Creating a model of the Earth System (MOTES): some experiences with parallel ABM

M.Bithell[1]

[1] University of Cambridge, department of Geography, Downing place, Cambridge, CB2 3EN, England
mike.bithell@geog.cam.ac.uk

Abstract. It is argued that certain kinds of problem in modern society impose a requirement on ABM to represent all human agents on the planet. Eleven reason are given, including the need for realism in social models, the importance of boundary conditions and scale, the need for global social justice, the existence of global scale dynamics created by and impacting directly on individuals, and the need to interface with other global models in order to address pressing problems such as climate change and ecosystem destruction. An indication of the difficulties involved in creating such models is given, drawing on experience of creating models with RepastHPC. The paper concludes by suggesting that rather than creating ever more model platforms and frameworks, what we need is a series of shared and collectively developed models, in a similar way to existing traffic models or, models of the physical parts of the earth system.

Keywords:Climate Change, Global models, Parallel processing, RepastHPC

1 Introduction

From a certain point of view, one might regard society as a massively parallel asynchronous processor of materials and information. Whether reasonable or not, it is perhaps a good description of the way ABM represent social systems, except that the majority of ABM at the present time are probably not working in parallel. For many models, particularly those testing concepts and ideas, or where the number of agents is quite small (less than about 1000 for Netlogo) and any spatial domain is not particularly highly resolved (so that the number of model grid patches is quite small), this is clearly fine. However, in this paper I argue that for certain classes of model, not only do we need models that work at large-scale (that is, with very large numbers of agents, acting in large highly resolved spatial domains), but we need them to work on the planet as a whole. In the next section suggestions are given as to why these global models are required. This is followed by a fairly high-level overview of the issues involved in making parallel models, and a description of using RepastHPC to model at scale. The conclusion reflects on how this could help, at least in part, to deal with the continual re-invention of ABM that currently hinders progress in the field.

2 Why we need global models

In short, global problems need global solutions.

2.1 Increasing Realism

Recently [1] analysed 30 years of global satellite data in order to try to understand the degrees of deforestation imposed over the whole planet by the human population. The potential exists within the same dataset to look at the evolution of cities, the growth of road networks and similar phenomena. As this kind of data becomes more common, more realistic ABM, using real-world geographies and representing steadily larger scales will become possible, and the demand to be able to embed models in such realistic environments is likely to increase (see e.g [2]). For policy purposes this kind of realism is likely to be necessary for ABM to make useful strides into actual decision making. The inter-connectedness of current society implies that such realism will ultimately demand a global approach, at least for some issues.

2.2 De facto dynamics is already global

The effects of human populations on the earth now dominates much of the global system ([3], [4]), to the extent that multiple planetary boundaries (in terms of exceeding available capacity) seem to have been overstepped [5]. The pressing problems of our time, such as anthropogenic climate change, deforestation, over-fishing, habitat loss, over-extraction of resources, poverty and inequality, economic fragility, food security, warfare and geo-political manoeuvering are global in scope, and cannot be effectively addressed using models that only cover small fractions of the global system. Economies are globally coupled through international trade and banking, populations through air transport and migration, societies through electronic communication. Potential threats from global pandemics [6] or large-scale societal collapse [7] cannot be meaningfully examined using only local case studies.

2.3 Integration is needed with other existing global models

Current “Earth System Models” do not include humans. The typical such model is based largely on comprehensive physical models of the atmosphere and ocean, land surface and terrestrial vegetation [8][9]. Recently models that include animals have become available, but not closely coupled to the other physical models [10]. Associated calls to “model all life on earth” distinctly lacked a human component [11]. “Integrated Assessment” models for looking at climate change effects are typically classical equilibrium models that look only at one way impacts from climate change to economics: Feedbacks between humans and their environmental impacts are entirely absent [12]. This is a significant hindrance to effective policy making for global scale phenomena e.g.[13] as the full complexity of social systems is not represented.

2.4 More is Different

This phrase comes from a well-known paper in the physical sciences [14], but applies with at least as much force in the case of societal phenomena. Dynamical phenomena change qualitatively with change in scale. The possibilities represented by a group of people in a small room differ from those within a building, a building differs from a street, a street from a city, a city from a country and a country from a region. At each level we encounter different dynamics, and different types of structure, with coupling both to levels above and below. Attempting to model the behaviour of people in a room depends for example on their perception of the transport system (will they get home in time given the city scale traffic jams?), where they will go for lunch (what has global trade provided by way of food?) and the ability to speak freely (what is the local political situation?). The background context is thus dependent on the embedding of people not just within social networks, but at the same time with infrastructural and economic ones, in a way that is inextricably bound up with global phenomena.

2.5 Boundary conditions are problematic (especially for validation)

Most current ABM that attempt to make a more-or-less realistic connection to observed social situations (rather than abstract representations of mechanism, for example) focus on case-studies, often backed up by social surveys. In some cases this will be supplemented by census data or other larger-scale snapshots of the people under study. When trying to make dynamical representations of such cases, the situation is hampered not just by limited abilities to gather all information that might be relevant, but by the necessary lack of embedding in the surrounding social and spatial structures. When trying to model a farming community, for example, a key issue is usually not only the price of agricultural inputs, but also sale prices for outputs. Depending on the location and the type of crop, these are typically determined by much larger scales than that of the case study, including regional-scale climate variability and global scale economic markets for agricultural commodities, such as fuel for machinery, fertilizer for fields, and international demand for food. The possibilities for validating the dynamics within a small community are thus depend on assumptions about the situation beyond the model boundary. Future projections are likely to be invalidated by a lack of representation of out-of-boundary dynamical evolution. On the other hand, were there a global-scale dynamical model within which to embed the case study, the possibilities for more realistic presentation of dynamics would be enhanced.

2.6 Space and timescales are not independent

Consider the case of modelling a city the scale of London. To know about traffic patterns a few hours ahead, we probably only need to model the city itself, plus a region outside to allow for traffic movements and interactions that can propagate on that timescale. However, consider instead trying to project energy needs in the same city for the next 30 years, something that is relevant to setting policy now, as infrastructure tends to have long lifetimes: just considering the city and a small region outside its boundary will be woefully inadequate. London is a globally coupled city, and the changes over 30 years bound to be connected not just to the whole UK but to Europe and beyond. It is not uncommon, however, to see case study ABM extrapolated for this kind of timescale (or possibly even longer). While projecting more than a few years into the future may be uncertain, the likely inaccuracies will become severe if long timescales are coupled with small spatial scales.

2.7 Social justice goes global

While [5] emphasised the overstepping of physical and ecological boundaries, [15] pointed out that there are social justice boundaries that simultaneously need to be considered. Whilst some of these may be purely local, the social issues that arise as a result of climate change, international finance, ecosystem pressures and inequality need global social, political and governance (not just economic) models. Otherwise the current tendency to reduce all global issues to economic ones is likely to continue.

2.8 Models that are not materially closed are under-constrained

In attempting to understand whether current societies are sustainable, one of the key factors is to be able to trace the flows of material goods through international trade and their processing and re-processing by international industrial systems, on their way to the consumer. The implied inputs of energy, raw materials, human labour, and pre-existing capital infrastructure impose strong constraints both on the timescales that are needed and on the amount that can ultimately be consumed. Data on these flows is currently very sparse, and most complete for financial transactions reported at country scale (rather than actual material flows). Models of the global system would potentially be able to help in assessing how accurate and complete these accounts are in practice. Perhaps more crucially, in many current models, inputs needed for change over time are often assumed to exist without either time or physical cost being accounted for. This allows models to have more degrees of freedom than exist in reality, and leads to under-constrained models that are harder to validate. As a simple example, the typical Sakoda/Schelling model assumes that movement is both costless and instantaneous: in practice people face multiple constraints, not least the time and material cost of getting from place to place along with transport of all associated material goods. Once one begins to make more detailed simulations that include, for example, construction of new houses or tower blocks, the requirement to make such construction both materially and energetically closed leads to coupling to the international trade system, which again becomes a key factor in the possibilities for social change.

2.9 Global datasets provide many constraints

Pattern oriented modelling [16] seeks to use multiple different observational measures to constrain models, or exclude those that are insufficiently realistic for a given purpose. Making comparisons of models to measurement in only a few or even a single dimension (e.g. population size, or area of land-use change) can lead to under-constrained models that are hard to generalise. For [16] the point is that with multiple criteria available, even weak constraints (such as the sign of a change) when combined with others, can lead to strong constraints on models. Global satellite data, for example, is spatially extended, rich in detail, but generally coarse in resolution. Trade transaction data is sectorally detailed but often incomplete, whereas census data is typically detailed in some regions but not in others. With a global model where dynamics is fully coupled, multiple and partial weak or incomplete datasets could become powerful tools to exclude models with unrealistic mechanistic under-pinnings.

2.10 “Rumsfeldian” reasons (we don’t know what to leave out)

The degree to which local dynamics are independent or otherwise of larger-scale factors is currently unclear. It is possible that some small-scale phenomena are dynamically de-coupled from larger scale, but which these might be is not necessarily obvious a priori. For example, the factors that lead to global price spikes for fuel and food might trigger events such as the Arab Spring, but the extent to which this might be true is hard to evaluate for lack of sufficiently detailed global models. At the moment we have no real idea whether a global ABM would be better (or worse) than the current approach (i.e. guesswork).

2.11 Attempting to construct such models will lead to learning

Simply making the attempt to build models at global scale will be sufficiently demanding that, even if not successful in addressing the issues mentioned above, new concepts, techniques, datasets, analyses and computational methods will almost certainly be required. A project to create a plausible and meaningful global model (or better, multiple different global models created by different research teams) of all society should also help to point out our areas of ignorance and stimulate future pathways for appropriate-scale case studies. Furthermore, linking to other communities of global modellers should help to raise the profile of ABM and break the stifling stranglehold of equilibrium economics on the thinking of decision makers.

2.12 Objections

The typical reaction to the idea of such large scale models is "because <insert reason here> this <u>cannot</u> be done" often (but not always) followed by an implication of "and therefore *it should not be tried*"; The classic example is the infamous paper by Lee [17]. This is much like the reaction of some social scientists to the idea of computational modelling of society, and thus to ABM in general. I give here a few samples and some possible counter arguments.

a)You can't create a model of everything.

This, however, is not the objective. The idea is that anything less than global is *insufficient* to the task for certain issues of relevance to society. What is *necessary* (i.e. how much of "everything" do we need to model) then becomes a question for experimentation; to do the experiments we need to build the models.

b)We are all doomed so why bother?

The doomsday dialogue has become all too common, particularly when talking about climate change. Warnings need to be given, but the mantra needs to change from "everything is hopeless" to "can we change things?", to which the answer is hopefully, "yes, we can". However, blindly changing at random may do as much harm as good – we need guidance in complex systems, and models may be able to help. If there are things that cannot be changed, then models may show how to cope. Either way global issues will remain hard to study without appropriate scale models.

c)If we have to model every person, then why not every cell in every person? Or every molecule in every cell?

Again this is a question of sufficiency versus necessity. Model boundaries should be extended as far as needed, but no further. The standard assumption for ABM is that people can be treated as indivisible discrete entities, as we do in most of our social lives. It could be, though, that in order to model, say, global disease propagation, we find we need to include a dynamical immune system in each person – perhaps for influenza, where mutation of the virus within the organism matters – and that models that do not do this make important errors. However, if this process is largely decoupled from disease spread between people, then perhaps not. Again, finding this out requires testing, and we should model as much of the system as is required to answer the questions of interest. As a result of building a detailed global model we might also find phenomena where social structures decouple from the larger scale, or can be treated in isolation, but we need the large scale models to find this out. On the other hand some issues, for example those associated with gender, cannot be adequately treated with aggregate entities like households – one needs to represent individuals.

d)Social systems are too complicated

Again an objection sometimes levelled at ABM in general, but here the implication is that on going global the number of social phenomena and the range of scales is so large that our lack of understanding will prove overwhelming. If on the other hand we try to cope by using some kinds of aggregates (e.g. modelling cities or regions as agents) then we again have the problem that we don't know what the appropriate dynamics for such agents might be. The way to deal with this is, as with any model project, to start simple and then build in structure as needed. Here I suggest that "simple" implies modelling individual people, as this does not, intially, require assumptions about how to model aggregates – we hope, as usual, that these can be made to emerge, in later developments, from agent interactions. On the other hand [10] makes some progress with cohorts of agents, at least for the purposes of modelling necessities such as food, metabolism and breeding. The useful level of approximation depends on whether the model can match datasets that are of interest for model output.

e)The model will be too complicated

The underlying complicatedness of a model is not necessarily related to the scale of the system it represents. The global scale ecosystem model of [10] is mechanistically quite simple, but the emergent spatial patterns have structure all the way from the grid scale out to half of the globe, driven by patterns in rainfall or ocean temperature, as well as by local dynamics. How complicated a global model needs to be will depend upon the purpose for which it is to be used, just as with a smaller scale model.

f)The output will be too hard to analyse

Large models already exist that produce tera-bytes of output. As data sizes and complexities have increased, the methods needed for their analysis have been developed. Machine learning may help, both in terms of processing inputs and understanding outputs. Although a model may be highly dis-aggregated, its outputs need not be – sanity checks can be made on highly aggregate variables (e.g. global total biomass, global GDP), and then one can examine output in increasing levels of granularity.

g)Just building a big model has no purpose

In the first instance the purpose is technical exploration. Can global models be constructed at all ([18] demonstrates "yes, at least at some level")? How long will they take to run? What input data is needed? How can we structure the output? These technical questions are the first port of call, and the beginnings are tackled in the next section. Beyond that there are multiple research questions that can be built around a model structure that has a global scope – e.g. Can we model the relation between global trade and water consumption? How are global ecosystems impacted by human activity? What policies might usefully limit global CO_2 emissions? Each of these is likely to need model (or sub-model) specialisation for a particular purpose, but the global model system needs to exist as a background, and to provide code to progressively build upon. Where questions overlap (e.g. land-use change and climate change) then we have the global technical frame ready to build the dynamics into a single model (rather than creating an "integronster" out of separate models with different reference scales, different levels of aggregation and so on).

None of these points, it is argued here, are reasons for *not trying* to build global models. As mentioned above, at the very least the process will lead to learning: Without the attempt we have no demonstration of the ways in which such models might fail. The key will, as always, not to be tempted to use models for policy advice too early, before model capabilities have been sufficiently tested.

The computing power exists to make models at such scales. The GSAM of [18] included one agent for every member of the global population: Since that time, however, little seems to have been published building on or extending this approach, although [2] implies that USA-scale models are now in routine use for decision making. More recently the Episimdemics model [19] has been run on a large high performance computing (HPC) machine using 655,360 cores (!). The focus of these very large models remains with disease however. The next largest scale is currently that of traffic modelling (on which the disease models tend to be based), one example being "virtual Belgium" [20], incorporating 10,000,000 agents. However, in order to make such large models it is necessary to use parallel programming techniques.

3 A parallel primer

Unfortunately the issue of creating parallel programs is a rather complex one. This is perhaps one of the reasons why few ABM to date have taken up the challenge to work at scale. Coupled to this are the difficulties of using the available platforms with which most modellers are familiar; only a few really support making models that are efficient when working with large numbers of agents and/or large spatial domains. Partly this is because there are multiple issues to do with hardware, software libraries and algorithms that do not arise in the case of standard serial (i.e. non-parallel) code.

Where the aim is to run a single large program and gain an improvement in speed of a single model run, the idea is to split the code in such a way that parts of the single model run concurrently on separate threads. Since we are thinking of single model in which interactions between different model parts are likely to be important (e.g. communication between different agents), exchange of data between threads becomes crucially important, and the speed of the code depends strongly on how efficient this communication takes place. Parallel models of this type (as opposed to parameter space exploration., for example) for this reason rarely achieve anything close to the maximum possible speed-up for a given number of threads: in many cases the communication overhead can become so high that speed-up saturates beyond some rather small number of threads, or performance even decreases.

Approaches to parallelisation differ depending both on the nature of the application and the programming language and parallel framework to be used. For example, while Java runs efficiently using its own built-in threading system on a single machine, it tends to suffer from high latency in communication between machines, and historically this has led to poor performance relative to other languages on HPC. In GSAM Parker and Epstein managed to circumvent this problem, but at some cost in designing their system, and newer libraries such as AKKA have promise in making Java applications scale (for example used in conjunction with the Matsim traffic modelling platform - see e.g. the BEAM platform http://beam.lbl.gov/). The “extreme scale” systems that allow for very large scale models, as given in the review of ABM platforms by [21] are RepastHPC [22], PDES-MAS Matsim and Swarm (Objective - C). The first two of these are based on C++. This also applies to the CHARM++ system used for the Episimdemics model, which is clearly “extreme scale” although not mentioned by [21]. [23] also include FLAME, D-Mason and Pandora. However, for direct comparison they only present results from FLAME and RepastHPC, of which the latter comes out with better performance overall. D-Mason they feel is currently rather at an early stage of development, and therefore perhaps not ready for such comparative evaluation. The virtual Belgium model combines Matsim with RepastHPC, and thus would appear to depend on C++. Systems such as FLAME and CHARM++ concentrate on message passing between agents and treat the simulation largely as task parallel. RepastHPC and D-Mason by comparison are more set up to use domain-decomposition – a model grid (or network in RepastHPC) is used as a container for agents, and sections are cut out and handed to individual threads. Agents may move between the various parts of the subdivided domain as required. Given the apparent performance advantages, and the spatial flavour of the application, the following reports some experience in global modelling using C++ and RepastHPC.

4 Making MOTES

The starting point for the current model development was a C++ version [24] of [10]. The original code was written in C#, translated in order to run it on a linux-based HPC. In the process model run-time improved somewhat, since C++ has a more efficient memory management model than C#. The code was first parallelised using the OpenMP library. This allows one to use “loop-level” or “block-level” parallelism to speed particular sub-sections of the code. This consists of “decorating” selected loops or regions of code with short directives to indicate that they should be run over multiple concurrent threads. Good speed up can then often be achieved with minimal modification to the code, although the degree of speedup then also depends on the relative proportions of time spent on the parallel and serial sections (as given by Amdahl’s law), and complexities ensue if there needs to be any cross-thread exchange of data.

However, OpenMP is only useful on a single machine and thus limited in the ability to scale: In the current study this limited exploration to just 24 threads (although machines exist that support 256). RepastHPC uses the MPI library, which allows code to be distributed over arbitrarily large numbers of machines. Usually though, more code modification is required, and this proved to be so for the current case.

This model used RepastHPC 2.2.0. This comes in two flavours – a Relogo version (which tries to look somewhat like Netlogo) and plain RepastHPC without logo-like syntax. In either case some understanding of C++ templates is needed to make progress. Reasonably good documentation is supplied and a set of example models can be downloaded, built around a set of online tutorials. In both cases the RepastHPC library hides details of the MPI library, so that in principle the programmer does not need to be concerned about this detail. Model stepping is carried out using the Repast scheduler as for a standard Repast model. Multiple agent-types can be used, with a user-defined type identifier. Agents are place in a container (called a "context") and then can be subdivided automatically across threads using "projections", the latter being either a grid or a network (or both). Grids can be hard edged or toroidal (cf Netlogo), and agents can be located either at grid-cell centres (more efficient) or have arbitrary positions, or again, both. Since the domain is subdivided, when there can be interaction between agents in different grid cells, some cells within a neighbourhood may end up on different threads. RepastHPC allows for agents in a remote grid cell from a given thread to be copied onto that thread, and distinguished as remote agents, as opposed to thread-local. A user-defined parameter allows for the programmer to specify how many rows and columns of "buffer" zone are shared in this way. A core part of the library is the provision of synchronization; this provides for synchronizing data to remote copies, but also for movement of agents across threads in their entirety. Multiple synchronization methods can be specified, so that, for example, just a sub-set of the agent information is sent to remote copies, as needed. Value-layers are also provided for representation of more continuous fields such as temperature, for example. A limited level of I/O is provided, in ascii or NetCDF (a binary format best suited to gridded time series, and much used in the atmospheric modelling community).

While the above set of facilities is quite useful, in practice there were difficulties in setting up the global model. In the first place, a globe needs to be wrapped differently at the poles compared to wrapping in longitude. Because the agent-copying mechanism for representing those remote to a given thread assumes toroidal geometry, it is not immediately possible to set up a globe without the risk of agents at the South pole encountering those at the North (and a hard edged grid would exclude wrapping in longitude). In the current case this issue can be worked around by having a distance measure between agents that depends on latitude and using this to detect when agents are actually at polar extremes. However, a proper fix requires an additional grid-type to be added to the library so as to deal with spheres. Secondly, distances on a plane or toroidal grid do not vary in the same way as on a sphere, so agents needed to keep their own local co-ordinates so as to be able to use the correct spherical distance. Again this could be added to an updated RepastHPC core.

A further difficulty arose with erratic crashes of a Relogo model version, and subsequently with agents disappearing from the model on movement between threads. This proved to be a result of assumptions about the range of movement likely to be undertaken by an agent in a given timestep. RepastHPC uses MPI cartesian grids to subdivide the domain, and then assumes that agents will not move beyond the range of a Moore neighbourhood of this larger grid (where each MPI grid box contains a potentially large number of model grid cells). In practice, in the case of the current model, movements were capable of taking agents beyond the limits of the neighbourhood in this larger grid, with the result that they became lost from the model, or else gave rise to pointers aimed at memory that had been de-allocated. This required a code edit in the RepastHPC core to make sure that grid cell searches could find agents anywhere in the model grid, rather than just the local MPI-cartesian neighbourhood.

As a final issue, (apart from some small bugfixes to do with mapping ranges of co-ordinates to grid cells, and some edits needed enhance the I/O) it became necessary both to move to the standard RepastHPC version (rather than Relogo) and to introduce some explicit MPI calls in order to deal with algorithmic details in the implementation of the application. The model of [10] is an agent-based eco-system model based on functional types (in a similar way to the suggestion of [25] for human agents). Since the numbers of animals and numbers of species on the planet is much larger than can be represented as individuals, the agents are cohorts representing many individuals across multiple species (similar to the super-individuals described in [26]), but the number of cohorts can easily be in the hundreds of millions. Predator cohorts eat based on the size distribution of prey: this is the expensive part of the algorithm, as it scales as the square of the number of cohorts within the range of any given predator. Nominally these cohorts include "wild" humans that behave exactly as other animals in their equivalent functional group. Migration of cohorts between cells is relatively less expensive, but once predation is parallelised, a significant extra gain in performance is obtained by also ensuring migration takes place in parallel. Figure 1 shows the kind of scaling that can be obtained from these various parallel methods on two different types of hardware.

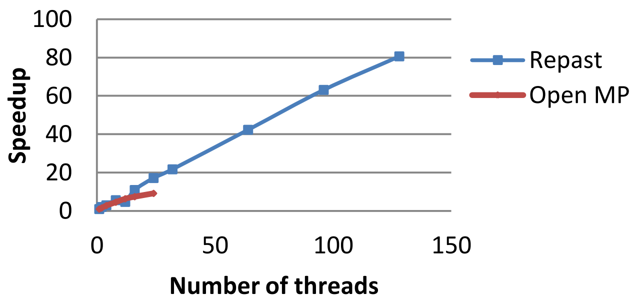

Fig.1Red line: Open MP version of the model running on a single workstation with up to 24 threads available. For comparison repastHPC was run on the Cambridge CSD3 Intel skylake cluster. For small numbers of core the two are somewhat comparable, but RepastHPC clearly scales better above 10 threads and continues to scale well out to 128 threads and beyond.

5 Conclusion

[21] indicate that using extreme-scale libraries for modelling with agents is difficult. Experience with RepastHPC certainly bears this out if the requirement is to implement a model essentially from scratch. Detailed knowledge of C++ and of MPI is required. RepastHPC comes with demonstration models, but no tests, and the reasons for model failures can be hard to diagnose. The way in which cross-cell synchronization works is not transparent, particularly if you have never encountered the Boost serialization library before. Understanding how to reliably adapt a working serial algorithm to this particular parallel environment can also be non-trivial. Any developer must either have a high level of programming skill, or know someone who does. Part

of the problem here may be starting from the wrong place: It may be that newer languages designed with parallelism in mind, such as Julia, Rust or Chapel, are the way to go in the future. Even so these are not easy for the naive programmer in the way that Netlogo can be. On the other hand more sophisticated ways of representing agents, such as discrete event systems, functional programming or declarative methods are also typically hard for the beginner to grasp: each of these may also have different implications for scaling up onto large machines.

On the other hand, the scaling offered by RepastHPC is excellent, showing no sign of saturating beyond 320 cores (the current tested limit), and promising to continue well beyond. Having created a working model and tested the fundamentals of its operation, further model additions now become straightforward: the current model has all the parallel machinery operational. The requirement for extensions to represent sophisticated human behaviour is one of defining agent rules and behaviours in a fairly standard way, using standard syntax accessible by those with reasonably elementary programming skills in C++ or any of its close cousins such as Java or C#.

The current proliferation of toolkits and platforms for creation of ABM is perhaps not helping the development of the field. I suggest that what is actually required to make significant advances is focussed models that are sufficiently complex to represent a wide range of behaviours and circumstances, are realistic in their representation of the world (rather than highly abstract) and to which programmers can add modules of their own whilst taking advantage of previous programmers' effort in developing well tested and robust code. This is the situation for the current generation of earth system models: writing such a model from scratch (in excess, typically of 500000 lines of code, albeit still typically in FORTRAN) for every new application would not be sensible. Traffic models form one such kind of development, with open-source examples such as TRANSIMS, Matsim and SUMO allowing newcomers to the field to access sophisticated models to which they can contribute. One possible application to encourage such model development in the current social and political situation might be to concentrate some efforts on global ABM for dealing with the social issues of disease, climate change and environmental destruction: this would have automatic policy relevance, whilst presenting many challenges to the field of social systems modelling. Such models must by definition, however, be global in scale and scope.

References

1. Hansen, M. C. P.,Potapov,P.V., Moore,R., Hancher,M. Turubanova,S.A., Tyukavina,A. et al, High-Resolution Global Maps of 21st-Century Forest Cover Change, *Science* 342, pp850-853, DOI: 10.1126/science.1244693 (2013).
2. Waldrop,M.M., Science 360,pp 144-147 DOI: 10.1126/science.360.6385.144(2018).
3. Vitousek, P.M., Mooney, H.A. ,Lubchenco , J. and Melillo, J.M. "Human domination of earth's ecosystems", Science, 25, pp 494-499 (1997).
4. Crutzen, P.J. and Stoemer, E.F., The Anthropocene, IGBP newsletter 41, pp 17-18, (2000)
5. Rockstrom, J., Steffen, W., Noone, K., Persson, A. Chapin, F.S., Lambin, et al, A safe operating space for humanity" Nature 461, pp 472-475 (2008),
6. Epstein, J.M. Modelling to contain pandemics. *Nature,* 460, pp 687. (2009).

7. Homer-Dixon, T., Walker, B. Biggs, R., Crepin, A-S, Folke, C., Lambin et al, Synchronous Failure: the emerging architecture of global crisis Ecology and society, 29, 6(2015)
8. Clausen, M., Ganopolski, A., Schellnhuber, J. and Cramer, W. "Earth Systems models of intermediate complexity"IGBP newsletter 41,pp 4-6 (2000).
9. Heavens, N.G., Ward, D.S, and Mahowald, N.M Studying and projecting climate change with earth system models Nature Education Knowledge, 4, 4 (2013)
10. Harfoot, M.B., Newbold, T., Tittensor, D.P., Emmott, S., Hutton, J., Lyutsarev, et al, Emergent global patterns of ecosystem structure and function from a mechanistic general ecosystem model. PLoS biology, 12(4), p.e1001841. (2014).
11. Purves, D., Scharlemann, J. P., Harfoot, M., Newbold, T., Tittensor, D. P., Hutton, J., & Emmott, S. Ecosystems: time to model all life on Earth. Nature, 493, pp295-297 (2013).
12. Farmer, D.J. Hepburn, C. Mealy P. and Teytelboym, A. A third wave in the economics of climate change, Environment and resource economics, 62, pp 329-357. (2015).
13. Lewis,K.H. and Lenton, T.M., Knowledge problems in climate change and security research. WIREs Clim Change 6:383–399. doi: 10.1002/wcc.346 (2016)
14. Anderson, P.W. More is Different. Science, 177, pp 393-397, (1972)
15. Raworth, K. Doughnut Economics:Seven ways to think like a 21st century economist. Random House Business Books, UK 372pp. .(2017)
16. Grimm, V. and Railsback, S.F. Pattern-oriented modelling: a 'multi-scope' for predictive systems ecology. Phil Trans. R. Soc. B, 367, 298-310. doi:10.1098/rstb.2011.0180 (2012).
17. Lee Jr., D.B. Requiem for Large-Scale Models. Journal of the American Institute of Planners, 39, 163-178. (1973).
18. Parker, J. and Epstein, J.M., A distributed platform for global-scale agent-based models of disease transmission. ACM Trans. Model. Comput. Simul. 22., 2:1-2:25 doi: 10.1145/2043635.2043637 (2011)1
19. Bhatele,A., Yeom. J-S., Jain,. N., Kuhlman, C.J., Livnat, Y., Bisset, K.R., et al, Massively parallel simulations of spread of infectious diseases over realistic social networks, 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. (2017).
20. Barthelemy, J. and Toint, P. A Stochastic and Flexible Activity Based Model for Large Population. Application to Belgium. Journal of Artificial Societies and Social Simulation 18 (3) 15 <http://jasss.soc.surrey.ac.uk/18/3/15.html>. doi: 10.18564/jasss.2819 (2015).
21. Abar, S., Theodoropoulos, G.K., Lemarinier, P. and O'Hare, G.M.P. Agent based modelling and simulation tools: A review of state-of-art software. Comp. Sci. Rev., 24, 13-33. (2017).
22. Collier, N. and North, M., Parallel agent-based simulation with repast for high performance computing Simulation 89 pp1215-1235 (2012)
23. Rousset, A., Herrmann, B., Lang, c., and Philippe, L. A survey on parallel and dsitrbuted multi-agent systems for high performance computing simluations. Com[uter Science review, 22, 27-46. (2016).
24. Bithell,M. Underwood,P. J., Harfoot,M.B. andTittensor, D. Testing robustness in a global ecosystem model:the effects of language translation http://www.geocomputation.org/2017/papers/8.pdf (2017)
25. Rounsevell, M.D.A., Robinson, D.T. and Murray-Rust, D. From actors to agents in socio-ecological systems models Phil. trans roy. soc. B, 367, 259-269 (2011) "
26. Parry, H. and Bithell, M., Large scale agent-based modelling: a review and guidelines for model scaling, in Heppenstall, A., Crooks, A.T., See, L.M. and Batty, M. (eds.) Agent-Based Models of Geographical Systems, Springer Verlag. (2011).