUBSTRATE subs)
  SEQ
    subs[cur] := (subs[cur] + subs[rpr])
    IF
      subs[cur] > subs[sat]
        subs[cur] := subs[sat]
      TRUE
        SKIP
:</pre></td></tr>
</table>

Table D.2: Parallel between the C and occam-π implementations of the substrate replenishment function

```
double reur;
// St release rate
double strr;
// Su release rate
double surr;
// Time dependent fecundity
int tdfv;
// Compartment partition Ps
double ps;
// Survival assessment period Vp
double rsp;
// Survival assessment threshold Vt
double rst;
// Plant death probability
double pdp;
```

```
  //−− STATES
  // Type reference
  int tref;
  // Allocated resource
  double aru;
  // development stage
  int dvs;
  // age
  int age;
  // St: structural store
  double st;
  // Su: surplus store
  double su;
  // Ss: structural compartment
  double ss;
  // resource history
  double *rhs;
} Plant;
```

The following code extracts emphasise the way data structures are represented in the occam-π simulation.

```
DATA TYPE PLANT
  RECORD
    −−* TRAITS
    −−* resource capture development type
    INT rcdt:
    −−* essential uptake, y0
    REAL32 eu.y0:
    −−* essential uptake, b
    REAL32 eu.b:
    −−* essential uptake, a and x0
    REAL32 eu.x0,eu.
    −−* required/essential uptake ratio
```

```
REAL32 reur:
--* structural store (St) release rate
REAL32 strr:
--* surplus store (Su) release rate
REAL32 surr:
--* structural compartment store proportion
REAL32 sssp:
--* V.t., resource shortfall period trait
INT rsp:
--* V.p., resource shortfall threshold trait
REAL32 rst:
--* time dependent fecundity trait
INT tdfv:
--* development dependent fecundity trait
INT ddf:
--* seed dispersal pattern trait
INT sdp:
--* plant death probability
REAL32 pdp:
--* amount of resource per seed
REAL32  sr:

--* STATES
--* plant's local id
INT id:
--* Type reference
INT tref:
--* the plant's location
VECTOR2.INT pos:
--* Ss: structural compartment
REAL32 ss:
--*  structural store
REAL32 st:
```

```
--* surplus store
REAL32 su:
INT age:
INT dvs:

--* resource history
[mpr]REAL32 rhs:
INT rhs.idx:

RANDOM.STATE rs:
```

C simulation	**occamp-π simulation**

DATA TYPE SUBSTRATE
  RECORD
    --* initial resource level
    REAL32 ini:
    --* current resource level
    REAL32 cur:
    --* saturation level
    REAL32 sat:
    --* resource release rate
    REAL32 rlr:
    --* resource replenishment rate
    REAL32 rpr:
:
	typedef struct {
	  // initial resource level
	  double ini;
	  // current resource level
	  double cur;
	  // saturation level
	  double sat;
	  // resource release rate
	  double rel;
	  // resource replenishment rate
	  double rep;
	} substrate;

Table D.3: Parallel between the C struct and occam-π record

Appendix E

Assumption Deviation Analysis Results

Note: the following figures have the simulator name and parameter values encoded in their legend entries. The parameters are listed in the following order: initial population size, substrate level, simulation length, plant death probability and memory length. For example, a *cplants* simulation with an initial diversity of 75 individuals, a substrate level of 3 units, a simulation length of 50000 time steps, a plant death probability of 0.001 and a memory length of 6 will appear in the legend as cplants.75.3.50000.0.001.2. In some figures, *cplants* is replaced with *cp*.

E.1 Parameter Variation

E.1.1 Varying the Size of the Environment

Figure E.1 shows the behaviour of Bown et al.'s [23] model, if the environment size is increased from 10×10 grid cells to 1000×1000. The resulting genotype-area curve follows the expected log-normal path ($n = kA^z$) up to an environment size from which it seem to level.

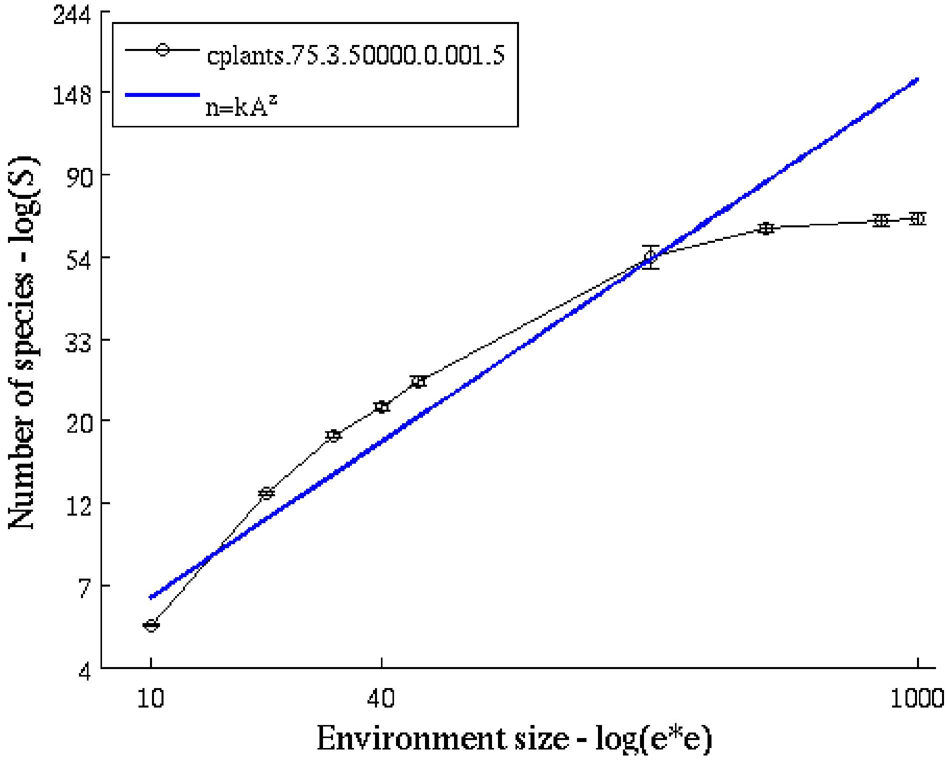


Figure E.1: Results after varying the size of the environment

This behaviour is acceptable as the initial genotype diversity becomes a limiting factor, for larger environments. These could support a larger diversity – and in this case, the genotype-area curve could remain log-normal –, but since the initial diversity is limited to 75 different genotypes, this number is asymptotically preserved as the environment gets larger.

E.1.2 Varying the Size of the Plant Memory

The plant memory is a modelling artefact, based on which it is assessed if the plant has been sufficiently under-resourced in the past, that it will die. The size of the plant memory represents the number of days for which the plant's available resources will be stored; the default value for this parameter is 5, meaning that it will record the values for the 5 previous days.

Figure E.2 presents results from experiments where this value was increased to 100. It is observable that there is no significant effect from increasing the size of the plant memory is.

E.1.3 Varying the Length of the Simulation

As simulation length increases, a larger number of genotypes are lost, hence the overall diversity is lower. Figure E.3 emphasises the fact that this decrease in diversity does not change the general outlook of the genotype-area curve generated by the simulation.

E.1.4 Varying the Plant Death Probability

Varying the plant death probability has two types of effects. If increased from 0.001 (the default value) to 0.010 (a ten-fold increase), then the resulting genotype-area curve pattern is preserved, albeit the overall diversity decreases (see Figure E.4.

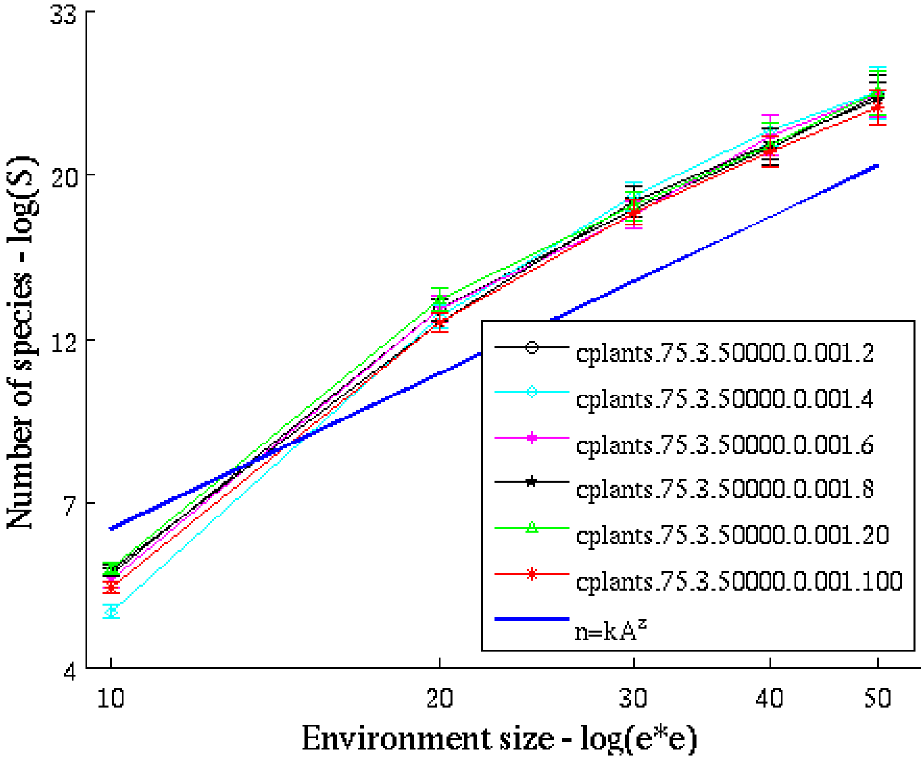


Figure E.2: Results after varying the size of the plant resource memory

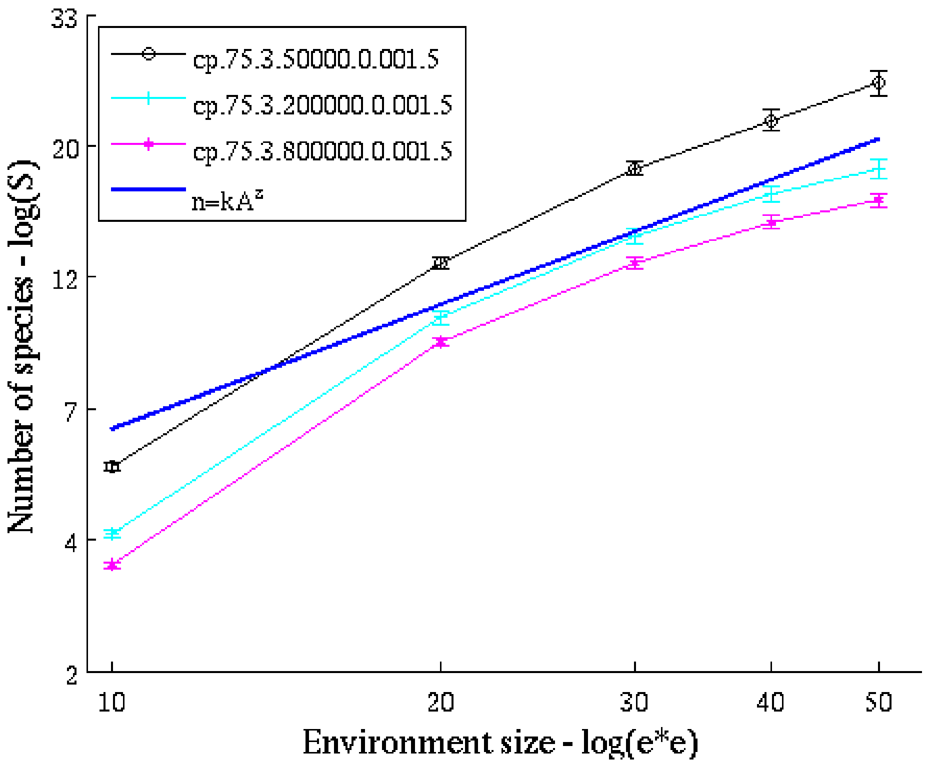


Figure E.3: Results after varying the length of the simulation

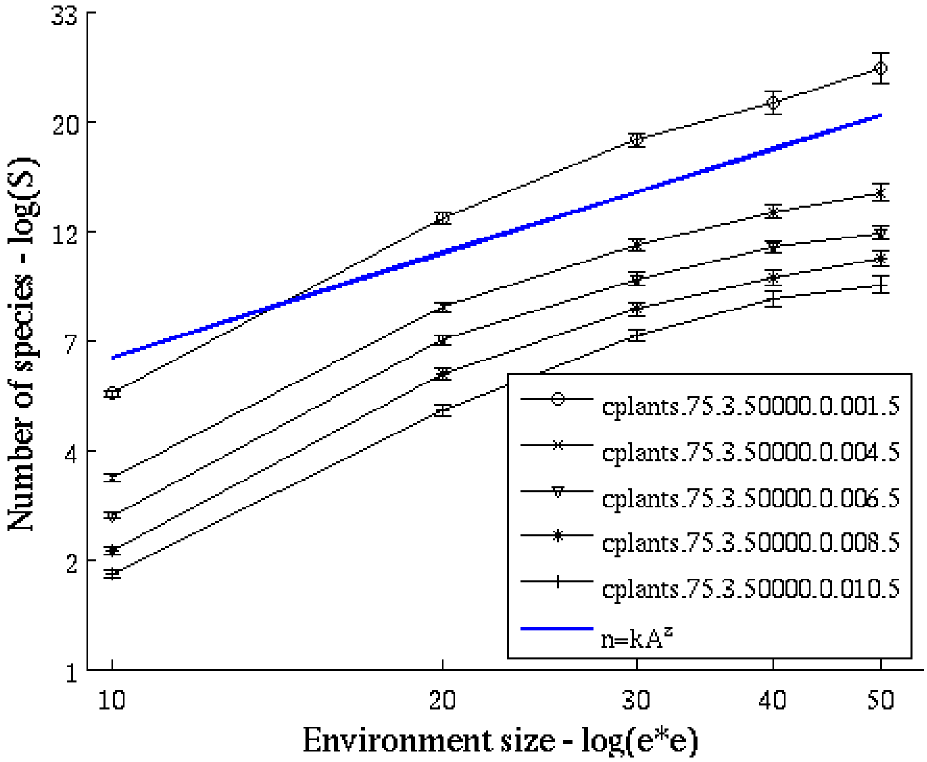


Figure E.4: Results after increasing the plant death probability

If decreased to zero (practically the assumption and parameter are removed from the model), then suddenly there is an important change: there is no more loss of diversity, irrespective of the environment size; the genotype-are relationship is no longer log-normal, but linear (see Figure E.5). This is a very important discovery as it implies that Bown et al.'s model is dependent on having this assumption of random death represented – it is a critical assumption.

E.2 Adding a Seedbank to the Model

In the original model, any seed that would land on an occupied location, would instantly die. Adding a simple seedbank to the model was performer by allowing locations to "store" the seeds that land on them; once the occupying plant died,

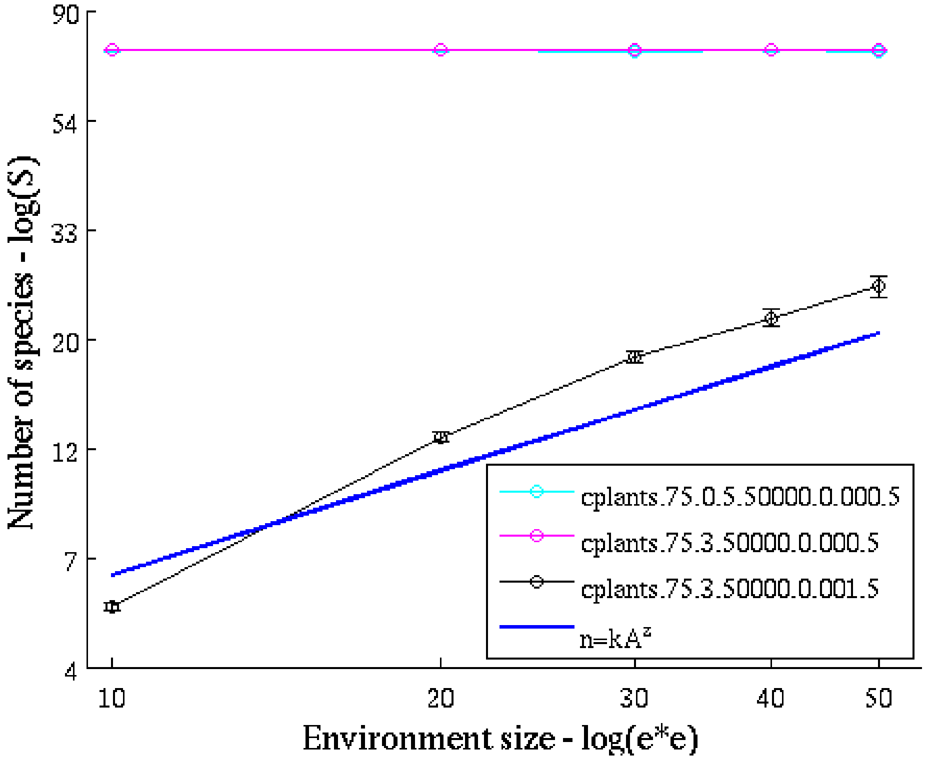


Figure E.5: Results after cancelling the plant death probability

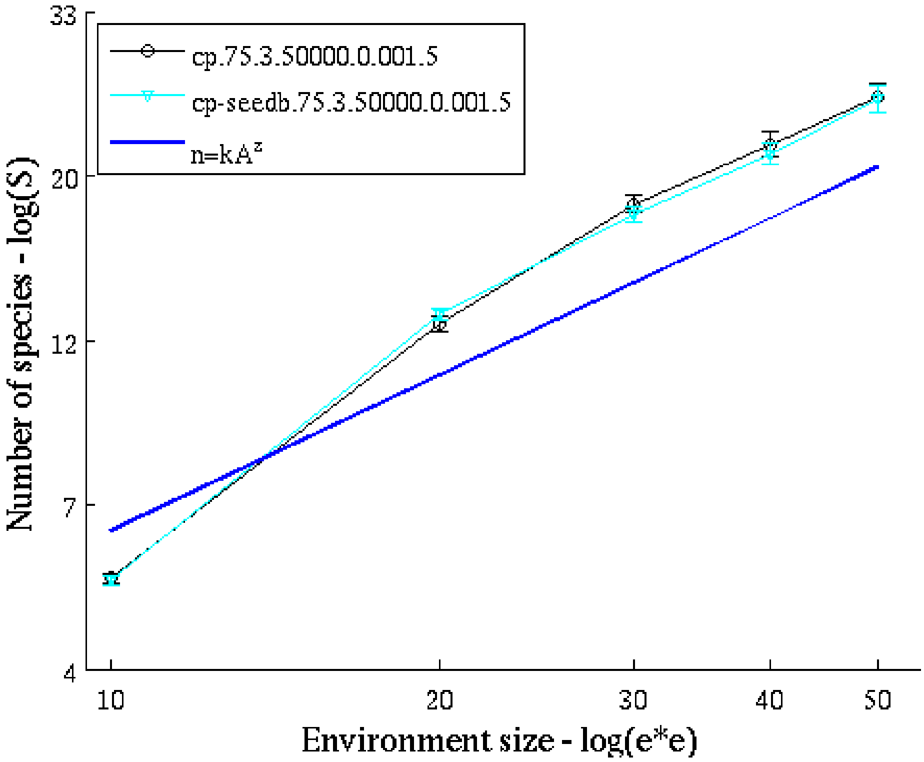


Figure E.6: Results after adding a seedbank to the model

a seed randomly selected from the location's seedbank would replace it.

Figure E.6 shows the difference between the original model and the seedbank enhanced one. We can observe that this assumption, invalid from a biological point of view, makes no difference if addressed through the addition of a seedbank. The resulting genotype-area curve of the altered model is approximately identical to the original one.

Glossary

ABM	Agent-Based Modelling
ALife	Artificial Life
ADA	Assumption Deviation Analysis
CAS	Complex Adaptive Systems
CoSMoS	Complex Systems Modelling and Simulation project
EBM	Equation-Based Modelling
GSN	Goal Structuring Notation
HAZOP	Hazard and Operability Study
IBM	Individual-Based Modelling
M&S	Modelling and Simulation
ODD	Overview, Design Concepts and Details protocol
POM	Pattern-Oriented Modelling
POMAC	Pareto Optimal Model Assessment Cycle
SA	Structured Argumentation
SBR	Simulation-Based Research
SCS	Safety-Critical Systems
TRACE	Transparent and Comprehensive Ecological Modeling
V&V	Verification and validation

Bibliography

[1] GSN community standard version 1. Technical report, Origin Consulting (York) Limited, November 2011. 16, 37, 42, 43, 44, 45, 47, 61, 64, 66, 67, 68, 69, 77, 83, 91, 101, 105, 115, 131

[2] Oxford dictionaries. definition of claim'. `http://oxforddictionaries.com/definition/claim`, November 2011. 40

[3] Oxford dictionaries. definition of 'critical'. `http://oxforddictionaries.com/definition/critical`, October 2011. 38

[4] Oxford dictionaries. definition of 'evidence'. `http://oxforddictionaries.com/definition/evidence`, November 2011. 41

[5] Oxford dictionaries. definition of fact'. `http://oxforddictionaries.com/definition/fact`, November 2011. 41

[6] Oxford dictionaries. definition of hazard'. `http://oxforddictionaries.com/definition/hazard`, November 2011. 38

[7] Oxford dictionaries. definition of risk'. `http://oxforddictionaries.com/definition/risk`, November 2011. 38

[8] Oxford dictionaries. definition of safety'. `http://oxforddictionaries.com/definition/safety`, November 2011. 38

[9] Oxford dictionaries. definition of safety-critical'. `http://oxforddictionaries.com/definition/safety-critical`, November 2011. 38

[10] Oxford dictionaries. definition of system'. `http://oxforddictionaries.com/definition/system`, November 2011. 38

[11] C. A. Abbott, M. W. Berry, E. J. Comiskey, L. J. Gross, and H. K. Luh. Computational models of White-Tailed deer in the florida everglades. Technical report, University of Tennessee, Department of Computer Science, Knoxville, TN, USA, 1995. 5

[12] R. Alexander. *Using Simulation for Systems of Systems Hazard Analysis.* PhD thesis, Department of Computer Science, University of York, 2007. 2

[13] Paul S. Andrews, Fiona Polack, Adam T. Sampson, Jon Timmis, Lisa Scott, and Mark Coles. Simulating biology: towards understanding what the simulation shows. In Susan Stepney, Fiona Polack, and Peter Welch, editors, *Proceedings of the 2008 Workshop on Complex Systems Modelling and Simulation*, pages 93–123, 2008. 16, 141

[14] Paul S. Andrews, Fiona A. C. Polack, Adam T. Sampson, Susan Stepney, and Jon Timmis. The CoSMoS process version 0.1: A process for the modelling and simulation of complex systems. Technical Report YCS-2010-453, Department of Computer Science, University of York, March 2010. 110, 112, 117

[15] Osman Balci. Credibility assessment of simulation results. In *Proceedings of the 18th conference on Winter simulation*, WSC '86, pages 38–44, New York, NY, USA, 1986. ACM. 14

[16] Osman Balci. Verification, validation, and accreditation. In *Proceedings of the 30th conference on Winter simulation*, WSC '98, pages 41–48, Los Alamitos, CA, USA, 1998. IEEE Computer Society Press. 13

[17] Lewis W. Beck. The natural science ideal in the social sciences. *The Scientific monthly*, **68**[6]:386–394, June 1949. 11

[18] M. A. Bedau. Can unrealistic computer models illuminate theoretical biology. *Proceedings of the 1999 Genetic and Evolutionary Computation Conference Workshop Program*, pages 20–23, July 1999. 2, 9, 11

[19] Boris Beizer. *Software Testing Techniques.* John Wiley & Sons, Inc., New York, NY, USA, 1990. 12, 121

[20] Uta Berger, Cyril Piou, Katja Schiffers, and Volker Grimm. Competition among plants: Concepts, individual-based modelling approaches, and a proposal for a future research strategy. *Perspectives in Plant Ecology, Evolution and Systematics*, **9**:121 – 135, 2008. 8, 10, 141

[21] M. Boerlijst and P. Hogeweg. Spiral wave structure in pre-biotic evolution: Hypercycles stable against parasites. *Physica D: Nonlinear Phenomena*, **48**[1]:17–28, February 1991. 141

[22] James L. Bown. Issues of scale in individual-based models : applications in fungal and plant community dynamics, 2000. 54

[23] James L. Bown, Elizaveta Pachepsky, Alistair Eberst, Ursula Bausenwein, Peter Millard, Geoff R. Squire, and John W. Crawford. Consequences of intraspecific variation for the structure and function of ecological communities: Part 1. model development and predicted patterns of diversity. *Ecological Modelling*, **207**[2-4]:264–276, October 2007. ix, xi, xii, xiii, 54, 55, 58, 59, 71, 72, 73, 75, 81, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 99, 100, 101, 102, 103, 104, 105, 107, 108, 112, 114, 115, 116, 117, 118, 127, 128, 129, 130, 133, 145, 146, 147, 151, 153, 157, 179, 182, 183, 184, 185, 189, 190, 191, 192, 209

[24] George E. P. Box and Norman R. Draper. *Empirical Model-Building and Response Surfaces (Wiley Series in Probability and Statistics)*. Wiley, 1 edition. 2, 9, 11, 139

[25] V. Breitenberg. *Vehicles: Experiments in Synthetic Psychology*. MIT Press, 1986. 9, 10

[26] Brooker, W. Rob, Maestre, T. Fernando, Callaway, M. Ragan, Lortie, L. Christopher, Cavieres, A. Lohengrin, Kunstler, Georges, Liancourt, Pierre, Tielborger, Katja, Travis, M. J. Justin, Anthelme, Fabien, Armas, Cristina, Coll, Lluis, Corcket, Emmanuel, Delzon, Sylvain, Forey, Estelle, Kikvidze, Zaal, Olofsson, Johan, Pugnaire, Francisco, Quiroz, L. Constanza, Saccone, Patrick, Schiffers, Katja, Seifan, Merav, Touzard, Blaize, Michalet, and Richard. Facilitation in plant communities: the past, the present, and the future. *Journal of Ecology*, **96**[1]:18–34, January 2008. 54

[27] J. Bryden and J. Noble. Computational modelling, explicit mathematical treatments, and scientific explanation. In L. M. Rocha, L. S. Yaeger, M. A. Bedau, D. Floreano, R. L. Goldstone, and A. Vespignani, editors, *Artificial Life X: Proceedings of the Tenth International Conference on Artificial Life*, pages 520–526. MIT Press, 2006. 2, 6

[28] Seth Bullock and Dave Cliff. The role of 'hidden preferences' in the artificial co-evolution of symmetrical signals. *Proceedings of the Royal Society of London, Series B*, **264**:505–511, 1997. 16

[29] Paul Cilliers. *Complexity and Postmodernism: Understanding Complex Systems*. Routledge, January 1998. 1

[30] P. Collard and S. Mesmoudi. How to prevent intolerant agents from high segregation? In T. Lenaerts, M. Giacobini, H. Bersini, P. Bourgine, M. Dorigo, and R. Doursat, editors, *Proceedings of the Eleventh European Conference on the Synthesis and Simulation of Living Systems*, pages 168–175. MIT Press, 2011. xiii, 161

[31] P. Cooley and E. Solano. Agent-Based model (ABM) validation considerations. In *SIMUL 2011, The Third International Conference on Advances in System Simulation*, pages 126–131, 2011. 6, 12, 13, 14, 28

[32] G. Despotou and T. Kelly. Extending safety deviation analysis techniques to elicit flexible dependability requirements. In *System Safety, 2006. The 1st Institution of Engineering and Technology International Conference on*, pages 29–38. IET, June 2006. 135

[33] R. Durrett and S. Levin. The importance of being discrete (and spatial). *Theoretical Population Biology*, **46**[3]:363–394, December 1994. 8

[34] Virginia N. L. Franqueira, Thein T. Tun, Yijun Yu, Roel Wieringa, and Bashar Nuseibeh. Risk and argument: A risk-based argumentation method for practical security. In *19th IEEE International Requirements Engineering Conference*. IEEE Computer Society, 2011. 42

[35] Xiaocheng Ge, Rui Rijo, Richard F. Paige, Tim P. Kelly, and John A. McDermid. Introducing goal structuring notation to explain decisions in clinical practice. *Procedia Technology*, **5**:686–695, January 2012. 111, 159

[36] T. Ghetiu, R.D. Alexander, P.S. Andrews, F.A.C. Polack, and J. Bown. Equivalence arguments for complex systems simulations - a Case-Study. In S. Stepney, P. Welch, P. Andrews, and J. Timmis, editors, *Complex Systems Simulation and Modelling Workshop (CoSMoS 2009)*, pages 101–129, Frome, United Kingdom, 2009. Luniver Press. xvii, 111, 119, 120, 121, 123, 126

[37] Teodor Ghetiu, Fiona A. C. Polack, and James Bown. Argument-Driven validation of computer simulations - a necessity rather than an option. In *VALID 2010: The Second International Conference on Advances in System Testing and Validation Lifecycle*, pages 1–4. IEEE Press, 2010. 111

[38] V. Grimm and A. Schmolke. How to read and write TRACE documentations. draft, March 2011. 138

[39] Volker Grimm. Ten years of individual-based modelling in ecology: what have we learned and what could we learn in the future? *Ecological Modelling*, **115**[2-3]:129 – 148, 1999. 2, 8, 10, 15, 23, 24, 26, 27

[40] Volker Grimm, Uta Berger, Finn Bastiansen, Sigrunn Eliassen, Vincent Ginot, Jarl Giske, John Goss-Custard, Tamara Grand, Simone K. Heinz, and Geir Huse. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, **198**[1-2]:115–126, September 2006. 2, 16, 20, 24, 25, 27, 30, 62, 142, 165

[41] Volker Grimm, Uta Berger, Donald L. DeAngelis, J. Gary Polhill, Jarl Giske, and Steven F. Railsback. The ODD protocol: A review and first update. *Ecological Modelling*, **221**[23]:2760–2768, November 2010. ix, 24, 25, 26, 27, 30, 165

[42] Volker Grimm, Eloy Revilla, Uta Berger, Florian Jeltsch, Wolf M. Mooij, Steven F. Railsback, Hans-Hermann Thulke,

Jacob Weiner, Thorsten Wiegand, and Donald L. DeAngelis. Pattern-Oriented modeling of Agent-Based complex systems: Lessons from ecology. *Science*, **310**[5750]:987–991, November 2005. 26, 28, 110, 140

[43] C. Haddon-Cave. The Nimrod Review: an independent review into the broader issues surrounding the loss of the RAF nimrod MR2 aircraft XV230 in afghanistan in 2006. Technical report, The Stationery Office (TSO), October 2009. 18, 66

[44] Bruce Hannon and Matthias Ruth. Game of life. In *Modeling Dynamic Biological Systems*, Modeling Dynamic Systems, pages 391–410. Springer International Publishing, 2014. 5

[45] I. Harvey. Opening stable doors: complexity and stability in nonlinear systems. In *Proceedings of the Eleventh European Conference on the Synthesis and Simulation of Living Systems*, pages 318–325, 2011. 2, 110

[46] Richard Hawkins, Tim Kelly, John Knight, and Patrick Graydon. A new approach to creating clear safety arguments. In Chris Dale and Tom Anderson, editors, *Advances in Systems Safety*, chapter 1, pages 3–23. Springer London, London, 2011. 45

[47] J. C. Helton. Uncertainty and sensitivity analysis for models of complex systems. In Frank Graziani, editor, *Computational Methods in Transport: Verification and Validation*, **62** of *Lecture Notes in Computational Science and Engineering*, chapter 9, pages 207–228. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. 137, 140

[48] M. B. Hesse. *Models and analogies in science*. University of Notre Dame Press, 1966. 11

[49] P. Humphreys. *Extending Ourselves. Computational Science, Empiricism, and Scientific Method*. Oxford University Press, New York, 2004. 9

[50] Michael Huston, Donald Deangelis, and Wilfred Post. New computer models unify ecological theory. *BioScience*, **38**[10]:682–691, 1988. 7

[51] A. J. Jakeman, R. A. Letcher, and J. P. Norton. Ten iterative steps in development and evaluation of environmental models. *Environmental Modelling & Software*, **21**[5]:602–614, May 2006. 6, 137, 139

[52] T. Kelly and S. Bates. The costs, benefits, and risks associated with Pattern-Based and modular safety case development. In *Proceedings of the UK MoD Equipment Safety Assurance Symposium*, October 2005. 19, 111, 112

[53] T. Kelly and R. Hawkins. A structured approach to selecting and justifying software safety evidence. In *IET System Safety Conference*, 2010. 18, 41, 123

[54] T. P. Kelly. *Arguing safety – a systematic approach to managing safety cases*. PhD thesis, Department of Computer Science, University of York, 1999. xi, 37, 39, 40, 42, 44, 67, 111, 120, 158

[55] H. Kitano, S. Hamahashi, J. Kitazawa, K. Takao, and Imai. The virtual biology laboratories: A new approach of computational biology. In P. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life (ECAL'97)*, pages 274–283. MIT Press˜/ Bradford Books, Cambridge, MA, 1997. 1, 11

[56] Trevor A. Kletz. HAZOP - past and future. *Reliability Engineering & System Safety*, **55**[3]:263–266, March 1997. 135, 140

[57] S. Lavorel and E. Garnier. Predicting changes in community composition and ecosystem functioning from plant traits: revisiting the holy grail. *Functional Ecology*, **16**[5]:545–556, 2002. 52

[58] S. A. Levin, B. Grenfell, A. Hastings, and A. S. Perelson. Mathematical and computational challenges in population biology and ecosystems science. *Science*, **275**[5298]:334–343, January 1997. 10

[59] Simon McDonnell and Moira Zellner. Exploring the effectiveness of bus rapid transit a prototype agent-based model of commuting behavior. *Transport Policy*, **18**[6]:825–835, 2011. 3

[60] Peter B. Medawar. *Induction and Intuition in Scientific Thought*. American Philosophical Society, June 1969. 3

[61] Cristina Montañola Sales, Stephan Onggo, and Josep Casanovas-Garcia. *Agent-based simulation validation: A case study in demographic simulation*. IARIA, 2011. 11, 12, 114, 121, 139

[62] B. Nicolescu. *Methodology of Transdisciplinarity-Levels of Reality, Logic of the Included Middle and Complexity*, pages 31–54. TheATLAS Publishing, 2010. 65

[63] Stacey D. Ostermann-Kelm, Esther S. Rubin, Jeremiah D. Groom, James R. DeForge, Guy Wagner, Pete Sorensen, Steven G. Torres, Mark C. Jorgensen, Aimee J. Byard, and Oliver Ryder. Flawed model has serious conservation implications: Response to turner et al. *Wildlife Society Bulletin*, **33**[4]:1456–1464, 2005. 16

[64] Elizaveta Pachepsky, James L. Bown, Alistair Eberst, Ursula Bausenwein, Peter Millard, Geoff R. Squire, and John W. Crawford. Consequences of intraspecific variation for the structure and function of ecological communities part 2: Linking diversity and function. *Ecological Modelling*, **207**[2-4]:277–285, October 2007. 54

[65] Elizaveta Pachepsky, John W. Crawford, James L. Bown, and Geoff Squire. Towards a general theory of biodiversity. *Nature*, **410**[6831]:923–926, April 2001. 54

[66] National Science Foundation (NSF) Blue Ribbon Panel. Report on simulation-based engineering science: Revolutionizing engineering science through simulation. Technical report, May 2006. 10

[67] V. D. H. Parunak, Robert Savit, and Rick L. Riolo. Agent-Based modeling vs. Equation-Based modeling: A case study and users' guide Multi-Agent systems and Agent-Based simulation. In Jaime S. Sichman, Rosaria Conte, and Nigel Gilbert, editors, *Lecture Notes In Computer Science*, **1534** of *Lecture Notes in Computer Science*, chapter 2, pages 277–283. Springer-Verlag London, UK, Berlin, Heidelberg, 1998. 6

[68] Fiona A. C. Polack, Alastair Droop, Philip Garnett, Teodor Ghetiu, and Susan Stepney. Simulation validation: exploring the suitability of a simulation of cell divis'ion and differentiation in the prostate. In *Proceedings of the 2011 Workshop on Complex Systems Modelling and Simulation*, pages 113 – 133. Luniver Press, August 2011. xii, xvii, 111, 119, 126, 127

[69] Fiona A. C. Polack, Tim Hoverd, Adam T. Sampson, Susan Stepney, and Jon Timmis. Complex systems models: engineering simulations. In *Eleventh International Conference on the Simulation and Synthesis of Living Systems*, pages 482–489. MIT Press, 2008. 121

[70] Iyad Rahwan and Chris Reed. The argument interchange format argumentation in artificial intelligence. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, chapter 19, pages 383–402. Springer US, Boston, MA, 2009. 42

[71] D. C. Rapaport. *The Art of Molecular Dynamics Simulation.* Cambridge University Press, New York, NY, USA, 1996. 9

[72] M. Read, P. Andrews, J. Timmis, and V. Kumar. Techniques for grounding Agent-Based simulations in the real domain: a case study in experimental autoimmune encephalomyelitis. *Mathematical and Computer Modelling of Dynamical Systems*, **17**[4]:296–302, 2011. 128

[73] Joel H. Reynolds and E. David Ford. Multi-Criteria assessment of ecological process models. *Ecology*, **80**[2], 1999. 140

[74] John Rushby. Formalism in safety cases. In Chris Dale and Tom Anderson, editors, *Making Systems Safer*, chapter 1, pages 3–17. Springer London, London, 2010. 107

[75] Adam T. Sampson, John M. Bjørndalen, and Paul S. Andrews. Birds on the wall: distributing a process-oriented simulation. In *Proceedings of the Eleventh conference on Congress on Evolutionary Computation*, CEC'09, pages 225–231, Piscataway, NJ, USA, 2009. IEEE Press. 119

[76] Robert G. Sargent. An overview of verification and validation of simulation models. In *WSC '87: Proceedings of the 19th conference on Winter simulation*, pages 33–39, New York, NY, USA, 1987. ACM. 13

[77] Robert G. Sargent. Verification and validation of simulation models. In *Proceedings of the 37th conference on Winter simulation*, WSC '05, pages 130–143. Winter Simulation Conference, 2005. xii, 12, 13, 14, 16, 28, 110, 111, 112, 113, 115, 121, 136, 138, 139

[78] S. Schlesinger, Roy E. Crosby, Roland E. Cagne, George S. Innis, C. S. Lalwani, J. Loch, Richard J. Sylvester, Richard D. Wright, N. Kheir, and Da Bartos. Terminology for model credibility. *SIMULATION*, **32**[3]:103–104, 1979. 115

[79] Amelie Schmolke, Pernille Thorbek, Peter Chapman, and Volker Grimm. Ecological models and pesticide risk assessment: Current modeling practice. *Environmental Toxicology and Chemistry*, **29**[4]:1006–1012, 2010. 15, 23, 24, 25, 26, 27, 28

[80] Amelie Schmolke, Pernille Thorbek, Donald L. DeAngelis, and Volker Grimm. Ecological models supporting environmental decision making: a strategy for the future. *Trends in Ecology & Evolution*, **25**[8]:479–486, August 2010. 13, 14, 16, 112, 117, 137, 158

[81] A. Shore. Managing assumptions for safety critical projects. Technical report, University of York, September 2008. 35, 135, 137, 142

[82] Eric P. Smith. Uncertainty analysis. **4**:2283–2297, 2002. 2

[83] Harold Stanislaw. Tests of computer simulation validity: what do they measure? *Simul. Gaming*, **17**[2]:173–191, 1986. 9, 11, 12, 13, 16, 110, 136, 137, 145

[84] S. Stepney, F.A.C. Polack, and H.R. Turner. Engineering emergence. In *Engineering of Complex Computer Systems, 2006. ICECCS 2006. 11th IEEE International Conference on*, pages 9–pp. IEEE, 2006. 139

[85] S. Toulmin. *The Uses of Argument*. 1958. 42

[86] Egbert H. Van Nes and Marten Scheffer. A strategy to improve the contribution of complex simulation models to ecological theory. *Ecological Modelling*, **185**[2-4]:153–164, July 2005. 140

[87] R. A. Weaver, J. A. McDermid, and T. P. Kelly. Software safety arguments: Towards a systematic categorisation of evidence. In *Proc. 20 th International System Safety Conference, Denver USA, System Safety Society*, 2002. 18

[88] Rob Weaver, Georgios Despotou, Tim Kelly, and John Mcdermid. Combining software evidence: arguments and assurance. *SIGSOFT Softw. Eng. Notes*, **30**[4]:1–7, jul 2005. 18

[89] Uri Wilensky and William Rand. Making models match: Replicating an Agent-Based model. *Journal of Artificial Societies and Social Simulation*, **10**[4]:2, 2007. 16, 119

[90] Oliver Will. Resolving a replication that failed: News on the Macy & Sato model. *Journal of Artificial Societies and Social Simulation*, **12**[4]:11, 2009. 119

[91] Oliver Will and Rainer Hegselmann. A replication that failed on the computational model in 'Ḿichael w. Macy and Yoshimichi Sato: Trust, cooperation and market formation in the U.S. and Japan. Proceedings of the National Academy of Sciences, May 2002'. *Journal of Artificial Societies and Social Simulation*, **11**[3]:3, 2008. 16, 119, 123, 133