Brigham Young University
BYU ScholarsArchive

International Congress on Environmental
Modelling and Software

8th International Congress on Environmental
Modelling and Software - Toulouse, France - July
2016

Jul 11th, 9:50 AM - 10:10 AM

Advances in probabilistic and parallel agent-based simulation: Modelling climate change adaptation in agriculture

Christian Troost
Universität Hohenheim, christian.troost@uni-hohenheim.de

Thomas Berger
Universität Hohenheim, 490e@uni-hohenheim.de

Advances in probabilistic and parallel agent-based simulation: Modelling climate change adaptation in agriculture

Christian Troost[a], Thomas Berger[b]
[a] *Universität Hohenheim (christian.troost@uni-hohenheim.de)*
[b] *Universität Hohenheim (thomas.berger@uni-hohenheim.de)*

Abstract: MPMAS is a software package for farm-level and agent-based simulation in agriculture that has been applied to a variety of agro-economic and bioeconomic case studies around the world in the last twenty years. We present recent advances to the software and its application that focus on uncertainty and sensitivity analysis, massive parallel computing, land market interactions and farm succession. These extensions improve the applicability of agent-based simulation for ex ante assessments of climate adaptation in agriculture.

Keywords: *Massive parallel computing; Uncertainty and sensitivity analysis; Structural change; Land markets; Farm succession*

1   INTRODUCTION

Increasingly, scholars interested in the analysis of climate change adaptation in agriculture consider household-level and agent-based modelling (ABM) to complement integrated assessment models and models of global economic trade. The main reasons for this growing interest in ABM are that it (i) offers a highly disaggregate "process-based" representation of human decision-making; (ii) does not exclusively rely on past empirical data and is therefore better suited for "out-of-sample" simulation; and (iii) can explicitly capture interactions and feedbacks between socioeconomic and biophysical processes (van Wijk 2014; Berger & Troost 2014; van Wijk et al. 2014).

MPMAS (Schreinemachers & Berger 2011) is a software package for agent-based simulation that stands in the agricultural economics tradition of recursive farm modelling and adaptive micro-systems (Berger & Troost 2012). Following its initial application to the analysis of trade liberalisation in Chile (Berger 2001), MPMAS has been applied in a variety of case studies around the world focusing on innovation diffusion (Schreinemachers et al. 2010), soil conservation (Marohn et al. 2013; Quang et al. 2014), irrigation water use (Arnold et al. 2015), pesticide use (Bannwarth et al. 2016) climate impacts (Wossen et al. 2014; Troost & Berger 2015a), and policy analysis (Troost et al. 2015). Since simulating climate change adaptation involves longer time horizons and substantial model uncertainty, the source code of MPMAS has been extended in several respects. The present article describes these recent extensions of MPMAS that focus on improving its applicability for integrated assessment of climate adaptation in agriculture: (i) Massively parallel computing; (ii) Agent interactions on land rental markets; and (iii) Common random number schemes for farm succession. We show that with these new features ABM can be applied successfully to stochastic integrated assessment and policy analysis.

2   MODEL UNCERTAINTY TESTING

As argued by Berger and Troost (2014), bottom-up farm level models are inevitably subject to considerable model parameter uncertainty, which should be clearly communicated by reporting model results for the full range of potential parameter settings. In addition, to avoid over-fitting the model and

deteriorating its out-of-sample properties especially for climate change simulation, Troost and Berger (2015a) suggest evaluating the full space spanned by the uncertain model parameters and not calibrating the model for perfect fit to one single point observation.

The number of repetitions that can in practice be evaluated is evidently a function of the available computer resources, the time constraints of the research project and the run-time of the model. The run-time of an agent-based model depends on five main factors: The number of agents, the number of simulation periods, the complexity of the agent decision problems, how often these agent decision problems have to be solved per simulation period, and the numeric solver used for solving the decision problems. In the case of MPMAS, run-time ranges between a few minutes and a week or more. An efficient sampling scheme is therefore paramount to represent the uncertain parameter space in as few model runs (and as little model run-time) as possible. Berger et al. (2015) achieve this by using a Sobol' sequence, a quasi-random sampling approach that tends to converge considerably faster than standard Monte-Carlo methods (Tarantola et al. 2012). When testing the various scenarios to be analysed in their study, Berger et al. (2015) found that convergence of differences in agent incomes was reached within 100 repetitions. Since each scenario was simulated using the same Sobol' sequence of parameter vectors, each point of the sequence provided a fully controlled experiment that isolated the scenario effect on each individual agent from any variation in other parameters. Mean effects and confidence intervals could therefore be calculated directly from the simulated distribution of the scenario effect over all points of the sequence.

3   MASSIVELY PARALLEL COMPUTING

Following the traditional agricultural economic approach (Berger & Troost 2012), MPMAS represents the production decision of each farm agent as a mixed-integer programming problem (MIP). Agents have to solve their MIP at least twice per period to make an investment and a production plan. If the simulations, in addition, consider crop yield and price variations or collective marketing schemes (producer organisations), agents have to solve their MIP after harvest one or two additional times. If land market interactions are to be included, agents will have to solve their MIP for every parcel (often dozens) that is offered to them on the land market. Solving the MIP is the most time-consuming part of the simulation in typical applications with fairly complex decision problems. So far, MPMAS sequentially solved the decision problems of one agent after the other, although conceptually each type of decision is assumed to be taken by all farm agents at the same time. Interactions between agents take place before all agents have taken a decision or after they have taken it.

This program structure can be exploited to parallelise agent decisions when several processors are available. Massively parallel computing has been implemented in MPMAS using OpenMPI (Gabriel et al. 2004). Figure 1 shows the program flow in the parallelised version of MPMAS. As can be seen, all agents are initialized and maintained on all processors. Whenever an agent has to take a decision, only one of the processors solves this decision problem and broadcasts the results to all other processors. This means, only the agent decisions have been parallelised in MPMAS, all other updating of agents is done on all processors. The reason why each processor maintains its own copy of every agent is that the number of agents changes over time (due to agent exits) and between decisions (in land markets only agents bidding on the same parcel can make their decisions in parallel). To allow efficient parallel computing, assignment of agents to processors therefore has to be flexible. Agents in MPMAS are rather large and complex memory structures. Communication from one processor to the other (including serialising and de-serialising the object), would otherwise consume considerable amounts of run-time.

Figure 2 shows the speed-up in simulation run-time that can be achieved by using several processors in parallel. Timings are for simulation with the Troost & Berger (2015a) model over ten simulation years and without land markets, using IBM-CPLEX as MIP solver. We can see that parallel computing reduces run-times considerably. We can, however, also appreciate the sequential overhead in the model that cannot be parallelised and leads to decreasing marginal reductions in run-time the more processors we add. If many repetitions are needed for uncertainty analysis (Troost & Berger 2015a,b) and the run-time of one repetition on one processor is within the upper time limits given by the queuing system, *trivial parallelisation*, i.e. using one processor for each scenario repetition, is more efficient than internal parallelisation with MPI, i.e. using several processors for the same run. Massive

parallelisation becomes essential if computing time for a single run exceeds the maximum job time limit on the computing resource as is the case when activating the land market in MPMAS.
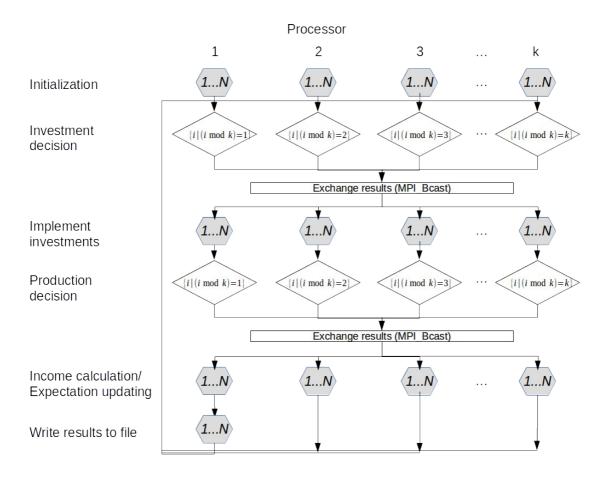

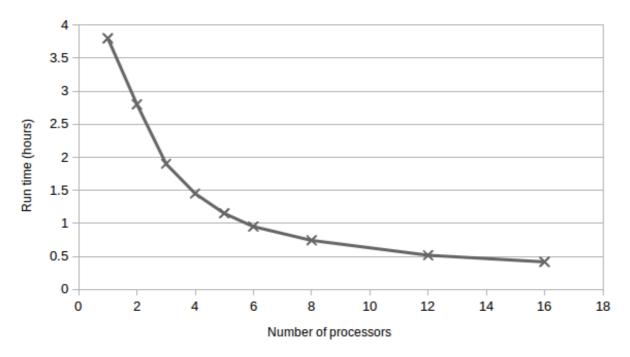**Figure 1:** Parallelisation of agent decision-making and agent interaction in MPMAS

Figure 2: Model run-times for the Troost & Berger (2015a) model over ten simulation periods (bwUniCluster)

4   AGENT INTERACTIONS ON LAND RENTAL MARKETS

In the original land market implementation of Berger (2001), each plot (pixel) in MPMAS corresponds to one parcel and is traded independently on the land rental market. To identify potential suppliers and renters of plots, each agent determines its shadow price for each soil type and compares it with the average shadow price of all agents. Agents will offer those plots for renting-out, for which the shadow price times a *markup* parameter minus the transport cost from farmstead to plot is lower than the average shadow price over all agents. Agents that decide not to offer plots of a given soil type will try to rent them in as long as their individual shadow price multiplied with a *markdown* parameter and corrected by internal transport costs is greater than the average shadow price. MPMAS then goes through all offered plots and identifies the bidder with the highest bid. If this bid is higher than the minimum rent expected by the owner, a rental contract is made and the rental payment is set to the average between bid and expected minimum rent. Both, owner and tenant compute their new shadow prices and decide whether to offer or bid for further plots.

For simulations of climate change adaptation in Germany, this original MPMAS land market implementation has been completely revised. To allow for long-term rental contracts, rental contracts can now have durations longer than a year, the shadow price is then assessed using the investment decision problem, and the model can be initiated with existing contracts. Further, parcels consisting of several adjacent plots (pixels) of the same owner are now traded as a whole and not pixel-by-pixel to better represent common practice in Germany. Now, when making their bid for a parcel, agents also take into account that this particular parcel is probably not the only parcel on offer and that they might find and rent another cheaper or closer parcel. The bid for an offered parcel is determined such that the agent is indifferent between the benefit of renting this parcel and the expected value of instead trying to rent another equivalent parcel at the average rental price observed, assuming that an equivalent other parcel can be hired for the average rental price with fixed probability. The agent that offers the parcel will accept the highest bid as long as it is higher than its reserve price, a certain fraction of the average rental price. The rental price "agreed" upon between the agents lies between the highest and the second-highest bid (or in the absence of a second bid, the reserve price of the landlord). The exact location depends on the bargaining power the modeller assumes each party to have.
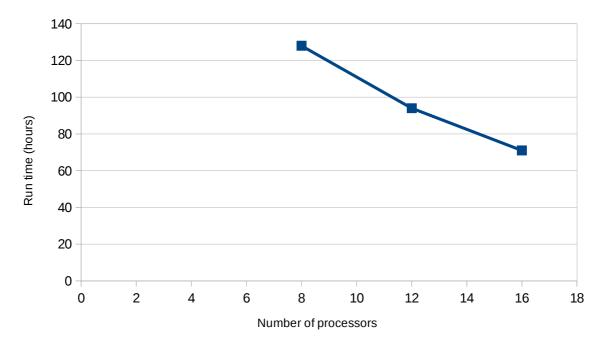

Figure 3: Model run-times for the Troost & Berger (2015a) model over ten simulation periods including land rental markets (bwUniCLuster)

Clearly, agent interactions on the land rental market increase the model run-time, since many bids have to be computed throughout the auction process, each time solving a new MIP to determine the agents' shadow prices for the parcel offered. The simulation experiment presented above with 10 years, 520 agents but including land markets would require more than 10 days on one processor. This goes far beyond the maximum run-time limit of 6 days on *bwUniCluster*, the high performance computing resource we are currently using for MPMAS simulation. With massively parallel computing, however, land market interactions can be simulated within the maximum run-time limit. Figure 3 shows the speed-up in simulation run-time that can be achieved for land market interactions.

5   COMMON RANDOM NUMBER SCHEMES FOR FARM SUCCESSION

The long-term fate of family farm holdings is not only a matter of economic profitability, but also of demographic factors: In developing countries, population growth plays a major role in causing unsustainable production patterns. In the European context, the existence of a successor that can and wants to take over the farm holding when the current farm manager retires or deceases is crucial for the continuity of the farm business. The existence of a potential successor, who in the vast majority of cases stems from the farm family itself, may also exert a strong influence for investment and production decisions long before the current farm manager retires. Hence, retirement age and mortality rate of the farm manager, birth rate and the willingness of children to take over the farm are important determinants of farm survival. Though partly these variables may be affected by the economic condition of the farm holding, they do not admit to a fully deterministic explanation and are partly or entirely modelled as stochastic processes in MPMAS. Each farm agent household member is associated with a "career path" (e.g. male household head or female unskilled worker) that defines age-specific probabilities of dying, giving birth, marrying, leaving and retiring as well as other characteristics such as labour provision and food demand. Agents may switch "career paths" during simulation, e.g. children may become household heads.

MPMAS uses the inverse transform method (Law 2007) to generate stochastic events: A uniform random variate is generated from a pseudo-random number generator (RNG) and then the inverse cumulative distribution function of the actual probability distribution is employed to determine the actual outcome. The "choice" of the outcome depends on two components: the probability distribution for the outcome, and the uniform random variate drawn. Without further control imposed on the RNG, differences in simulation outcomes between two scenarios may result from two sources: a shift in probabilities due to some actual scenario effect or as a pseudo-effect due to the use of a different variate generated by the RNG. To avoid drawing false conclusions from confusing such pseudo-effects with actual scenario effects, one option is to simulate sufficient repetitions with different initialization seeds for the RNG, so that positive and negative pseudo-effects cancel out when calculating the average scenario effects over all repetitions. In combination with many agents, many repetitions for analysis of epistemic uncertainty (Troost & Berger 2015a,b) and many scenarios, however, the computational burden necessary to simulate enough repetitions usually becomes prohibitive. Alternatively, necessary repetitions can be reduced employing the common random numbers (CRN) variance reduction technique (Law 2007) making sure that the same random variate is used for the same decision in each scenario. This makes each repetition with the same RNG seed an experiment that has been fully controlled, i.e. differences observed are due to differences in shifts of probability distributions and not differences in "randomness".

The deterministic nature of RNG ensures that the value of the *i*th variate in a random number sequence is always the same given equal initial seeds for the RNG. Still, ensuring that the same position in the sequence is used for the same decision in the model is not trivial in a simulation that mixes random events for many farm households composed of many members with deterministic, scenario-dependent economic decisions that may also alter event probabilities. The exit of a farm holding agent in one scenario means that variates for subsequent farm agents will be taken from a different position in the RNG stream than in another scenario where it does not exit. The same is true for exits of individual household members (for illustrative examples in a different modelling context see Stout & Goldie 2008).

Practically, the use of CRN in such a setting of many individuals can be achieved by reserving non-overlapping sequences within one larger random number stream for individuals and types of decisions. This could, for example, be done by pre-drawing a reservoir of random sequences for each

individual and type of decision before simulation start and feed it as input into the simulation (Stout & Goldie 2008). In our case with many new household members that may or may not be generated by birth or marriage during simulation, we might end up pre-drawing many more numbers than actually needed, occupying unnecessarily large amounts of memory. Alternatively, we can pre-determine only the position in the stream from which the random variate should be taken and generate the corresponding number only when it is actually needed. In addition to the common requirements for a good random number generator (long recurrence period, uniformity and space-filling in multiple dimensions), such an approach demands a RNG that provides the possibility of fast-forwarding it to a given position in the stream.

Like Stout & Goldie (2008), we chose *RngStream*, a C++-software library presented by L'Ecuyer et al. (2002), for our implementation of common random numbers in MPMAS. *RngStream* is based on the `MRG32k3a` RNG suggested by L'Ecuyer (1999), whose recurrence period of 2^{191} is divided into 2^{64} streams of 2^{127} random numbers, which can each further be subdivided into 2^{51} substreams of length 2^{76}. The substream starting points were chosen by L'Ecuyer et al. (2002) such as to maximize uniformity and independence of random numbers taken from the same position in different substreams. Corresponding positions in other streams can be reached from a given state by subsequently applying pre-calculated transition matrices for jumping 2^{127}, respectively 2^{76} numbers ahead.

For the application of CRN in MPMAS, we associate each household member with one of the 2^{51} substreams contained in one stream. The association is based on a 48bit code. The first 24 bits encode a unique numeric farm household ID that identifies agent farms throughout the whole simulation project, and might e.g. be based on a survey observation ID when modelling farms one-to-one or defined during the generation process of a synthetic model population (Berger &. Schreinemachers 2006). The second 24 bits are used to encode the farm household members based on their descendance from household members at model initialization. Table 1 illustrates the scheme with some examples: The initial members receive an unique ID encoded in the last five bits (OM). This allows 32 initial household members, which should be plenty. The preceding two bits (P0) are reserved for partners of the initial members that enter the household during simulations. The preceding four bits (C1) for the children born to initial household members or their partners and the 2 bits before that (P1) for the partners of the latter, and so on. Children are only born to female members and therefore only associated to the female part of a couple. Using this descendance-based encoding to associate substreams to household members ensures that the first child of household member *j* in household *i* will always have the same gender and career assuming probabilities remain constant. It does not depend on whether any other household member married earlier or gave birth earlier as would be the case if the association to substreams was based on the order in which members ``joined'' the household.

Table 1: Composition of the 24-bit household member code

	Bits							
	23	19-22	17-18	13-16	11-12	7-10	5-6	0-4
Examples	P3	C3	P2	C2	P1	C1	P0	OM
Original household member with ID 3	0	0000	00	0000	00	0000	00	0011
His wife married during the simulation	0	0000	00	0000	00	0000	01	0011
The second child born to the wife	0	0000	00	0000	00	0010	01	0011
The partner of that child	0	0000	00	0000	01	0010	01	0011
Third child born to the child	0	0000	00	0011	00	0010	01	0011

To allow fast-forwarding to the substream designated to a specific household member, transformation matrices for all powers of 2 between 2^{76} and $2^{76} * 2^{48} = 2^{124}$ have been pre-calculated in addition to the ones provided by L'Ecuyer (2002). In this way, the household member substream can be reached by subsequently applying the transformation matrices for the powers of 2 corresponding to the bits in the member code that are equal to one. From each household member substream, random numbers are drawn in the moment the member is generated (at initialization or when it joins the household) in the following order (see Table 1). First, one random variate is drawn for each event assumed to occur only

once per simulation. Second, random variates for three potential marriages are drawn to determine the career of the member after marriage and the career of the new partner. The next fifteen random variates of the stream are used to determine the careers of potential children. Finally, the four yearly-recurring determinants for potential death, birth, marriage, or leaving are taken. Since all four variates for one year are taken before the ones for the subsequent years, the same events are guaranteed to occur to the same household member in the same year irrespective of the total simulation time, as long as the household member does not switch careers.

Variation of event sequences between runs can be achieved by using different seeds for the random number generator, i.e. choosing a different beginning for the overall stream. Further, the farm encoding scheme also leaves a lot of room to shift the beginning within the stream: Using 24 bits for the farm encoding, the model would be able to accommodate substreams for $2^{24} = 16,777,216$ farm household IDs, which is far from the maximum that has been used with MPMAS so far (~ 10,000 agents). E.g. adding a constant multiple of the number of farm agents to the farm agent identifier before generating the 24-bit farm code allows different non-repeating substreams to be used within the stream. In combination with the MPI parallelization the CRN scheme also ensures that the same event sequences are generated on each processor. A simulation study that employs this CRN scheme for the simulation of structural change in Germany is presented in a parallel article submitted to session D6 of this conference.

6   CONCLUSIONS

Simulating climate change adaptation in agriculture involves time horizons of about 15-20 years, as farm managers may not change their short-term crop mix only but also adopt cost intensive and far-reaching adaptation measures such as investment into new machinery, switching from rainfed to irrigated cropping systems, acquiring more land from other farmers or, alternatively, take on off-farm employment or close down their farm business.

Longer time horizons with agent interactions on land markets and intra-household decisions on farm succession add substantial model uncertainty to ABM that can only be tackled with efficient sampling schemes and sufficient computing power. We showed in this paper that by introducing an MPI-layer into our agent-based software MPMAS together with a common random-number scheme the computational burden can be shouldered successfully. Work is ongoing on coupling MPMAS with the biophysical software package EXPERT-N (Biernath et al., 2011),which has also been MPI enabled recently. Together these technological advances in the software package pave the way for a practically applicable and methodologically sound use of ABM in the analysis of climate change adaptation in agriculture.

REFERENCES

Arnold, R. T., Troost, C., Berger, T., 2015, Quantifying the economic importance of irrigation water reuse in a Chilean watershed using an integrated agent-based model. Water Resources Research 51, 648-66.

Bannwarth, M., Grovermann, C., Schreinemachers, P., Ingwersen, J., Lamers, M., Berger, T., Streck, T., 2016, Non-hazardous pesticide concentrations in surface waters: An integrated approach simulating application thresholds and resulting farm income effects. Journal of Environmental Management 165, 298 - 312.

Berger, T., 2001, Agent-based Spatial Models Applied to Agriculture: A simulation tool for technology diffusion, resource use changes and policy analysis. *Agricultural Economics* 25, 245-260.

Berger, T., Schreinemachers, P., 2006, Creating agents and landscapes for multiagent systems from random samples. *Ecology & Society* 11, 19.

Berger, T., Troost, C. 2014. Agent-based modelling of climate adaptation and mitigation options in agriculture. Journal of Agricultural Economics 65(2), 323-348.

Berger, T., Troost, C., 2012. Agent-based Modelling in the Agricultural Economics Tradition of Recursive Farm Modelling and Adaptive Micro-Systems. In: Seppelt, R., Voinov, A., Lange, S., Bankamp, D. (Eds.), International Environmental Modelling and Software Society (iEMSs): 2012 International Congress on Environmental Modelling and Software. Managing Resources of a Limited Planet, Sixth Biennial Meeting, Leipzig, Germany.

Berger, T., Wossen, T., Troost, C., Latynskiy, E., Tesfaye, K., Gbegbelegbe, S., 2015, Adaptation of farm-households to increasing climate variability in Ethiopia: Bioeconomic modeling of innovation diffusion and policy interventions. *International Association of Agricultural Economists, 2015 Conference, August 9-14, 2015, Milan, Italy.* http://purl.umn.edu/229062

Biernath, C., Gayler, S., Bittner, S., Klein, C., Högy, P., Fangmeier, A., Priesack, E., 2011, Evaluating the Ability of Four Crop Models to Predict Different Environmental Impacts on Spring Wheat Grown in Open-Top Chambers. *European Journal of Agronomy* 35, 71 – 82.

Gabriel, E., Fagg, G.E., Bosilca, G., Angskun, T., Dongarra, J.J., Squyres, J.M., Sahay, V., Kambadur, P., Barrett, B., Lumsdaine, A., Castain, R.H., Daniel, D.J., Graham, R.L., Woodall, T.S., 2004. Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation. Proceedings of 11th European PVM/MPI Users' Group Meeting, Budapest, Hungary.

L'Ecuyer, P., 1999, Good parameters and implementations for combined multiple recursive random number generators. Operations Research 47(1) 159-164.

L'Ecuyer, P., Simard, R., Chen, E. J., Kelton, W. D., 2002. An object-oriented random-number package with many long streams and substreams, Operations Research 50(6), 1073-1075.

Law, A. M. 2007. Simulation Modeling & Analysis - International Edition, 4th Edition, McGraw-Hill, Boston.

Marohn, C., Schreinemachers, P., Quang, D. V., Berger, T., Siripalangkanont, P., Nguyen, T. T., Cadisch, G., 2013, A software coupling approach to assess low-cost soil conservation strategies for highland agriculture in Vietnam. Environmental Modelling & Software 45, 116 – 128.

Quang, D. V., Schreinemachers, P., Berger, T., 2014, Ex-ante assessment of soil conservation methods in the uplands of Vietnam: An agent-based modeling approach . Agricultural Systems 123, 108 – 119.

Schreinemachers, P., Berger, T., 2011**,** MP-MAS: An agent-based simulation model of human-environment interaction in agricultural systems. Environmental Modelling & Software 26, 845-859.

Schreinemachers, P., Potchanasin, C., Berger, T., Roygrong, S., 2010, Agent-based modeling for ex-ante assessment of tree crop innovations: Litchis in northern Thailand. Agricultural Economics 41, 519-536.

Stout, N. K., Goldie, S. J., 2008. Keeping the noise down: common random numbers for disease simulation modeling, Health Care Management Science 11(4), 399-406.

Tarantola, S., Becker, W., Zeitz, D., 2012, A comparison of two sampling methods for global sensitivity analysis. *Computer Physics Communications* 183, 1061-1072.Troost, C., Berger, T., 2015a. Dealing with uncertainty in agent-based simulation: Farm-level modeling of adaptation to climate change in Southwest Germany. American Journal of Agricultural Economics 97(4), 833-854.

Troost, C., Berger, T., 2015b. Process-based simulation of regional agricultural supply functions in Southwestern Germany using farm-level and agent-based models. International Association of Agricultural Economists, 2015 Conference, August 9-14, 2015, Milan, Italy. http:// purl.umn.edu/211929

Troost, C., Walter, T., Berger, T., 2015, Climate, energy and environmental policies in agriculture: Simulating likely farmer responses in Southwest Germany. Land Use Policy 46, 50 - 64

van Wijk, M. T., 2014. From global economic modelling to household level analyses of food security and sustainability: How big is the gap and can we bridge it? Food Policy 49, Part 2, 378-388.

van Wijk, M., Rufino, M., Enahoro, D., Parsons, D., Silvestri, S., Valdivia, R., Herrero, M., 2014, Farm household models to analyse food security in a changing climate: A review. Global Food Security 3, 77-84.

Wossen, T., Berger, T., Swamikannuh, N., Ramilan, T., 2014: Climate variability, consumption risk and poverty in semi-arid Northern Ghana: Adaptation options for poor farm households. Environmental Development 12, 2-15.