HAL
open science

Cartography of Multi-Agent Model Parameter Space through a reactive Dicotomous Approach

Patrick Taillandier, Frédéric Amblard

To cite this version:

Patrick Taillandier, Frédéric Amblard. Cartography of Multi-Agent Model Parameter Space through a reactive Dicotomous Approach. European Simulation and Modelling Conference, 2011, Guimaraes, Portugal. pp.38-42. hal-00688412v1

HAL Id: hal-00688412

https://hal.science/hal-00688412v1

Submitted on 17 Apr 2012 (v1), last revised 3 Jun 2020 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CARTOGRAPHY OF MULTI-AGENT MODEL PARAMETER SPACE THROUGH A REACTIVE DICOTOMOUS APPROACH

Patrick Taillandier
MTG Lab. UMR-IDEES 6228
1, rue Thomas Becket,
Mont-Saint-Aignan, France
E-mail: patrick.taillandier@univ-rouen.fr

Frédéric Amblard
IRIT - Université Toulouse 1 Capitole
2, rue du Doyen Gabriel Marty
Toulouse, France
E-mail: frederic.amblard@univ-tlse1.fr

KEYWORDS
Agent-based Simulation, Model Parameter Space Cartography, Reactive Exploration, Dichotomous Approach

ABSTRACT

Parameter space exploration is a key issue in agent-based modeling. Many approaches were proposed concerning the optimization of a specific output of the model, but rare are the ones that aim at making the map of the parameter space. Yet, this map can bring very important information about a model. In this paper, we propose a new approach dedicated to this map making. Our approach is based on a reactive dichotomy of the parameter space following a criterion and on the use of random decompositions. We present two experiments that show that our approach allows, with the same number of simulations, to make a more relevant map than a uniform splitting.

INTRODUCTION

The agent-based modeling is now widely used to study complex systems. Its ability to represent several levels of interaction along a detailed environment representation favored such a development. There are nowadays numerous tools to help modelers to develop agent-based models. However analyzing such models can be very complex. Indeed, agent-based models can bring into plays numerous parameters that can each have an impact on the global dynamic of the system. Moreover, the stochastic nature of the agent-based models makes their analysis even more difficult.

The problem of the parameter space exploration is a classic problem in simulations. If numerous approaches were already proposed (e.g., (Gramacy et al. 2004; Lee et al. 2006)), very few specifically concern the agent-based simulations. Indeed, agent-based models are generally characterized by a large number of parameters that have a deep impact on the global dynamics of the system: a small modification of the value of one single parameter can lead to a radical modification of the dynamics of the whole system. Moreover, carrying out a simulation of an agent-based model is often very time-consuming. It is thus very important to limit as much as possible the number of simulations carried out.

Most of the existing approaches dedicated to agent-based models search to find the parameter values that allow to optimize a given fitness function (e.g., (Brueckner and Parunak 2003; Rogers and Tessin 2004; Calvez and Hutzler 2007)). While these studies are important for some application contexts (e.g., calibration of a model), they usually give rather few information on the system dynamics. In order to get a better view of the system dynamics, we propose to analyze the whole parameter space and not only to focus on values that optimizes a given output. The goal is to give the best approximation of the studied outputs according to the whole parameter space, i.e., to make a map of the model parameter space, while carrying out as few simulations as possible.

The mostly used approach to build such a map consists in splitting the parameter space into a uniform grid and to compute for each cell a representative output value. The problem of this approach comes from the uniform size of the cells: whatever the interest of the area covered by the cell, its size will remain the same. Thus, there is a risk of too many simulations that will be carried out in uninteresting areas (typically, areas where there are no variations of the output values), and too few in interesting ones (areas with important variations of the output values). Another problem comes from the size chosen for the cells: if too big, some important properties of the system dynamic can be missed; if too small, the computation times will be very high.

In order to tackle this issue, we propose a new approach based on a reactive dichotomy of the parameter space and on the use of random splitting. The next section describes the approach.

APPROACH PROPOSED

As mentioned in the previous section, our parameter space exploration approach is based on a reactive decomposition of the parameter space. It is based on the algorithm proposed in (Munos and Moore 2002), which is dedicated to the state abstraction for continuous time and space, deterministic dynamic control problems (e.g., reinforcement learning).

The principle of the approach is to start from a rough uniform grid and to split, in an iterative way, the most "promising" cells into sub-cells until a stopping criterion is checked. The choice of the most "promising" cells is made

according to a criterion that characterized the quantity of variations of the output inside this cell. Higher the variations, more the cell has a chance to be split. The general algorithm is the following:

Input values: S, K and R : integer
Initialization stage: decomposition of the parameter space into a grid
While the number of sets of parameter values for which the output values was computed < S:
    Selection of the *K* most promising cells
    For each selected cell:
        Splitting of the cell
    End for
    For *R* points built randomly
        Splitting of the cell that contains the point
    End for
End while

Three parameters have to be defined:

- *S*: number of sets of parameter values for which the output value was computed during the exploration
- *K*: number of "promising" cells that will be split at each iteration
- *R*: number of random points that will be drawn at each iteration

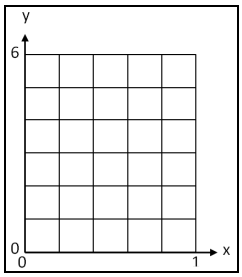
Our approach is composed of three important stages:

- Building of the initial grid.
- Selection and splitting of the most promising cells.
- Drawing of random points and splitting of the cells containing these points.

These stages are described in the following sections.

Initialization

The first step of our approach consists in splitting the model parameter space into a uniform grid. Figure 1 gives an example of initial grid that can be built for a parameter space composed of two parameters *x* and *y* which values are real between respectively [0,1] and [0,6].



Figures 1: Example of grid of dimension 2 (x,y)

Each cell will be composed of 2^N points with *N*, the dimension of the model parameter space. Each point corresponds to a vector of parameter value. Figure 2 gives an example of a grid for a parameter space of dimension 3.

For each point, the value of the studied output will be computed. In order to cope with the stochastic nature of the agent-based models, we propose, to determine the output value of a point, to carry out several simulations and to compute their average output value. We propose as well to define the global output value of a cell as the average output value of each point composing it. Thus, for example, in the case of the cell shown Figure 2, the global output value will be equal to the average output value of P_1, P_2, P_3, P_4, P_5, P_6, P_7 and P_8.

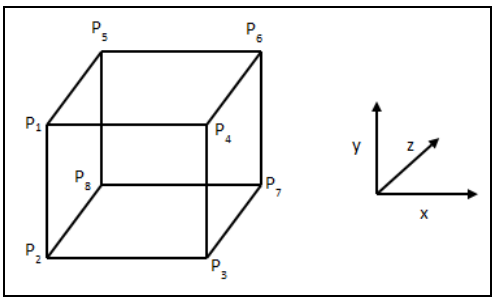


Figure 2: Cell of dimension 3 (x,y,z)

Selection and splitting of the most promising cells

This stage consists in selecting and splitting the most "promising" cells, i.e., the ones for which the value of the output studied varies the most.

Munos and Moore (Munos and Moore 2002) present several criteria that can be used for the selection. In our context, where the goal is to give a global view of the model parameter space and not only to focus on optimal parameter values, the two most relevant ones are:

- Average corner-value difference (ACVD): average of the absolute difference of the output value between two points of the cell along a direction.
  For example, in the case of the cell shown in Figure 1, the value of this criterion for the direction x, y and z are computed as follows:

$$Val_x^{ACVD} = \frac{1}{4}\cdot\begin{pmatrix} |val(P_1)-val(P_4)|+|val(P_2)-val(P_3)| \\ +|val(P_5)-val(P_6)|+|val(P_8)-val(P_7)| \end{pmatrix}$$

$$Val_y^{ACVD} = \frac{1}{4}\cdot\begin{pmatrix} |val(P_1)-val(P_2)|+|val(P_4)-val(P_3)| \\ +|val(P_5)-val(P_8)|+|val(P_6)-val(P_7)| \end{pmatrix}$$

$$Val_z^{ACVD} = \frac{1}{4}\cdot\begin{pmatrix} |val(P_1)-val(P_5)|+|val(P_2)-val(P_8)| \\ +|val(P_3)-val(P_7)|+|val(P_4)-val(P_6)| \end{pmatrix}$$

- *Value non-linearity (VNL)*: variance of the absolute increase of the output values between two points of the cell along a direction. The computation of this criterion is similar than the one based on the average, except that here the variance is used instead of the average.

Thus, for each cell, the criterion value will be computed for each direction, but only the direction that maximizes the criterion value will be kept.

Once each cell characterized by a criterion value and a direction, we propose to select the K cells that maximize the criterion value. The main interest to select K cells at the same iteration instead of just one is to allow the distribution of simulations. Indeed, each splitting can be computed independently on different processors or computers to minimize the computational time of the parameter space exploration process. Several frameworks specially dedicated to this task were proposed (Rioux et al. 2008; Abramson et al. 2009; Lorca et al. 2011).

Remarks that other selection methods such as selection by roulette, by tournament (see (Mitchell, 1996) for a description), and others, can be used at this step of our approach. The conclusion of the paper comes back on this point.

The K cells selected are then split in two sub-cells of same size along the direction chosen. Figure 3 shows the three possible decompositions for a cell of dimension 3.

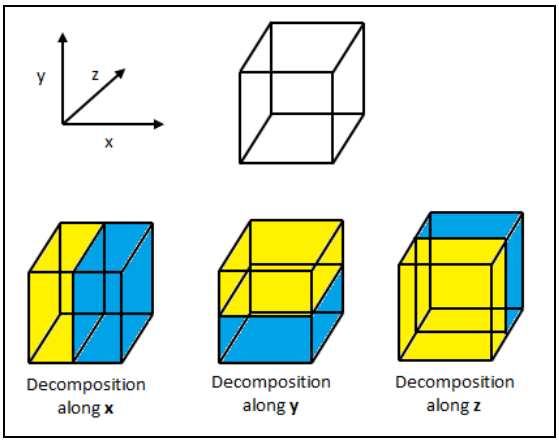


Figure 3: Possible decompositions for a cell of dimension 3 (x,y,z)

Drawing of random points and splitting of the cells containing these points.

The initial decomposition of the model parameter space into a grid can have a deep impact on the final result. As the cell selection method we propose in the previous section is purely greedy, i.e., it selects only the cells with the highest criterion value, a risk is to miss very local behavior of the system dynamic. Figure 4 illustrates this problem: in this example, cell 3 that hides a very interesting local phenomenon has little chance to be selected as the variation of the output value between the points P_2 and P_3 is very small.

A solution to this issue consists in using a more precise initial decomposition (with smaller cells), but it will increase the number of simulations that will be carried out during the initializing and thus decrease the number of simulations dedicated to the next stages of the exploration process.

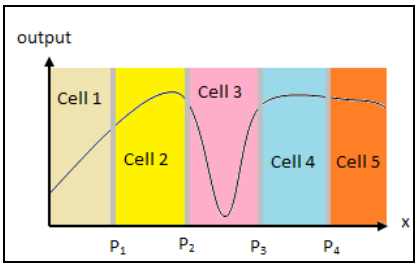


Figure 4: Example of initial decomposition for a parameter space of dimension 1 (x)

Another solution that we propose to apply in our approach is to draw random points in the parameter space to carry out random splitting of the cells.

Thus, R points will be drawn in the model parameter space. For each point, the cell containing this point will be split according to this point in two sub-cells along a random direction. Figure 5 shows the three possible decompositions for a cell of dimension 3 according to a given point.

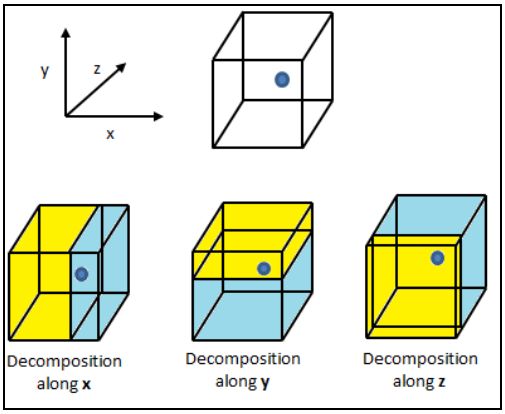


Figure 5: Possible random decompositions for a cell of dimension 3 (x,y,z)

R is a parameter of our approach. If $R = 0$, the approach will be fully greedy. A high value of R will favor the diversification of the search process over its intensification on the "interesting" cells detected.

The next section presents two experiments we carried out to evaluate our approach.

EXPERIMENTS

In order to test our approach, we propose two case-studies. Each of these case-studies consists in comparing the results obtained with a uniform decomposition of the model parameter space into a grid and with our approach for the same number of simulations. The goal is to evaluate the representativeness of the map built with these two approaches.

Our approach was implemented within the GAMA simulation platform (Taillandier et al. 2010; GAMA 2011). This platform provides a complete modelling and simulation

development environment for building spatially explicit multi-agent simulations. Its main advantage comes from the simplicity to define a model with it. Indeed, GAMA provides a rich modelling language, GAML, for easily modelling agents and environments. Moreover, this platform provides a batch mode that allows to run sets of simulations with different parameter values. At last, this platform, which is implemented in Java, is easily extensible.

The first case-study we propose consists in testing our approach for a mathematical function. The goal is to have first evaluation of our approach for a very simple deterministic model.

The function chosen is:

$$f(x,y) = \frac{2 \cdot x^2 + y^2}{1 + x^2}$$

We propose to make the map of this function for:

- $x \in [-10,10]$
- $y \in [-10,10]$

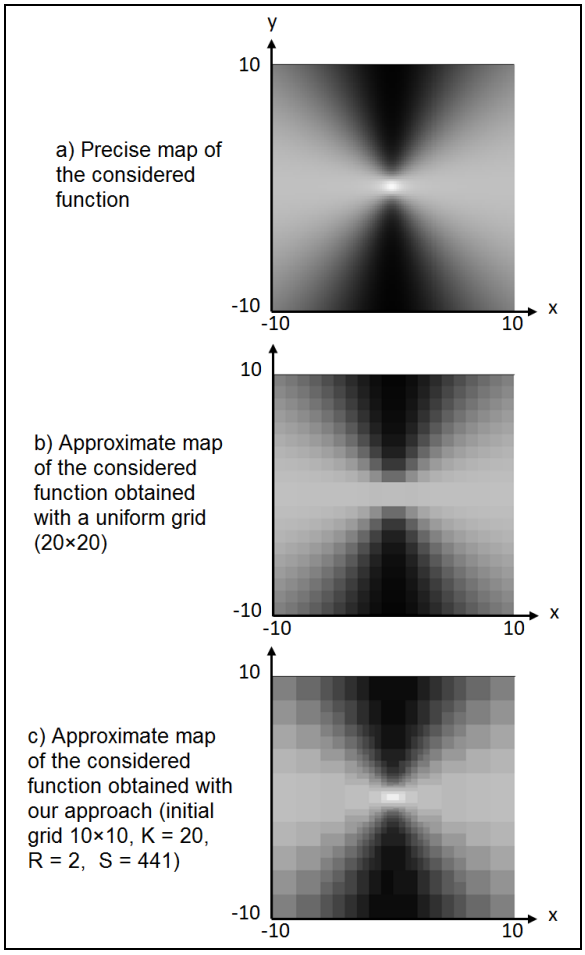


Figure 6: a) Precise map of the considered function; b) Approximate map of the considered function obtained with a uniform grid (20×20); c) Approximate map of the considered function obtained with our approach (initial grid 10×10, K = 20, R = 2, S = 441)

Figure 6a shows the precise map of this function. The darker, the higher the value of the function is. Figure 6b shows the map obtained with a uniform decomposition (grid 20 × 20, thus 441 simulations) and Figure 6c with our approach (we used the "*Average corner-value difference*" criterion, an initial grid of 10 ×10, K = 20, R = 2, S = 441). As shown by the figures, our approach allowed to make a more relevant map than the uniform decomposition. Indeed, the map made by our approach presents a more precise frontier between the black and the gray areas. Moreover, it allows to visualize the white area (minimum of the function) at the center of the map.

The second case-study we propose consists in testing our approach for a classic model: the ant foraging model (Resnick 2000). The goal is to have first evaluation of our approach for a stochastic model.

In this model, a colony of ants forages for food. When an ant finds a piece of food, it carries the food back to the nest and drops pheromone as it moves. When an ant smells the pheromone, it follows the pheromone.

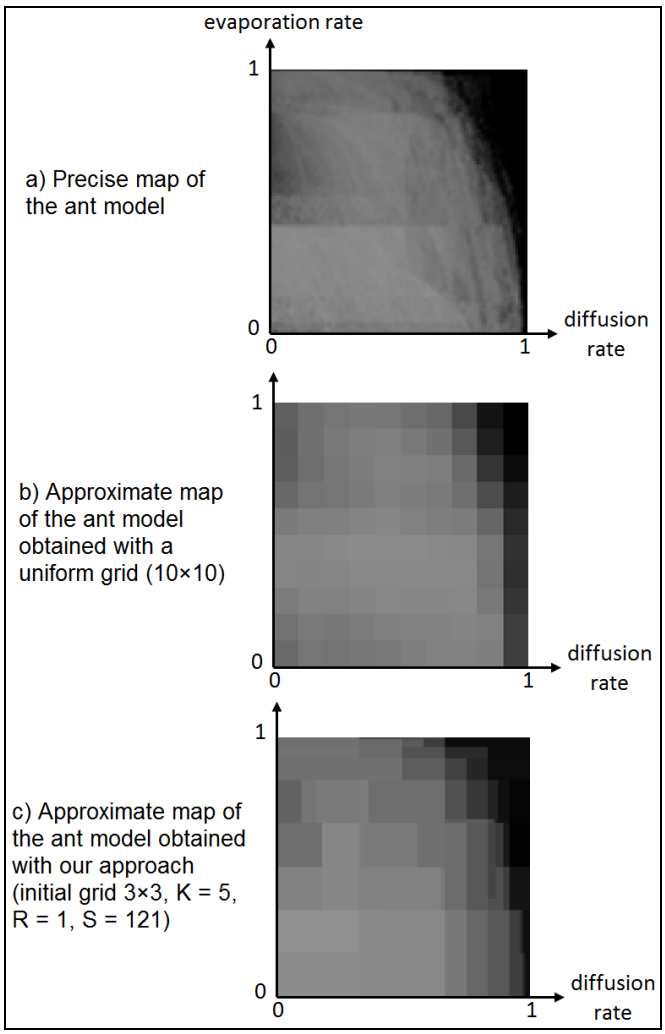


Figure 7: a) Precise map of the ant model; b) Approximate map of the ant model obtained with a uniform grid (10×10); c) Approximate map of the ant model obtained with our approach (initial grid 3×3, K = 5, R = 1, S = 121)

Two parameters are defined in this model:

- diffusion rate ∈ [0,1]: diffusion rate of the pheromone

- evaporation rate ∈ [0,1]: evaporation rate of the pheromone

We propose to study the influence of the diffusion and evaporation rates on the time necessary for the ant colony to carry all the food in the nest. We then make the map of the function *Time(diffusion_rate, evaporation_rate)*. As the model is stochastic (the movement of the ants, when no pheromone is detected, is random), we propose to carry 5 simulations and to compute the average *time* obtained to determine the global *time* of a set of parameter values.

Figure 7a shows the precise map obtained for the ant model. The darker, the higher is the time necessary to carry all the food to the nest. Figure 7b shows the map obtained with a uniform decomposition (grid 10 × 10, thus 121 × 5 simulations) and Figure 7c with our approach (we used the "*Average corner-value difference*" criterion, an initial grid of 3 ×3, $K = 5$, $R = 1$, $S = 121$). As shown by the figures, our approach allowed to make a more relevant map than the uniform decomposition. Indeed, the map made by our approach presents a far more precise frontier between the black and the gray areas in the right part of the map.

CONCLUSION AND FUTURE WORK

In this paper, we proposed a new approach to make the map of an agent-based model parameter space. Our approach is based on a reactive decomposition of the parameter space. We presented two case studies that show that our approach can allow to obtain, with the same number of simulations carried out, a more representative map than with a uniform decomposition.

The first experiments carried out concerned simple models. A first perspective is to apply our approach in the context of real models integrating a great number of parameters.

In the experiments, we used a simple criterion for the choice of the most "promising" cells. We would like to study and test more complex criteria, in particular criteria that take into account expert knowledge. Indeed, domain-experts that study a complex system often have knowledge about its dynamic and the impact of the different parameters. We propose to use this knowledge to define more relevant criteria.

As mentioned in a previous section, we would like to test other cell selection strategies. In particular, strategies that are not purely greedy. We can base these strategies on classic strategies used by optimization algorithms such as Simulated Annealing (Kirkpatrick 1983).

At last, we would like to work on the formalization of the output function. Indeed, in some context, the formalization of what the modeller wants to observe can be complex: for example, formalising the concept of "line of ants" in the ant foraging model is not straightforward. Thus, we propose to develop a new method based on the work we carried out concerning the definition of objective functions for optimisation problems (Taillandier and Gaffuri 2009). This approach is based on man-machine dialogue: several solutions (that can be simulation results) are presented to the user and are commented by him/her. The analysis of the comments allows to learn a function that translates the user needs.

REFERENCES

Abramson, D.; Bethwaite, B.; Enticott, C.; Garic, S. and Peachey, T. 2009. "Robust Workflows for Science and Engineering", *ACM Workshop (MTAGS)*.

Brueckner, S.A. and Parunak, H.V.D. 2003. "Resource-aware exploration of the emergent dynamics of simulated systems", *AAMAS '03*, 781-788.

Calvez, B. and Hutzler, G. 2007. "Ant colony systems and the calibration of multi-agent simulations: a new approach". In *Multi-Agents for modelling Complex Systems (MA4CS'07)*.

GAMA platform, 2011, http://gama-platform.googlecode.com [Accessed 27 July 2011]

Gramacy, R.B.; Lee, H.K.H. and Macready, W. 2004. "Parameter space exploration with Gaussian process trees", *Proceedings of the International Conference on Machine Learning*, 353-360.

Kirkpatrick, S.; Gellatt, C. and Vecchi, M.P. 1983. "Optimization by Simulated Annealing", *Science* 220.

Lee, B.; Schulz, M. and de Supinski, B. 2006. "Regression Strategies for Parameter Space Exploration: A Case Study in Semicoarsening Multigrid and R", Tech. Rep. UCRL-TR 224851, Lawrence Livermore National Laboratory, September.

Lorca, A.; Huedo, E. and Llorente, I.M. 2011. "The Grid[Way] Job Template Manager, a tool for parameter sweeping", *Computer Physics Communications* 182, Issue 4, 1047-1060.

Mitchell. M. 1996. An introduction to genetic algorithms. MIT Press, Cam- bridge, MA, USA.

Munos, R. and Moore, A. 2002 "Variable resolution discretization in optimal control", *Machine Learning* 49, 291-323.

Resnick, M. 2000. Turtles, termites and traffic jams. MIT Press.

Rogers, A. and Tessin, P.V. 2004. "Multi-objective calibration for agent-based models", *Workshop on Agent-Based Simulation*.

Rioux, F.; Bernier, F. and Laurendeau, D. 2008. "Multichronia – A Framework for the Exploration of Parameter, Simulation, Data and Visual Spaces", *The Interservice/Industry Training, Simulation & Education Conference*.

Taillandier, P.; Drogoul, A.; Vo, D.A. and Amouroux, E. 2010. "GAMA: bringing GIS and multi-level capabilities to multi-agent simulation", *The 8th European Workshop on Multi-Agent Systems*, Paris, France.

Taillandier, P. and Gaffuri, J. 2009. "Objective Function Designing Led by User Preferences Acquisition" *ICITA International Conference*, Ha noi, Viet Nam.

AUTHOR BIOGRAPHY

PATRICK TAILLANDIER graduated in artificial intelligence from the University of Lyon 1 (France) in 2005 and received his PhD degree in 2008 at the University Paris Est (COGIT lab – IGN). After working two years for the MSI research team (IFI – Hanoi, Vietnam) and one year for the SMAC research team (IRIT – Toulouse, France), he was recruited in 2011 as an associate professor by the University of Rouen.