We show how to automatically **transform any existing population-based simulator** into a **probabilistic program**, **without re-writing the simulator**, enabling the simulator **to be analysed for interpretable inference.**

Hijacking Malaria Simulators with Probabilistic Programming

Bradley J. Gram-Hansen[1,2,*] Christian Schroeder de Witt[1*]
Tom Rainforth[2] Phil H.S. Torr[1] Yee Whye Teh[2]
Atılım Güneş Baydin[1]

[1]Department of Engineering Science, [2]Department of Statistics, University of Oxford

* Equal Contribution

Background

- Simulators arise in a number of industrial and scientific domains, encoding sophisticated generative models.
- Probabilistic programming provides a way to perform statistical inference over simulations of events in a programmatic way.
- Thus, by design, simulators are ideal programs for probabilistic programming.
- However, within existing probabilistic programming systems (PPSs) one would have to re-implement the simulator via the PPS language specification, which is inefficient and often not feasible due to the complexity of such scientific and industrial simulators.
- Recent work by Baydin et al. demonstrated a pathway to turn a particular type of event-based simulator into a probabilistic program, without having to re-implement the simulator in the existing probabilistic programming systems (PPS).
- But, this still meant that a large class of critically important population-based simulators could not be turned into probabilistic programs and as such could not be used within a probabilistic programming framework.

What we do

- In this work, we extend that framework to encompass population-based simulators, a very large class of simulators that are used extensively across epidemiology, multi-agent and financial modeling.
- We demonstrate how we can extract interpretable outcomes from that, which can then be used by decision makers in the fight against Malaria.

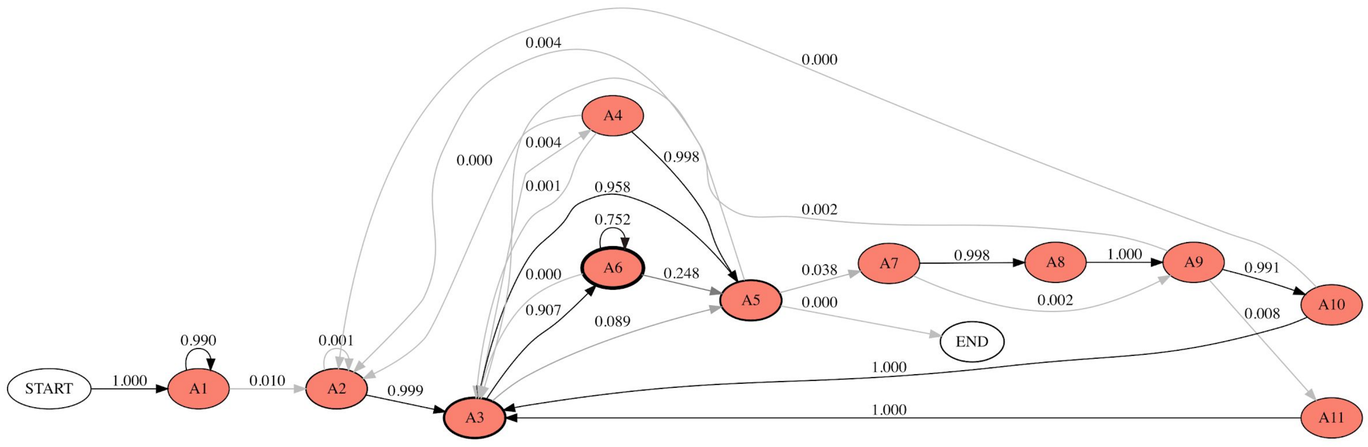
Interpretability: What does this mean?

Address ID	Full address
A1	[forward()+0x204; OM::Simulator:: start(scnXml::Monitoring const)+0x28a; OM::Population::createInitialHumans()+0x94; OM::Population::newHuman(OM::SimTime)+0x5c; OM::Host::Human::Human(OM::SimTime)+0x12b; OM::WithinHost::WHInterface:: createWithinHostModel(double)+0x99; OM::WithinHost::DescriptiveWithinHostModel ::DescriptiveWithinHostModel(double)+0x3a; OM::WithinHost::WHFalciparum:: WHFalciparum(double)+0xe6; OM::util::random:: gauss(double, double)+0xb4]__Normal
⋮	⋮

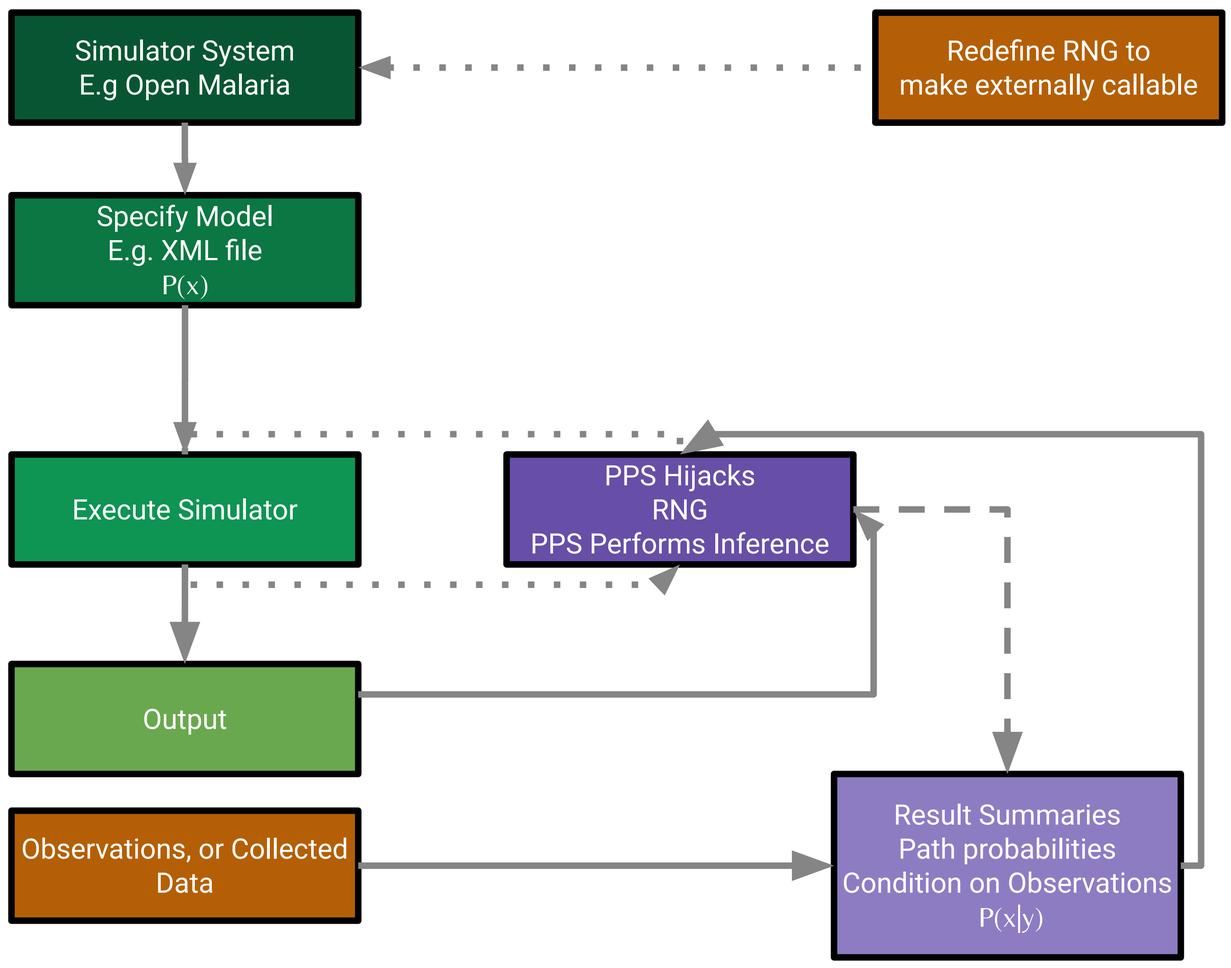
An example of an address generated for the model run in the OpenMalaria simulator. We can see that A1 relates to Generating a member of the human population who may or may not be infect with the Malaria disease. We get something similar for EMOD, except this relates to A7 in the EMOD program execution.

Address ID	Interpretation
A1	Generate a human in the population within host dynamics
A2	Generate another human in the population within host dynamics
A3	The population is updated and a new human, or humans, may get infected
A4, A5	Potential child deaths within the population are simulated
A6	Determines parasite density of an individual infection
A7	Models how the disease is progressing within the infected humans
A8	Models how the disease is progressing within the population
A9	Models how the disease is progressing within the infected humans after the population has been updated
A10	Full clinical update on the population for those without severe or no Malaria infection.
A11	Full clinical update on the population for those with severe Malaria infections

An interpretation table for each of the address of the overall trace generated from the corresponding forward run of OpenMalaria model.



How does it work?



References

Baydin, Atilim Gunes and Heinrich, Lukas and Bhimji, Wahid and Gram-Hansen, Bradley and Louppe, Gilles and Shao, Lei and Cranmer, Kyle and Wood, Frank and others, Efficient Probabilistic Inference in the Quest for Physics Beyond the Standard Model, arXiv preprint arXiv:1807.07706, 2018.

Acknowledgements

We thank Ewan Cameron of MAP at the Big Data Institute and the OpenMalaria team at the Swiss Tropical Health Institute for their time and help. BGH is supported by the EPRSC Autonomous Intelligent Machines and Systems grant (EP/L015897/1). CSW is supported by the project Free the Drones (FreeD) under the Innovation Fund Denmark and Microsoft. YWT's and TR's research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) ERC grant agreement no. 617071. AGB and PH are supported by EPSRC/MURI grant EP/N019474/1 and AGB is also supported by Lawrence Berkeley National Lab.

Take a picture to **download** the **full paper**