Algorithmes évolutionnaires sur grille de calcul pour le calibrage de modèles géographiques

Romain Reuillon (1), Sebastien Rey (2), Clara Schmitt (3), Mathieu Leclaire (4), Denise Pumain (5)
(1) romain.reuillon@iscpif.fr, Géographie cité, CNRS UMR 8504
(2) rey.sebastien@parisgeo.cnrs.fr, Géographie cité, CNRS UMR 8504
(3) cl-schmitt@parisgeo.cnrs.fr, Géographie cité, CNRS UMR 8504
(4) mathieu.leclaire@iscpif.fr, Géographie cité, CNRS UMR 8504
(5) pumain@parisgeo.cnrs.fr, Géographie cité, CNRS UMR 8504

Overview
As dynamic geographic models integrate very large number of spatial interactions, large amount of computing is necessary for their simulation and calibration, in order to validate them. Here a new automated calibration procedure is experimented on the European computational grid EGI using evolutionnary algorithms. The application to the Simpoplocal model enables to reduce the computing time (one week) for managing about 7 millions runs for a preliminary validating step of the processes and parameters introduced in the model.

Résumé
Les modèles géographiques dynamiques mettant en jeu des interactions nombreuses entre des lieux exigent des masses importantes de calcul pour leur simulation et leur calibrage en vue de leur validation. Nous expérimentons ici une procédure de calibrage automatisé sur la grille de calcul européenne EGI fondée sur l'emploi d'algorithmes évolutionnaires. L'application au modèle Simpoplocal permet en un temps assez court (une semaine) de traiter les quelque 7 millions d'exécutions nécessaires à une première validation des processus et des paramètres introduits dans le modèle.

Enjeux scientifiques, besoin en calcul, stockage et visualisation
Les géographes modélisant les systèmes de villes s'appuient sur l'hypothèse que les interactions microgéographiques sont susceptibles de favoriser l'émergence de "dynamiques stylisées" aux échelles macro-géographiques, qui constituent une des caractéristiques récurrentes, "universelles", de ces systèmes complexes (Pumain 2007). L'hétérogénéité et le grand nombre de scenarii d'implémentation possibles pour représenter les processus en jeu, la description de dynamiques principalement individu-centrées, la non linéarité des interactions, la diversité des formes de relations spatiales et l'importance du contexte historique (Pumain et al. 2006, Sanders 2005), sont tout autant de raisons qui amènent les géographes à utiliser régulièrement des modèles agents (Agent Based Models) comme support de réflexion et d'expérimentation (Sanders 2007, Batty 2008). Tout l'enjeu d'une modélisation "réussie" en géographie est donc d'arriver à réunir un faisceau de mécanismes dans le but de valider, ou plutôt d'évaluer avec un intervalle de confiance élevé, un modèle pour une question donnée (Sargent 2010).

Dans un contexte de travail en géographie quantitative souvent inter-disciplinaire (Archeomedes: Durand-Dastès et al. 1998 , l'ANR Transmondyn[1], l'ERC Geodivercity[2]), le choix des mécanismes et des règles du modèle passe par une phase de dialogue et d'échange caractéristique de la co-évolution entre les concepts des thématiciens experts du domaine et des modélisateurs pilotant la construction d'un modèle. La construction d'un modèle sur ces bases interdisciplinaires engage bien souvent les modélisateurs dans une démarche de construction itérative et incrémentale. C'est sur cette base mouvante que nos outils et protocoles doivent se construire pour répondre au mieux à ces besoins d'expérimentation de nature itérative (Louail, Pumain, 2009). La réussite de cette construction est liée à la possibilité d'évaluation et au savoir expert qui doit être mobilisé à toutes les étapes de la construction et de l'expérimentation du modèle, lorsqu'il s'agit de confronter les résultats d'une instance d'un modèle par rapport à la question théorique qui l'a motivé.

Ce processus d'évaluation doit être rendu plus systématique qu'il n'était possible de le faire lors des premières applications de modèles multi-agents en géographie (Bura et al., 1996, Poix et Michelin, 2001). Nommée "face validity" selon un des principes fondateurs de la VV&T (Verification Validation and Testing) (Osman Balci 1998, Sargent 2005), cette étape permet de concrétiser les "semblants d'évaluation" ou les recherches de "comportements attendus raisonnables" par la définition d'indicateurs objectifs émanant des thématiciens experts du domaine (Sharma et al. 2006). C'est à la lecture de ces résultats, et au regard de la question initialement posée que les mécanismes doivent être pré-calibrés pour éviter une divergence incrémentale et grandissante entre l'implémentation (vérification interne) et les questions initialement posées (validation externe).

Le calibrage, ou du moins le pré-calibrage est un des nombreux problèmes inverses que se posent les modélisateurs (Bourgine et al. , 2008). Un problème inverse est une question sur les sorties d'un modèle qui vise à comprendre quelles

1 http://www.transmondyn.parisgeo.cnrs.fr/
2 http://geodivercity.parisgeo.cnrs.fr/blog/

valeurs des paramètres d'entrée permettent d'obtenir certaines dynamiques ou certains motifs. Quelle que soit la proximité attendue entre les résultats du modèle et des observations, que les modèles soient "guidés par les données" ou plus "théoriques" ou simplifiés, il est important de pouvoir déterminer si l'ensemble des paramètres estimés par le modèle à l'issue d'une simulation, en fonction des règles qui y ont été intégrées, correspond à une solution unique ou très probable, ou bien s'il ne s'agit que d'un "minimum local" que la dynamique a moins de chances de rencontrer dans le paysage vallonné de l'espace des phases des états du modèle. Cette phase de calibrage est généralement menée à bien par essai-erreur, et a conduit par exemple à se contenter d'une centaine de simulations lors de l'expérimentation sur différents systèmes de villes d'une version consolidée du modèle SIMPOP2 (Bretagnolle, Pumain, 2010) ; cependant, la non-linéarité des interactions et le fait que certains paramètres n'aient pas d'équivalence empirique, associés au fait que chaque nouvel incrément dans l'avancement du modèle puisse venir perturber l'ancienne dynamique, rendent alors très difficile et fastidieuse une calibration manuelle du modèle. Il est ainsi crucial d'automatiser le processus de calibrage et de substituer de manière temporaire l'expertise humaine sur le modèle par une expertise automatisée, opérant à la manière d'un filtre qui requiert une traduction quantitative de l'évaluation experte. Il est donc nécessaire d'envisager des méthodes et outils qui permettent de penser cette automatisation dans ce cadre original qu'est la calibration systématique de nos modèles.

Outils

L'automatisation du calibrage nécessite dans un premier temps la définition d'un certain nombre d'objectifs résumant la qualité d'un jeu de paramètres par rapport à des résultats attendus du modèle. Ces fonctions objectifs sont calculées à partir de l'exécution d'un ensemble de réplications d'une simulation utilisant un même jeu de paramètres (le modèle étant stochastique, chaque réplication correspond à un flux de nombres aléatoires indépendant de ceux des autres réplications). Ensuite, le processus de calibrage sélectionne les meilleurs jeux de paramètres d'après les valeurs atteintes pour ces objectifs. Pour calibrer des modèles multi-agents l'évaluation répétée des objectifs constitue donc une charge de calcul très importante qui ne peut être menée à bien que sur des environnements de calcul haute performance.

Du fait du nombre des paramètres et du caractère généralement continu de leur domaine de variation, l'habituel "parameter sweeping" distribué sur grille (Casanova, 2003), ou plan complet, est généralement inefficace pour calibrer les systèmes en haute dimension. D'une part, le nombre d'expériences à mener pour ce type de plan croit de manière exponentielle avec le nombre de paramètres à l'étude. D'autre part, un plan complet génère une grande quantité de données qu'il faut traiter et visualiser a posteriori. Le problème de recherche de motif dans l'espace des dynamiques du modèle est ainsi remplacé par un problème de recherche de motif dans une base de données. Le post-traitement de cette masse de données est cependant long et fastidieux.

L'impossibilité de recourir à ces méthodes exhaustives nous pousse ici à envisager le calibrage non plus comme une recherche de comportement satisfaisant parmi l'ensemble des possibles, mais bien comme objectif qu'il nous faut atteindre, moyennant un questionnement inverse : Existe-t-il une combinaisons de paramètres permettant de remplir les objectifs fixés ? L'observation a posteriori devient question a priori, puisque les objectifs préalablement définis pour faire état d'un résultat deviennent les nouveaux guides permettant d'alimenter une ou plusieurs méthodes d'optimisation issues de la méta-heuristique.

Toutes les méthodes d'optimisation permettant une recherche guidée dans un espace inconnu doivent jongler entre deux buts contradictoires, en réduisant le temps de recherche des solutions optimales tout en parcourant au mieux l'ensemble de l'espace porteur à la fois de bonnes et de mauvaises solutions. Pour notre problème, nous avons choisi d'utiliser les algorithmes génétiques (Holland, 1975) du fait qu'il proposent une recherche globale dans l'espace des paramètres et présentent des aspects naturellement parallèles qui peuvent être exploités sur grille de calcul. On trouve des exemples récents de calibrage automatisé de modèles qui soulignent l'intérêt d'une recherche globale dans l'espace des paramètres (via des algorithmes évolutionnaires) plutôt que par des approches de recherche locales (Solomatine, 1999). De plus, l'emploi récent d'algorithmes évolutionnaires pour calibrer des modèles ABM (Rogers et Tessin 2004) en géographie (Heppenstall et al. 2007) a prouvé toute l'utilité de ces techniques pour le calibrage de modèle.

L'utilisation de méthodes de recherche globale pour le calibrage de modèles est généralement extrêmement coûteuse en calcul. Même si des méthodes ont été développées pour réduire le nombre d'évaluation du modèle dans certains cas (Liu et al., 2005) l'utilisation d'algorithmes évolutionnaires pour le calibrage de modèles de systèmes complexes requiert l'utilisation de parallélisme massif.

Développements, utilisation des infrastructures

L'utilisation d'algorithmes génétiques pour le calibrage d'un modèle multi-agents (et plus particulièrement multi-agents stochastiques) demande une charge de calcul très importante qui ne peut être fournie que par des environnements de calcul massivement distribués.

Afin d'illustrer les méthodes expliquées ici, prenons l'exemple du modèle SimpopLocal. Celui-ci étudie la structuration hiérarchique de peuplements au moment de l'émergence des villes, quelque 3000 ans après celle de l'agriculture (Bairoch, 1985) en simulant la dynamique de croissance de systèmes de peuplement sous contrainte environnementale. A visée clairement exploratoire, ce modèle cherche d'abord à reproduire et à étudier dans des conditions particulières (fortes contraintes environnementales) un phénomène très bien décrit dans la littérature par un fait stylisé : l'organisation hiérarchique de systèmes de peuplements sous forme d'une distribution Rank-Size à partir des interactions spatiales (échanges d'information notamment) entre les unités de peuplement (Berry, 1964). Dans la mesure où le processus a pris des formes diverses selon les régions du monde, bien documentés par les travaux des archéologues (Marcus et Sabloff, 2008), le modèle Simpoplocal a été conçu comme une simplification abstraite, générique, de l'émergence des villes, à partir d'un très petit nombre de règles (5) et de paramètres (une dizaine). On ne dispose d'aucune information historique permettant d'attribuer des valeurs numériques à au moins 7 de ces paramètres.

Dans son implémentation la plus efficace une exécution de Simpoplocal dure 40s sur un processeur moderne. Le modèle étant stochastique, les variables de sorties sont aléatoires et l'évaluation d'un jeu de paramètres donné nécessite a minima 30 réplications indépendantes de ce modèle et donc l'utilisation de 1200s de temps processeur. Il est donc impératif de paralléliser les évaluations pour calibrer ce modèle.

De nombreuses méthodes ont été développées dans le but de paralléliser efficacement les algorithmes génétiques (Alba et al., 1999). Les algorithmes à état stationnaire ou "steady state" (Syswerda, 1991) présentent une alternatives aux algorithmes générationnels en prenant en compte les solutions à mesure qu'elles sont évaluées. Ils présentent à la fois de meilleurs propriétés de convergence (Durillo, 2008) et son caractère asynchrone lui procure une robustesse naturelle face aux hétérogénéités en temps de calcul des environnements de type grille.

Pour le problème de calibrage de SimpopLocal qui vise à satisfaire 3 objectifs, nous avons utilisé l'algorithme génétique NSGA II steady state (Syswerda, 1991, Deb, 2001). Cet algorithme a été implémenté au sein de la bibliothèque MGO[3] et distribué sur grille grâce à OpenMOLE (Reuillon, 2010)[4], deux logiciels libre écrit en scala.

Résultats scientifiques

L'application de cette démarche a donné des résultats probants pour répondre au problème de calibrage de SimpopLocal. Avec sept paramètres dont la valeur ne pouvait être estimée de façon empirique, la recherche d'un jeu de paramètres qui puisse satisfaire les trois objectifs de simulation (une forme de distribution de tailles de villes, une taille maximale et la durée nécessaire pour les structurer) s'avérait fastidieuse. La première expérience ayant abouti à des propositions de jeux de paramètres candidats au calibrage a nécessité la simulation de près de 6,9 millions d'exécutions du modèle, ce qui représente plus de 10 années consécutives de calcul. Ces chiffres impressionnants illustrent bien la difficulté de cette tâche de calibrage devant l'absence de connaissances a priori des comportements du modèle. On se demande si par le moyen d'une recherche manuelle, le même résultat aurait pu être trouvé avec moins d'expériences et autant de certitude... le test n'en vaut certainement pas le coût!

Par retro-projection des solutions de l'espace des objectifs dans l'espace des paramètres, cette procédure d'automatisation du calibrage du modèle a permis de trouver un ensemble de jeux de paramètres possibles amenant de bons résultats en termes de reproduction des faits stylisés, et donc contribuant à valider la qualité du modèle. Une évaluation multi-objectif ne conduit pas forcément à la sélection d'un jeu de paramètres unique et peut mener à un ensemble de candidats possibles au calibrage : chaque jeu de paramètres sélectionné par la procédure est un compromis pour chaque objectif. L'ensemble de ces candidats forment un "front de Pareto" (cette expression recouvre un ensemble de solutions dans lequel aucun des éléments n'est meilleur que les autres sur l'ensemble de ces objectifs). Néanmoins, parmi les jeux de paramètres proposés par la procédure, certains sont moins satisfaisants que les autres du point de vue des thématiciens. L'introduction de connaissance experte a posteriori dans cette procédure a permis d'affiner la fonction d'évaluation et d'opérer une discrimination bénéfique dans les jeux de paramètres candidats.

Ces techniques de calibrage automatique sont maintenant utilisées de manière systématique pour réaliser des séries d'expérimentations qui cherchent à préciser et à valider l'implication dans la dynamique du modèle de chaque nouveau mécanisme implémenté. Ces expérimentations visent à tester la qualité des solutions possibles en fonctions de l'activation / désactivation de mécanismes dans le modèle. On dispose dès lors d'un moyen de tester non seulement la validité des valeurs des paramètres, mais aussi d'évaluer la plausibilité des règles introduites dans le modèle.

3https://forge.iscpif.fr/projects/mgo
4http://www.openmole.org

Conclusion et perspectives

Le calibrage automatique fondé sur un algorithme évolutionnaire multi-objectifs nous a permis d'utiliser des environnements de calcul haute performance et notamment la grille de calcul pour produire un ensemble de solutions compromis. Ces solutions ont ensuite été analysées avant d'être exploitées afin de comprendre comment des dynamiques interagissent pour simuler la hiérarchisation progressive d'un système de villes. Cependant, dans le cas de Simpoplocal, un état vers lequel l'algorithme a convergé est atteint au bout d'une semaine en temps effectif de calcul sur la grille, temps nécessaire pour mener à bien les 10 années / CPU de simulation. Pour utiliser cette méthode de manière routinière lors de la conception itérative de modèles de systèmes complexes, il est certainement possible d'améliorer encore le temps global de calcul en utilisant des algorithmes évolutionnaires plus récents que NSGA 2 comme MO-CMA-ES (Igel et al., 2007) qui pourraient s'avérer plus efficaces ou encore en développant des stratégies de distributions de populations sur un modèle d'îlots (Whitley, 1997) qui sont particulièrement efficaces dans un contexte de calcul distribué.

Remerciements

Ce travail est financé par l'ERC Geodivercity, l'Agence de l'Environnement et la Maitrise de l'Energie (ADEME), le réseau Réseau de Recherche sur le Développement Soutenable (R²DS), l'Institut des Systèmes Complexes Paris Ile-De-France (ISC-PIF)

Références

Alba, E. & Troya, J. M. A survey of parallel distributed genetic algorithms. *Complex.* 4, 31–52 (1999).

Balci, O. Verification, Validation, and Testing. *Handbook of Simulation* 335–393 (2007).

Batty, M. Fifty Years of Urban modelling : Macro-Statics to Macro-Dynamics. *The Dynamics of Complex Urban Systems* 1–21 (2008).

Berry, B. J. L. Cities as systems within systems of cities. *Papers in Regional Science* 13, 147–163 (1964).

Bourgine, P. *et al.* French Roadmap for complex Systems 2008-2009. *arXiv:0907.2221* (2009).

Bretagnolle, A. & Pumain, D. Simulating Urban Networks through Multiscalar Space-Time Dynamics: Europe and the United States, 17th-20th Centuries. *Urban Studies* 47, 2819–2839 (2010).

Bura, S., Guérin-Pace, F., Mathian, H., Pumain, D. & Sanders, L. Multiagent Systems and the Dynamics of a Settlement System. *Geographical Analysis* 28, 161–178 (1996).

Casanova, H. & Berman, F. Parameter Sweeps on the Grid with APST. *Grid Computing* 773–787 (2003).

Deb, K., Agrawal, S., Pratap, A. & Meyarivan, T. A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. Parallel Problem Solving from Nature PPSN VI 1917, 849–858

Durand-Dastès, F. et al. Archaeomedes : Des oppida aux métropoles. (Anthropos: Paris, 1998).

Durillo, J. J., Nebro, A. J., Luna, F. & Alba, E. A study of master-slave approaches to parallelize NSGA-II. IEEE International Symposium on Parallel and Distributed Processing, 2008. IPDPS 2008 1 –8 (2008).doi:10.1109/IPDPS.2008.4536375

Heppenstall, A., Andrew, E. & Birkin, M. Using Hybrid Agent-Based Systems to Model Spatially-Influenced Retail Markets. (2006).

Holland, J. Adaptation in Natural and Artificial Systems. (University of Michigan Press: 1975).

Igel, C., Hansen, N. & Roth, S. Covariance Matrix Adaptation for Multi-objective Optimization. *Evolutionary Computation* 15, 1–28 (2007).

Liu, Y. & Ye, W.-J. Time Consuming Numerical Model Calibration Using Genetic Algorithm (GA), 1-Nearest Neighbor (1NN) Classifier and Principal Component Analysis (PCA). *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the* 1208 –1211 (2005).doi:10.1109/IEMBS.2005.1616641

Louail, T. & Pumain, D. Interaction des ontologies informatique et géographique pour simuler les dynamiques multiscalaires. *Rochebrune'09 : Ontologie et dynamique des systèmes complexes, perspectives interdisciplinaires* (2009).

Marcus, J. & Sabloff, J. A. *The ancient city: New perspectives on urbanism in the old and new world*. 2005, (School for Advanced Research: Santa Fe, 2008).

Michelin, Y. & Poix, C. Simulation paysagère : un modèle multi-agents pour prendre en compte les relations sociales. *Cybergeo : European Journal of Geography* (2000).doi:10.4000/cybergeo.2242

Pumain, D., Bretagnolle, A. & Daudé, E. From theory to modelling : urban systems as complex systems. *Cybergeo : European Journal of Geography* (2006).

Pumain, D. Une approche de la complexité en géographie. *Géocarrefour* 78, 25–31 (2007).

Reuillon, R. *et al.* Declarative task delegation in OpenMOLE. *2010 International Conference on High Performance Computing and Simulation (HPCS)* 55 –62 (2010).

Rogers, A. & von Tessin, P. Multi-Objective Calibration For Agent-Based Models. (2004).

Sanders, L. Intelligence artificielle et agents collectifs : le modèle EUROSIM. (2005).

Sanders, L. Objets géographiques et simulation agent, entre thématique et méthodologie. *Revue Internationale de Géomatique* 17, 135–160 (2007).

Sargent, R. G. Verification and validation of simulation models. *Simulation Conference (WSC), Proceedings of the 2010 Winter* 166 –183 (2010).doi:10.1109/WSC.2010.5679166
Sargent, R. G. Verification and validation of simulation models. *Proceedings of the 37th conference on Winter simulation* 130–143 (2005).
Solomatine, D. P., Dibike, Y. B. & Kukuric, N. Automatic calibration of groundwater models using global optimization techniques. *Hydrological Sciences Journal* 44, 879–894 (1999).
Syswerda, G. A study of reproduction in generational and steady-state genetic algorithms. *Foundations of Genetic Algorithms* 94–101 (1991).
Vimal Sharma, Swayne David, David Lam & William Schertzer *Auto-Calibration of Hydrological Models Using High Performance Computing*.
Whitley, D., Rana, S. & Heckendorn, R. B. Island Model Genetic Algorithms and Linearly Separable Problems. *In Evolutionary Computing, AISB Workshop* 109–125 (1997).