Tutorial: The OpenMOLE platform for model exploration and validation

Juste Raimbault[1,2,3,4,*], Paul Chapron[1], Maxime Colomb[5,1], Julien Perret[1], Romain Reuillon[3,4], Mathieu Leclaire[3,4]

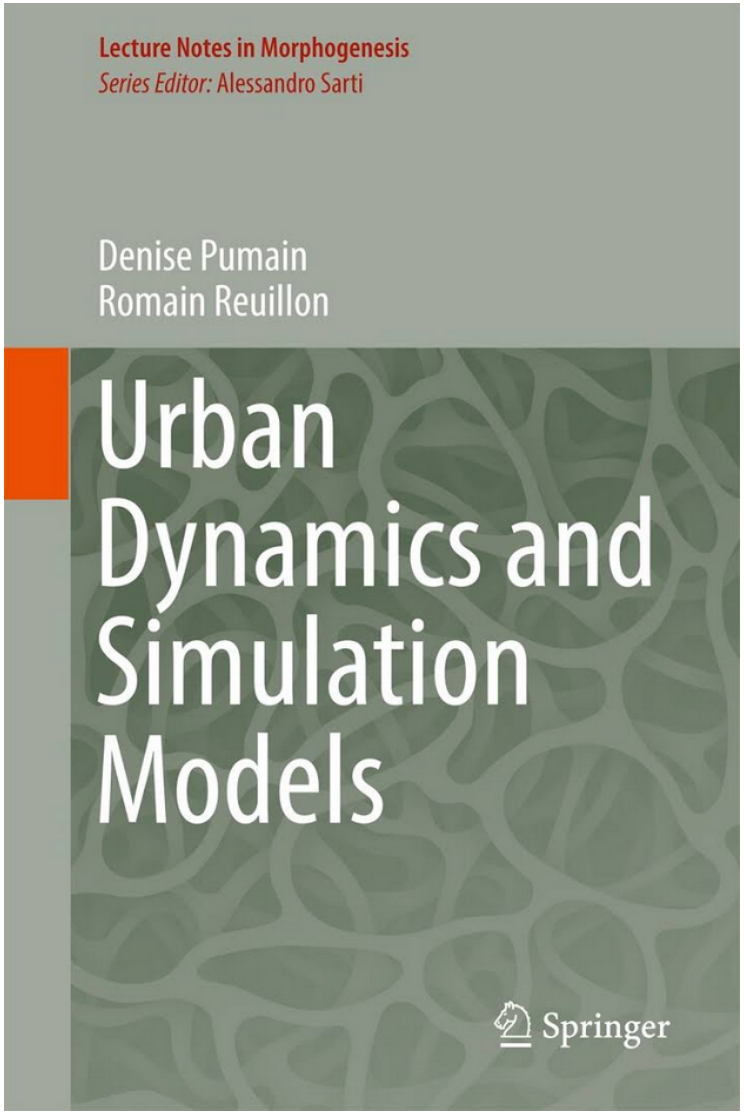
* juste.raimbault@ign.fr

[1] LASTIG, IGN-ENSG
[2]CASA, UCL
[3]UPS CNRS 3611 ISC-PIF
[4]UMR CNRS 8504 Géographie-cités
[5]ASCII, INRIA

Journée de la Recherche UGE-IGN-ENSG 2024

28/03/2024

ERC Geodivercity project: validation of simulation models



Evolutionary urban theory [Pumain, 2018]

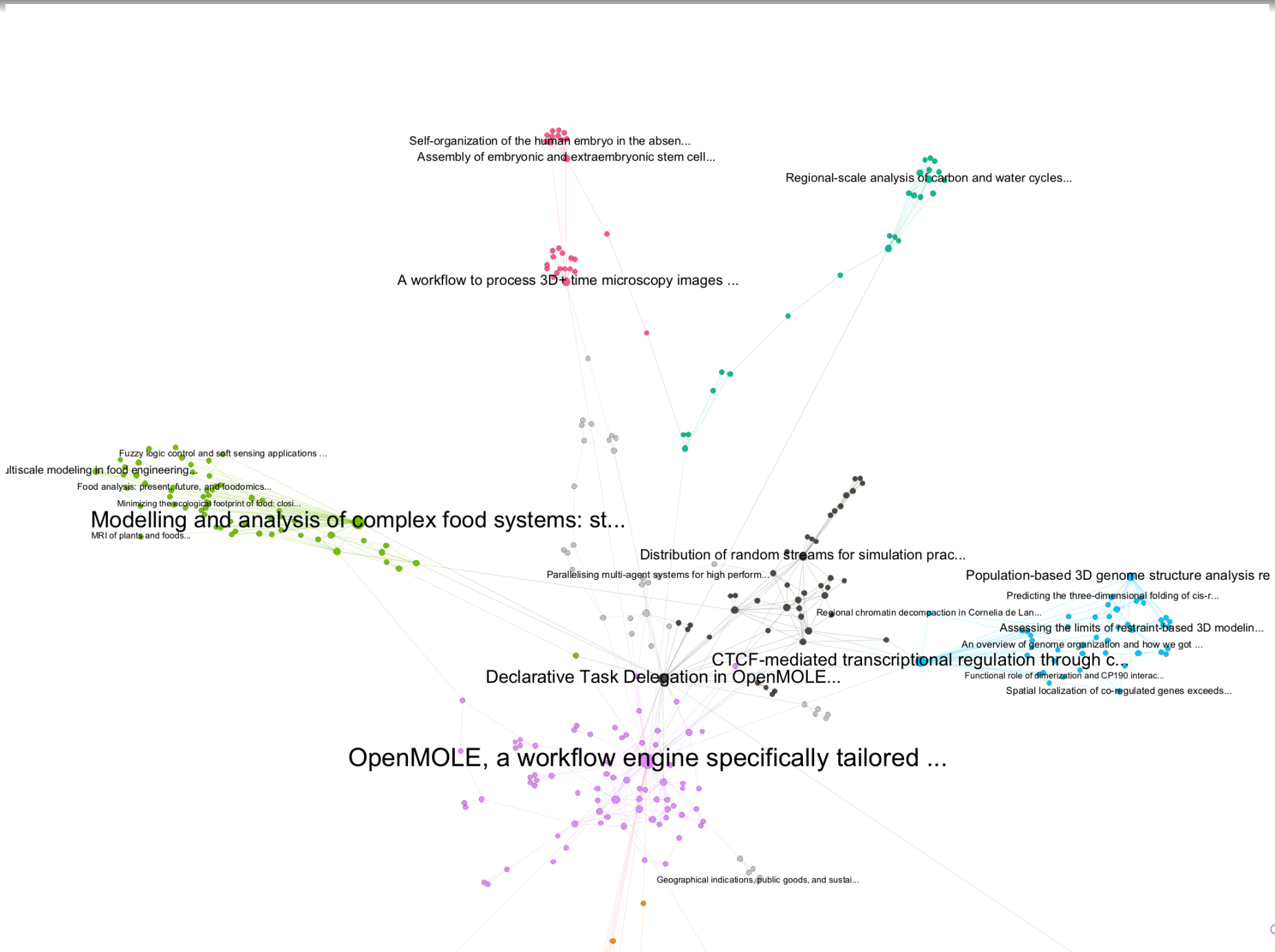
→ Stylised facts on main systems of cities worldwide

→ Simulation models with an explicative function

→ Tools and model exploration methods: OpenMOLE mainly developed by R. Reuillon and M. Leclaire since 2008 at l'ISC-PIF

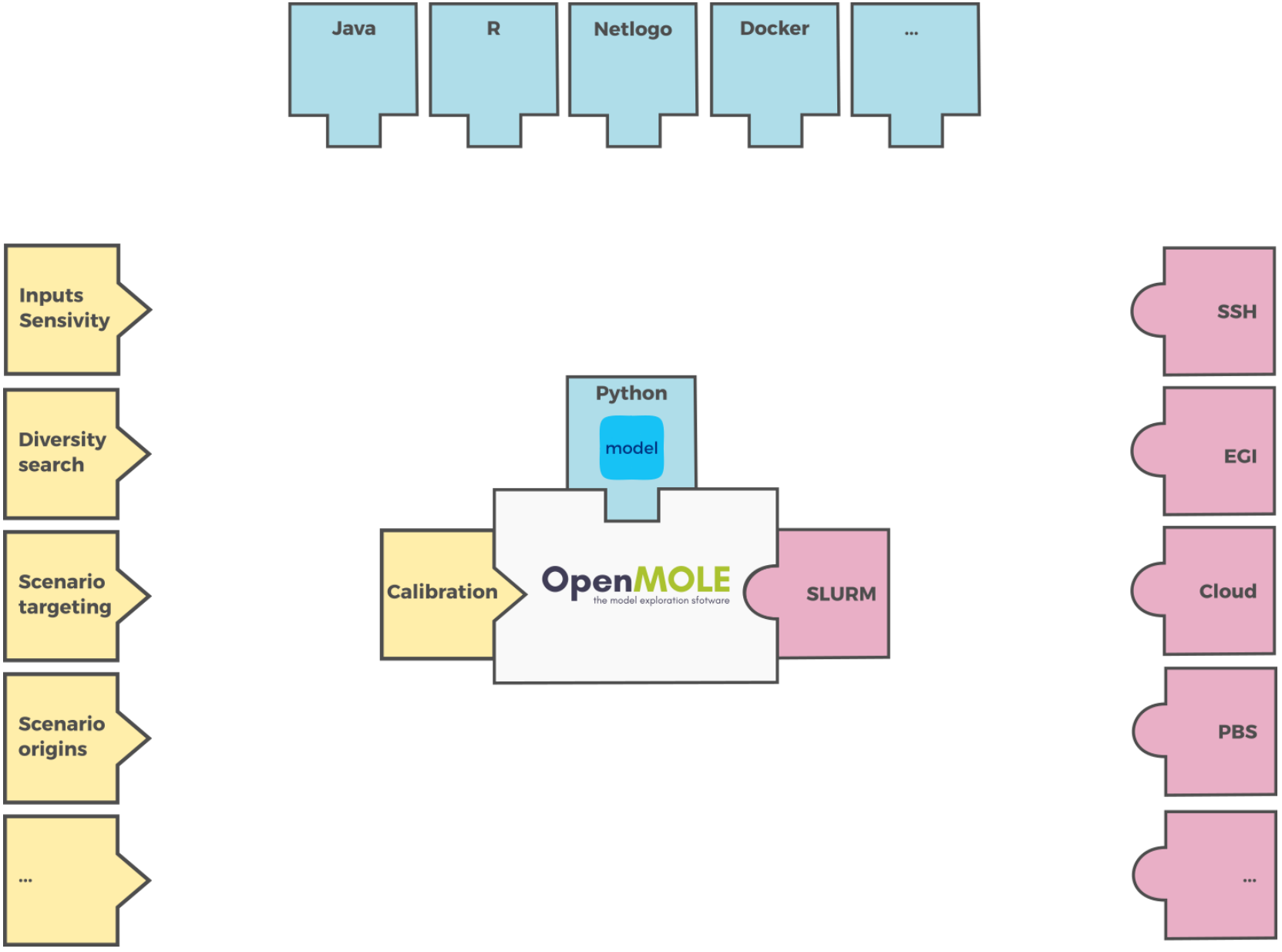
OpenMOLE

Scientific environment [Raimbault, 2018]

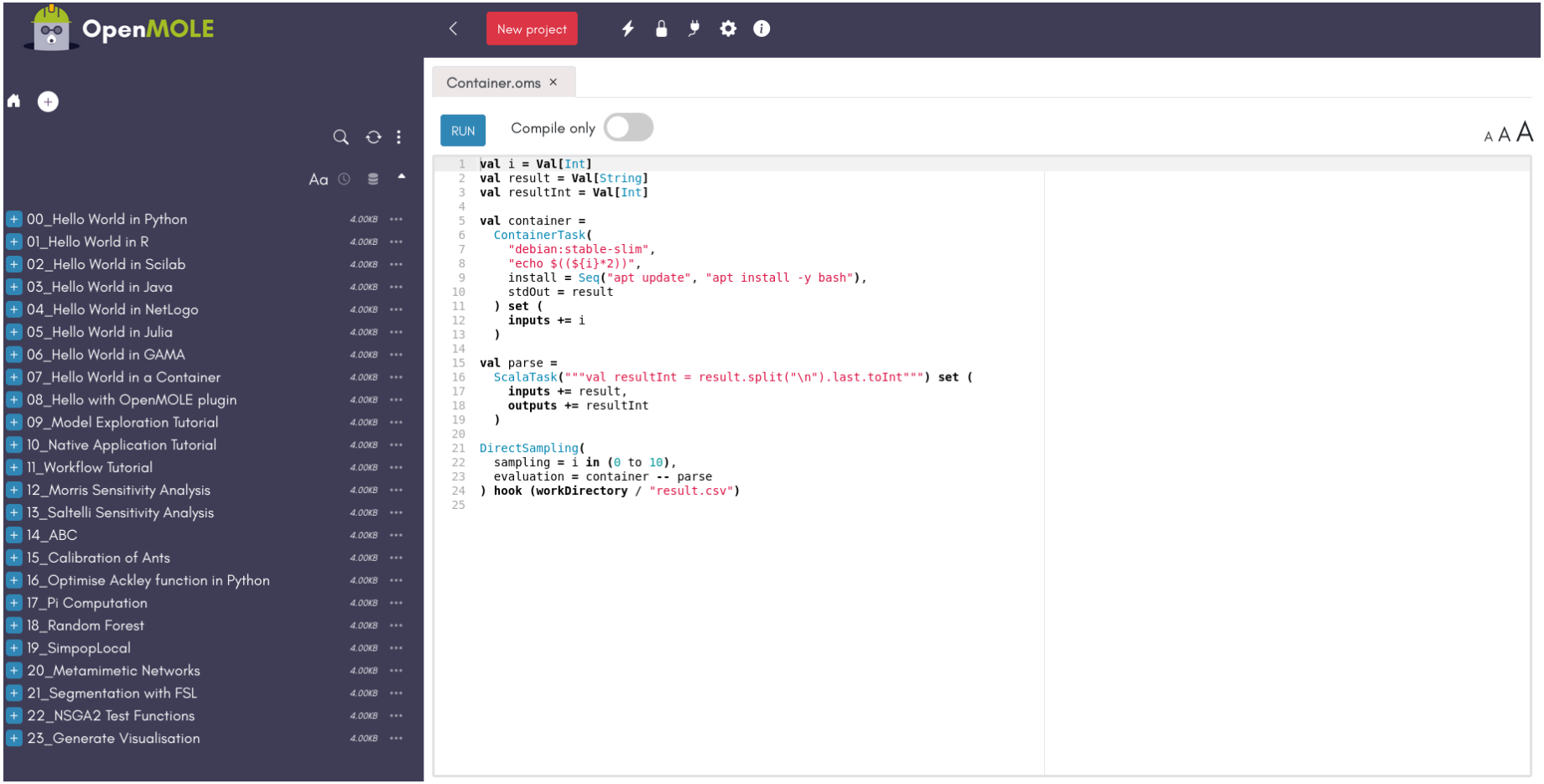


OpenMOLE principles

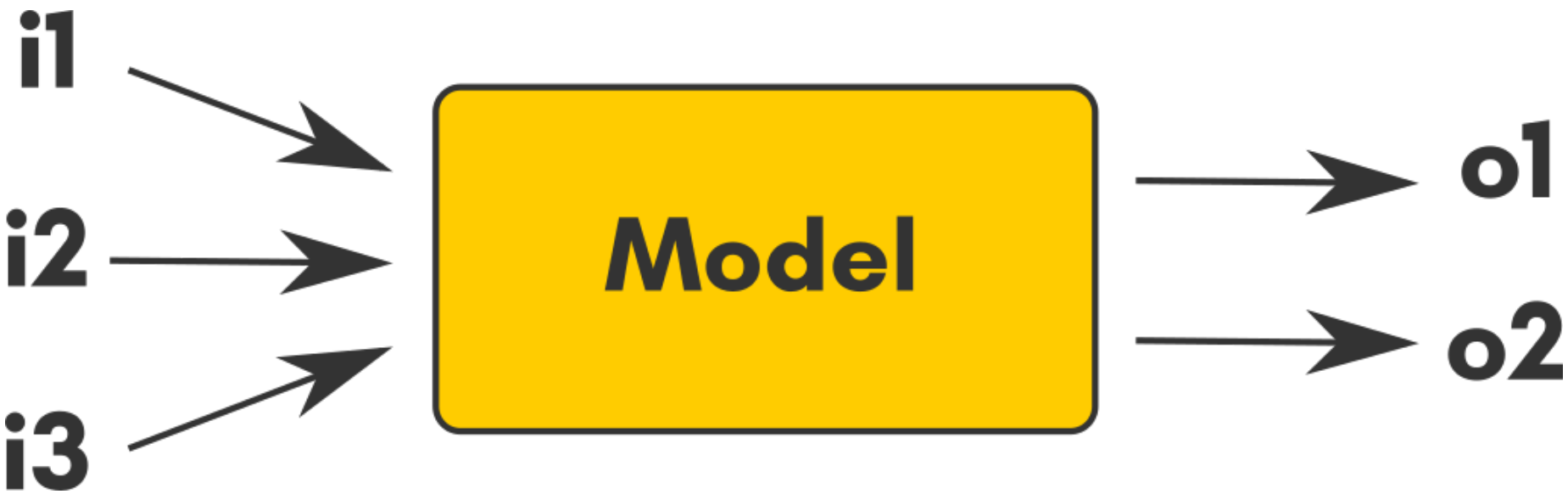
(i) State-of-the-art exploration and validation methods; (ii) Scaling with High Performance Computing; (iii) Model embedding.



Web application interface



Simulation models



Language agnosticity

C
R
C++
Java
Scala
Scilab
Octave
Python
Netlogo
...

Example: NetLogo model

```
val model =
  NetLogo6Task(
    workDirectory / "Fire.nlogo",
    List("setup", "while [any? turtles] [go]")) set (
    inputs += seed,
    outputs += (seed, density),
    inputs += density mapped "density",
    outputs += burned mapped "burned-trees"
  )
```

DSL based on scala for scripts

R code

```
val i = Val[Int]

val rTask =
  RTask("""
    source("function.R")
    function(i)""") set (
    resources += workDirectory / "function.R",
    inputs += i
  )
```

Similar syntax for the `PythonTask`

Docker container

```
val i = Val[Int]
val result = Val[String]
val resultInt = Val[Int]

val container =
  ContainerTask("debian:stable-slim", "echo $((${i}*2))",
    install = Seq("apt update", "apt install -y bash"),
    stdOut = result
  ) set (
    inputs += i
  )

val parse =
  ScalaTask("""val resultInt = result.split("\n").last.toInt""") set (
    inputs += result,
    outputs += resultInt
  )

DirectSampling(
  sampling = i in (0 to 10),
  evaluation = container -- parse
) hook (workDirectory / "result.csv")
```

Methods

- Parameter estimation
- Sensitivity analysis
- Robustness analysis
- Optimisation

Designed to be scalable, to take stochasticity into account, to be usable on any model and computing environment.

Methods

```
DirectSampling(
  evaluation = myModel,
  sampling =
    LHS(
      500,
      diffusion in (10.0, 100.0),
      evaporation in (10.0, 100.0)
    )
)
```

Example of method syntax: explicit sampling

Computing environment: up-scaling

Local prototypes, transparent scaling.

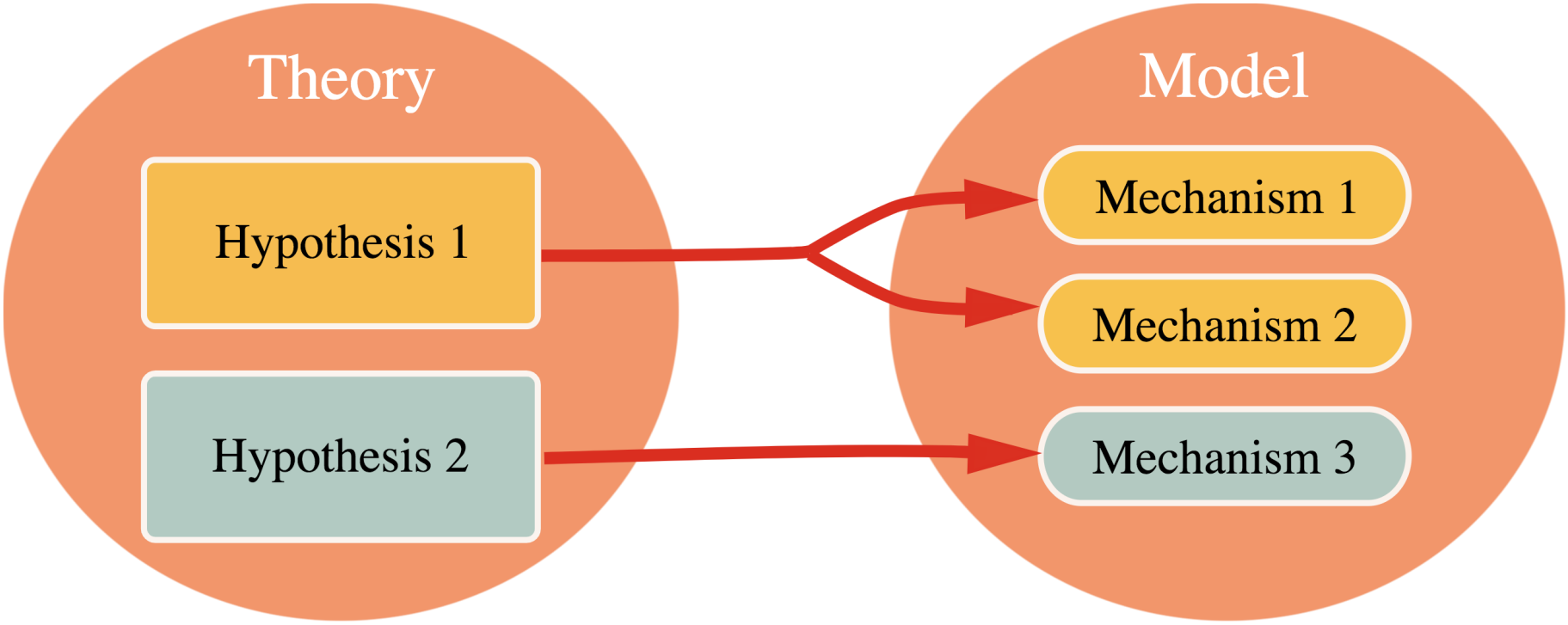
```
val cluster = SLURMEnvironment("login", "cluster.domain.org")

DirectSampling(
  evaluation = myModel on cluster,
  sampling =
    LHS(
      500,
      diffusion in (10.0, 100.0),
      evaporation in (10.0, 100.0)
    )
)
```

Computing environment

- Multi-thread
- Delegation through SSH
- PBS
- SLURM
- Condor
- SGE
- OAR
- EGI Grid

Theories and model evalution



Construct and evaluate a theory implying causal mechanisms.

Evaluation: How to ensure

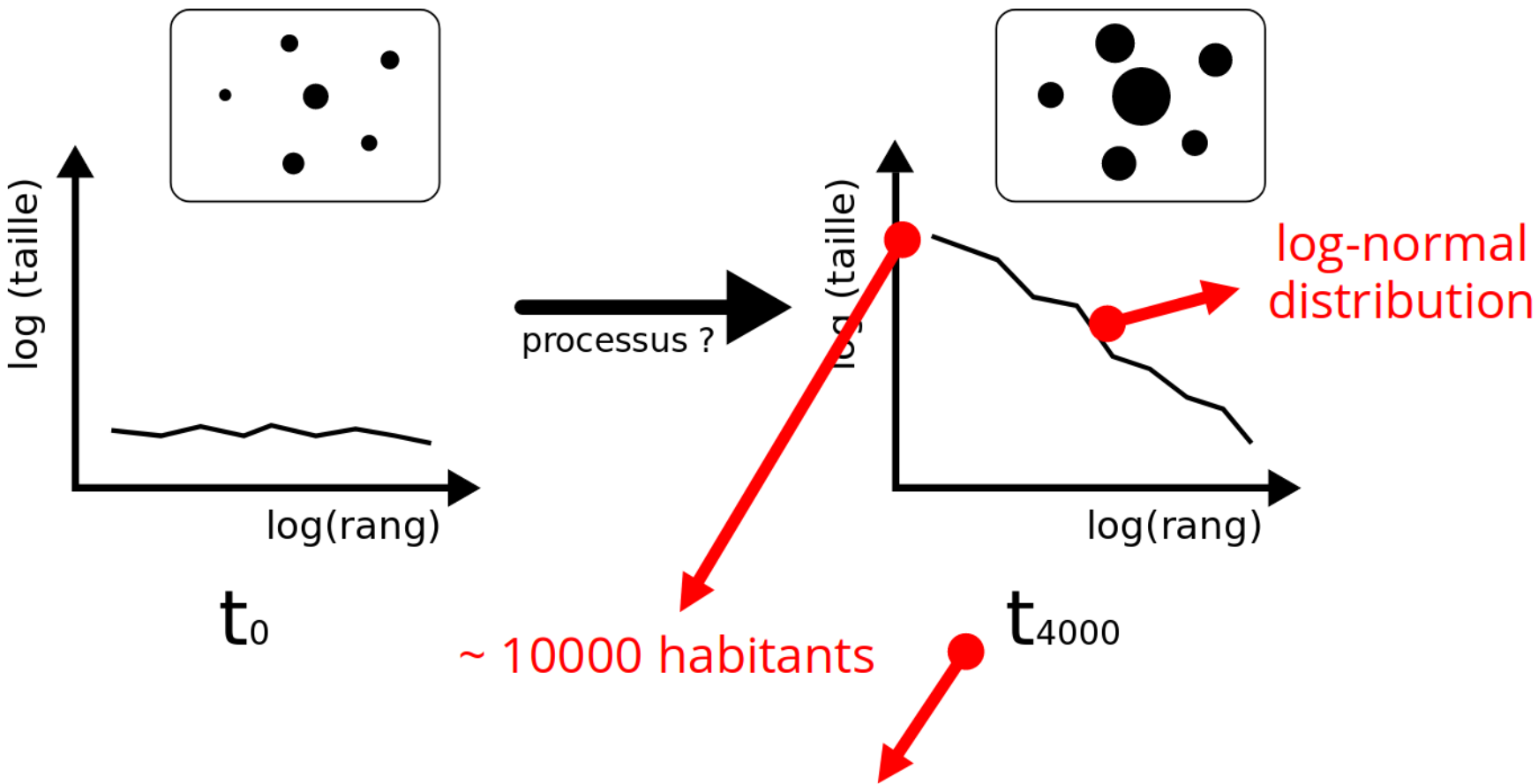
1. the sufficiency of the mechanisms ?
2. the necessity of the mechanisms ?
3. the uniqueness of the mechanisms ?

Sufficiency [Schmitt et al., 2015]

Classical approach: parameter space sampling (ex. Sobol) $\rightarrow$ large dataset produced and parameter space remains unexplored.

Inverse approach: from outputs to parameters

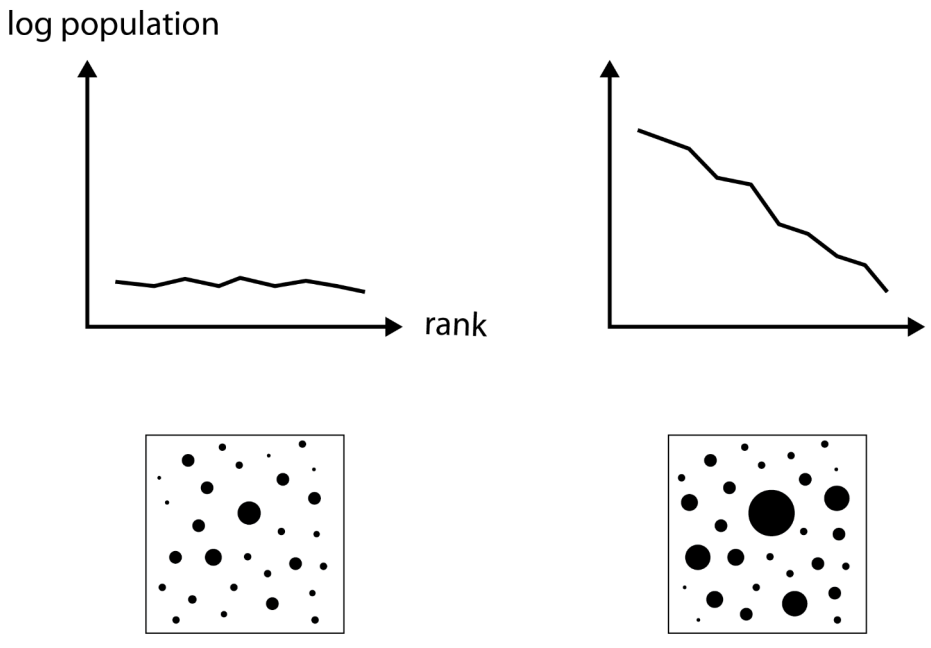
Formalising the expectations:



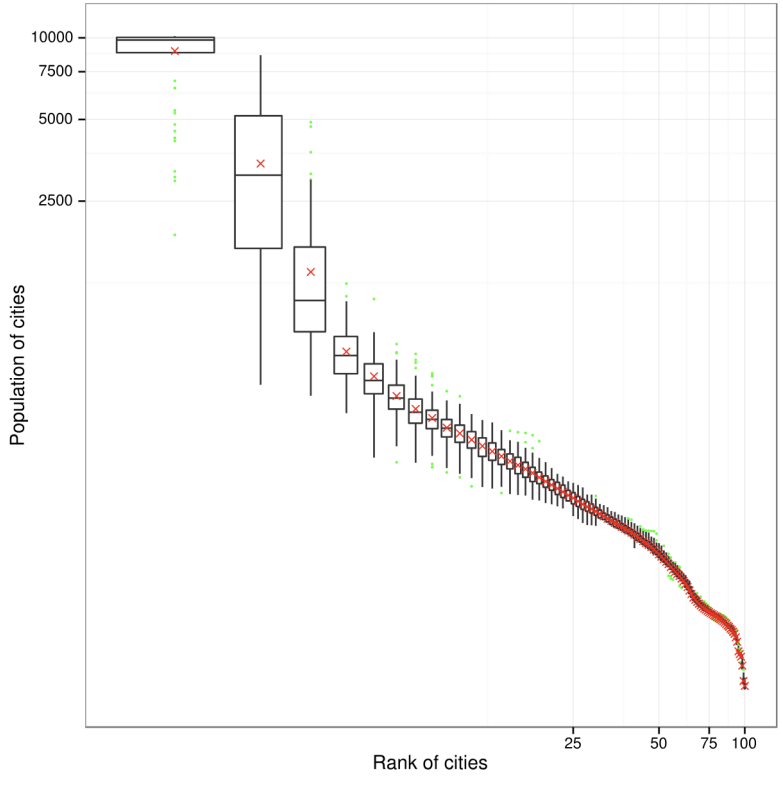
Calibration results

No compromise between the 3 objectives.

Searched pattern



Produced pattern



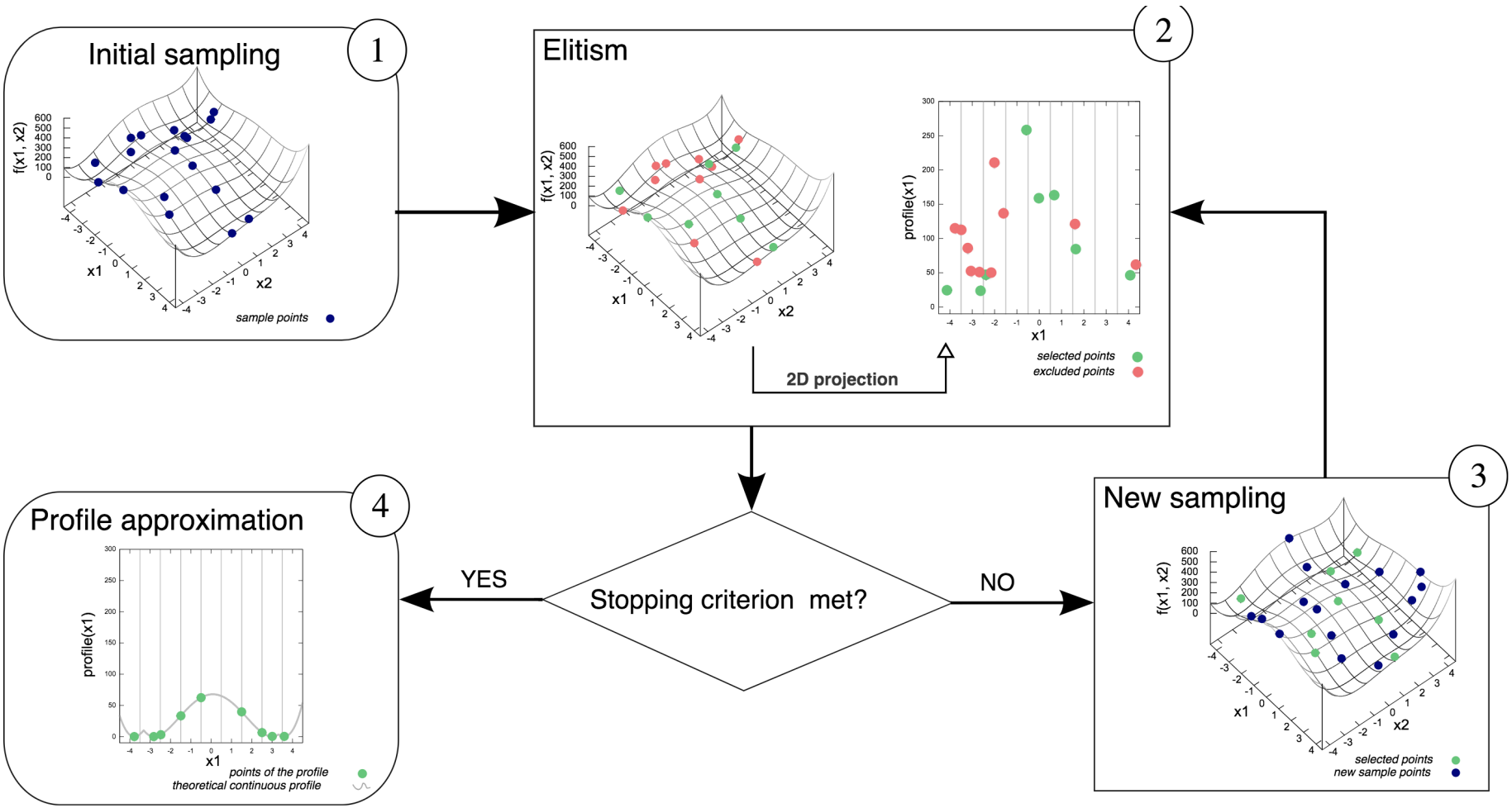
The method is tractable (even for ABMs): Handles stochasticity: 100x gain; Support for distributed computing: 1000x gain.

Necessity [Reuillon et al., 2015]

A new algorithm

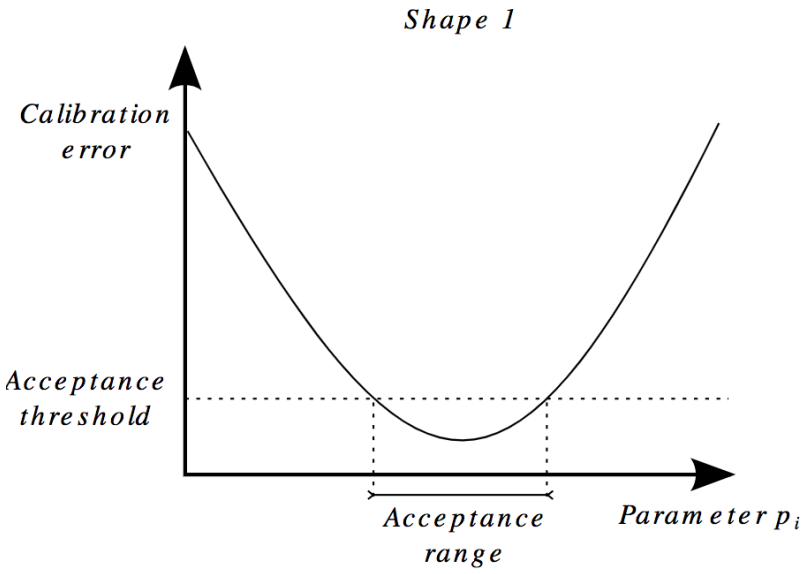
1. To detect if a parameter is useful: it impacts the capacity of the model to produce plausible outcomes.
2. To better constrain the parameter ranges.
3. As an indirect way to detect if some of the mechanisms are expandable

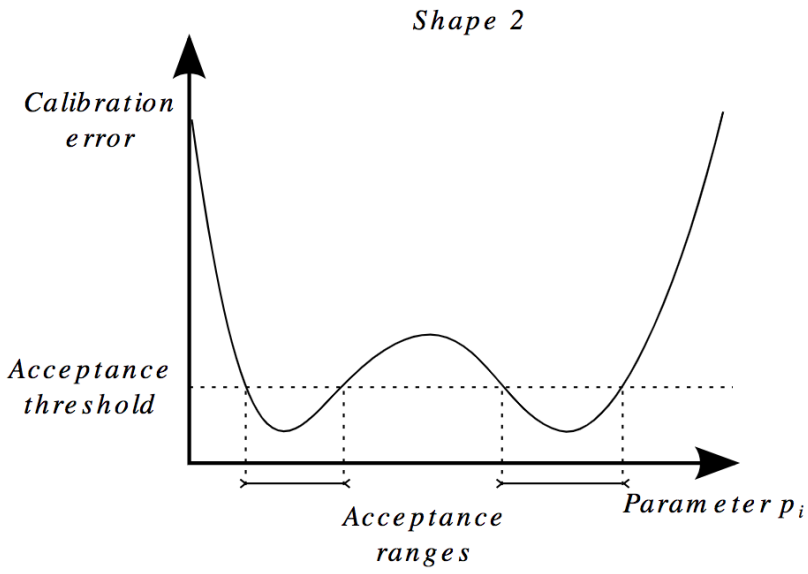
Profile algorithm

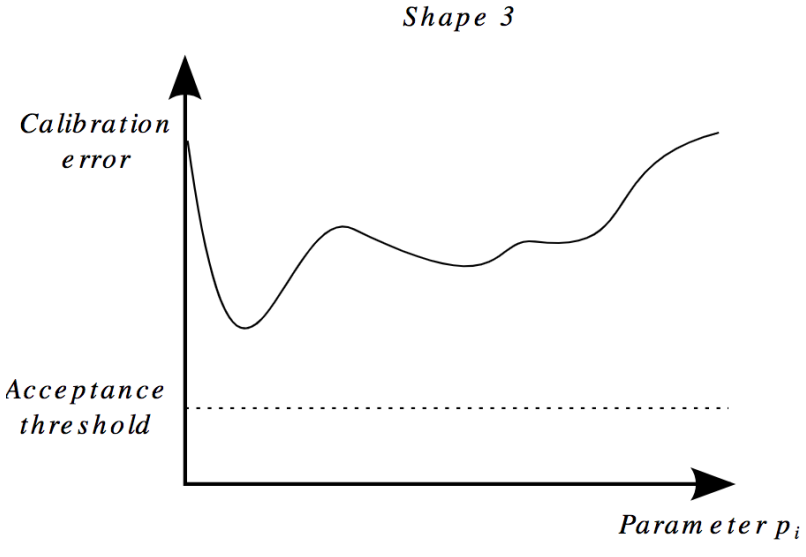


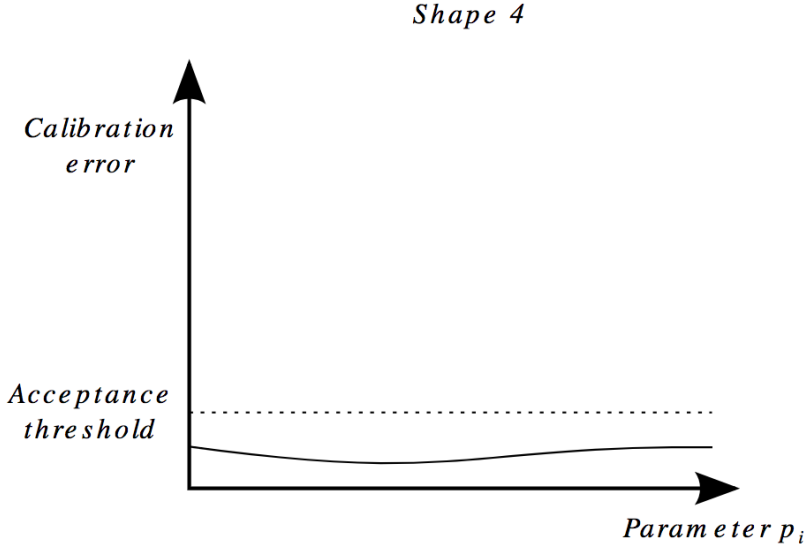
Compute the best of calibration for hundreds of values along the definition domain of a parameter.

Profile algorithm

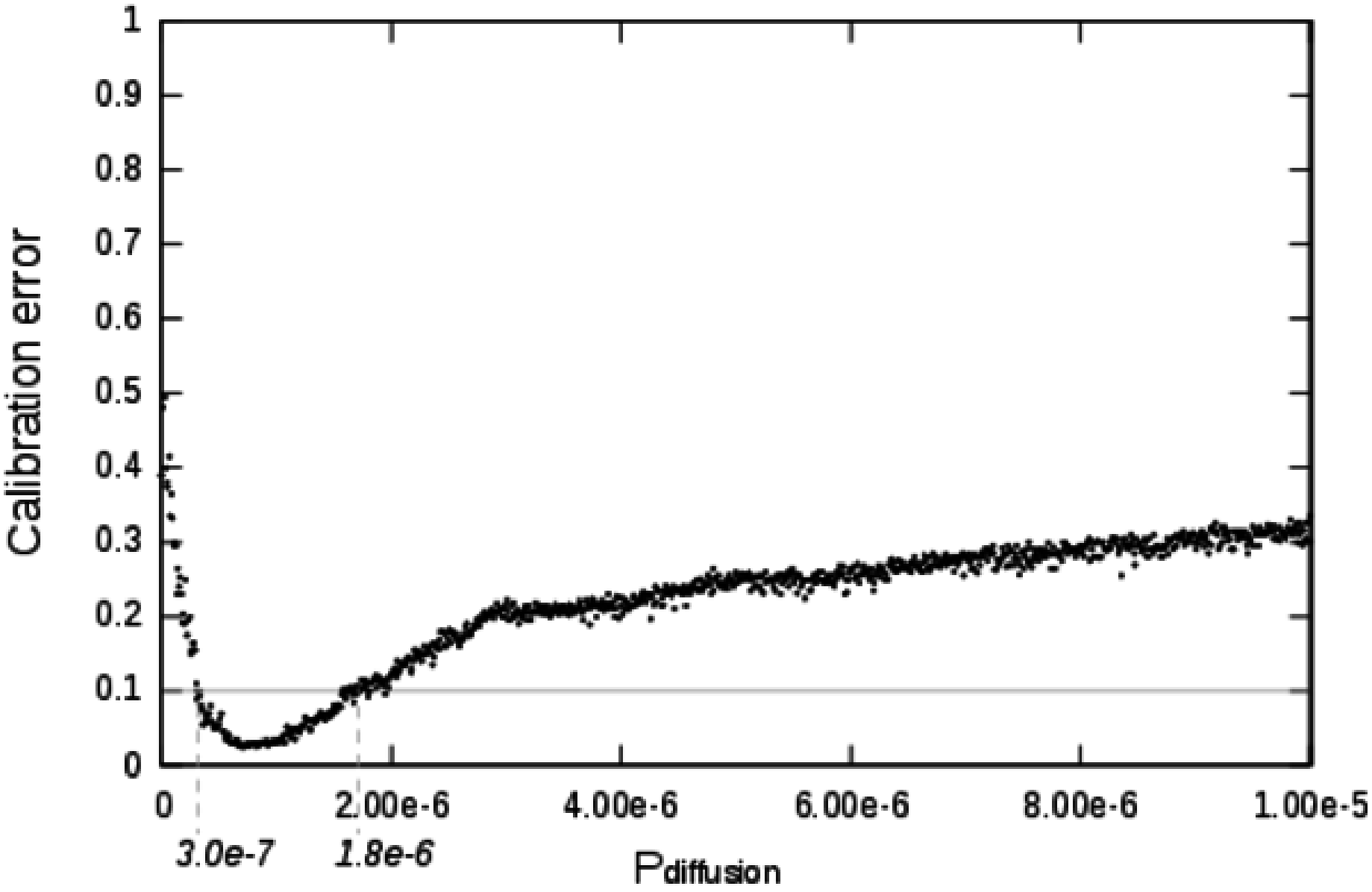




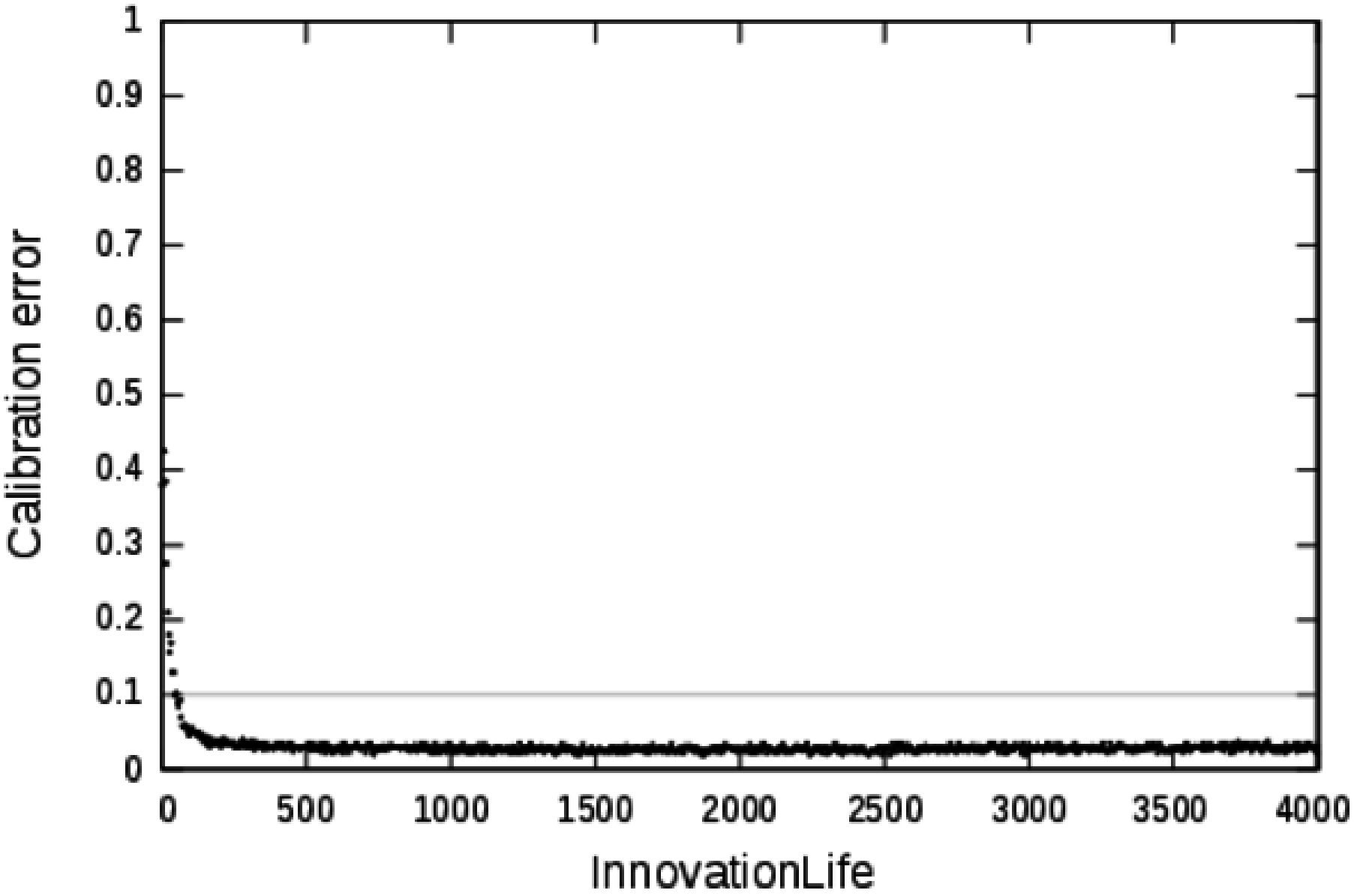




Results

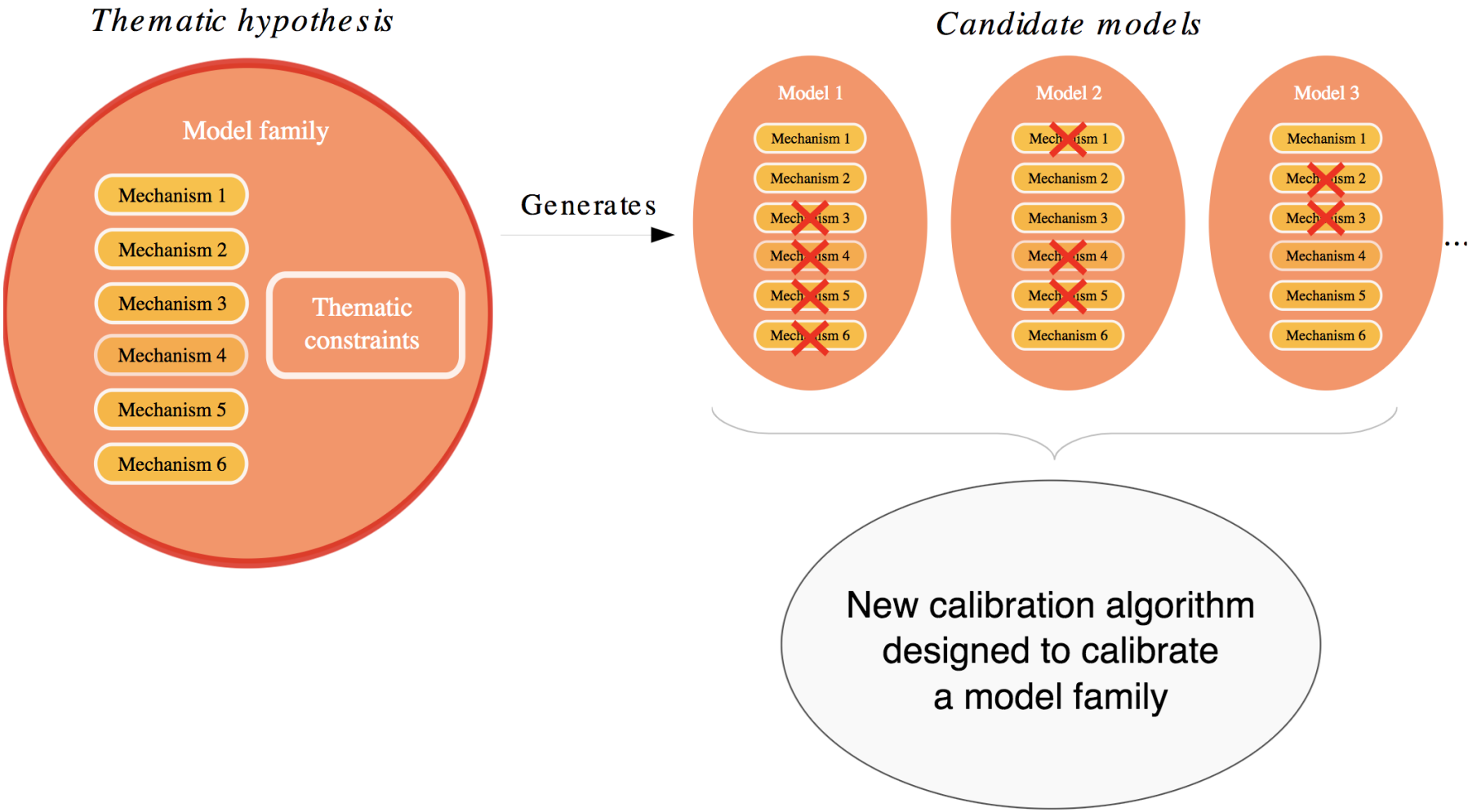

Calibration error
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
2.00e-6
4.00e-6
6.00e-6
8.00e-6
1.00e-5
3.0e-7
1.8e-6
Pdiffusion

Results

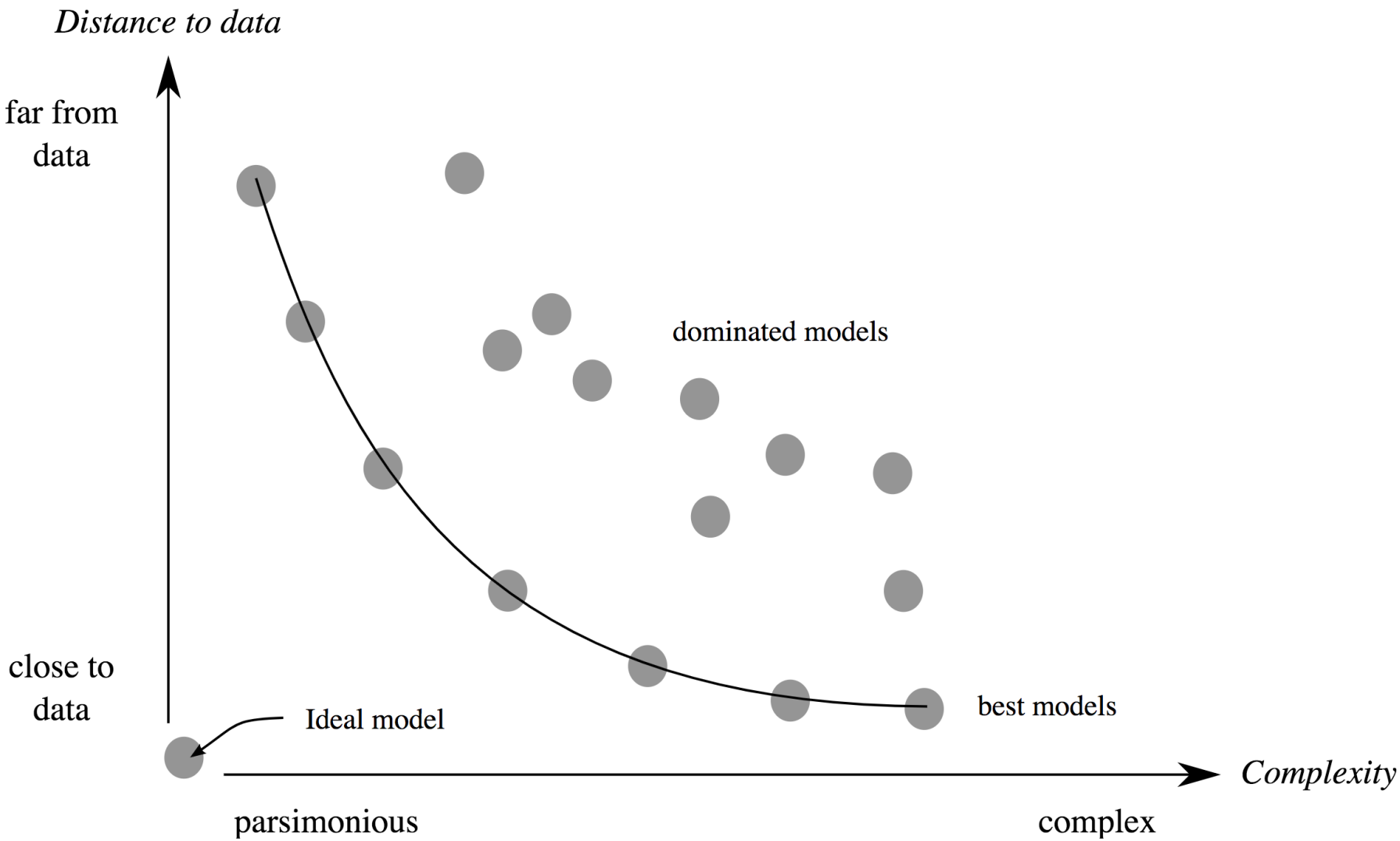

Calibration error
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
500
1000
1500
2000
2500
3000
3500
4000
InnovationLife

Unicity [Cottineau, 2014]

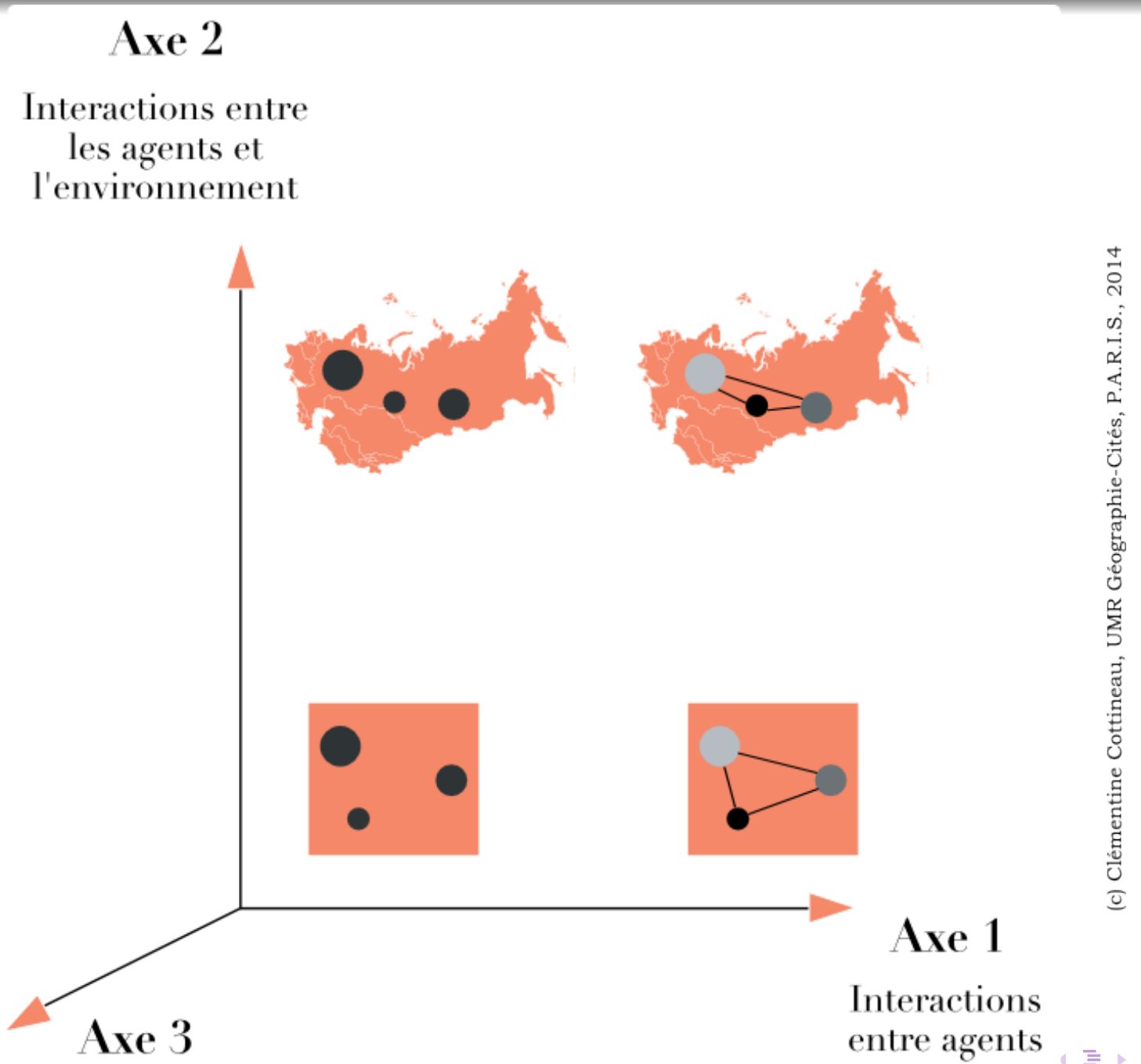
Automate the confrontation of alternative hypothesis / mechanisms.



Objective

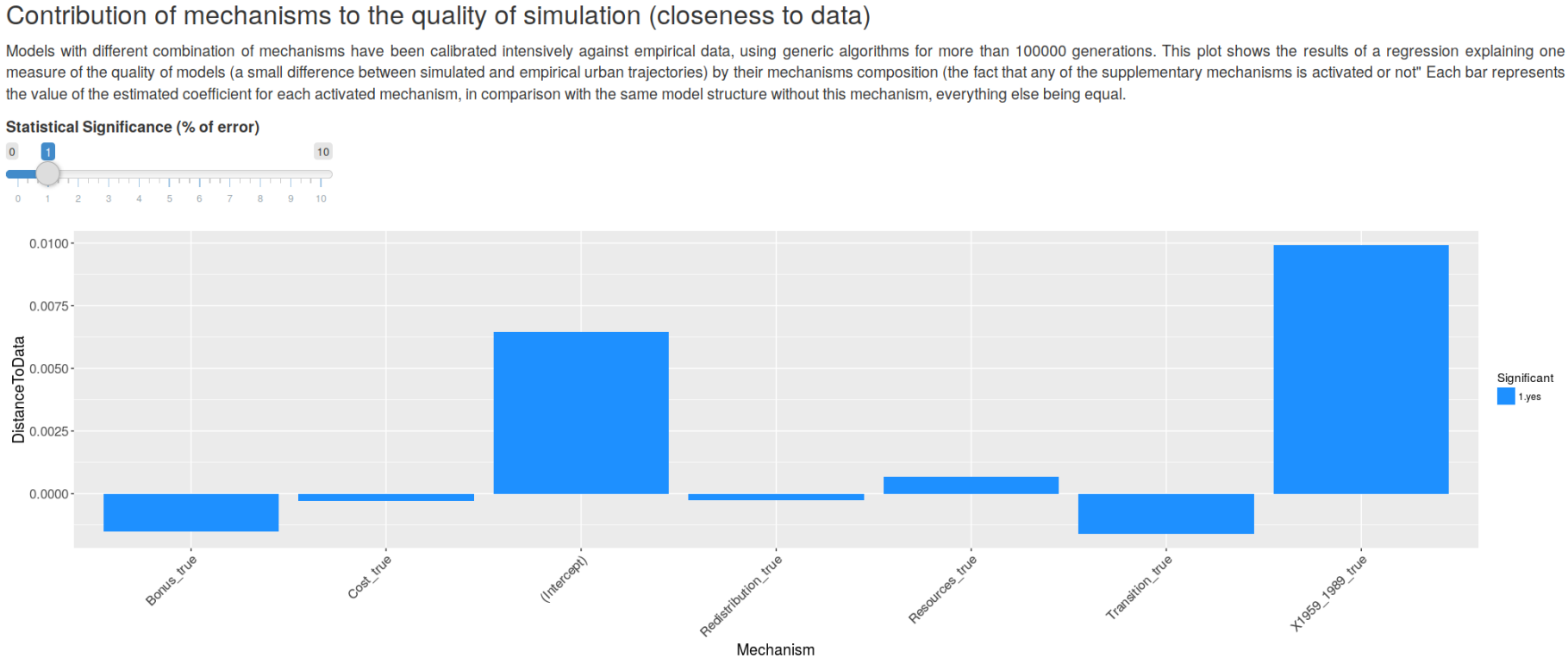

Distance to data
far from data
close to data
dominated models
best models
Ideal model
Complexity
parsimonious
complex

Multi-modeling (64 models)

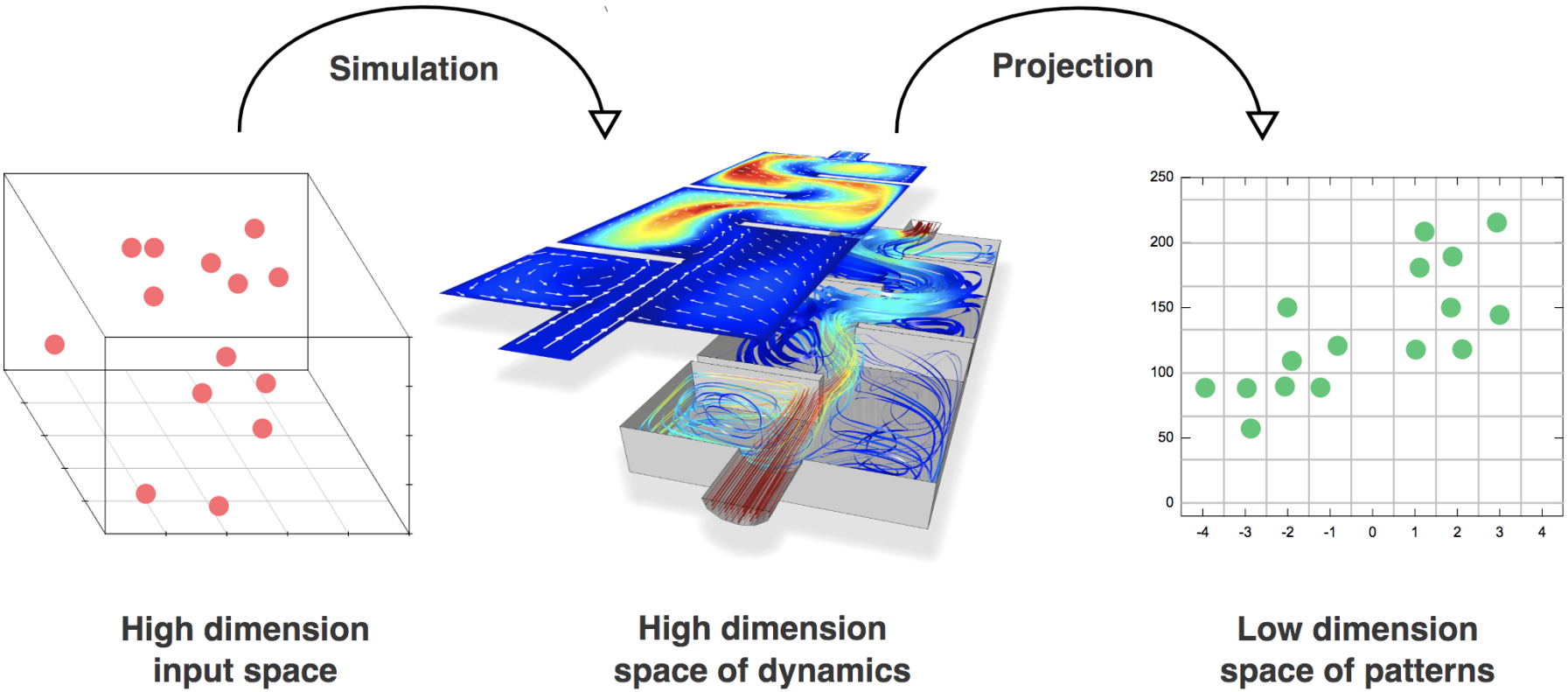


Calibration of the model family

Compute the best set of parameters for all 64 models, using a niched NSGA2 algorithm.

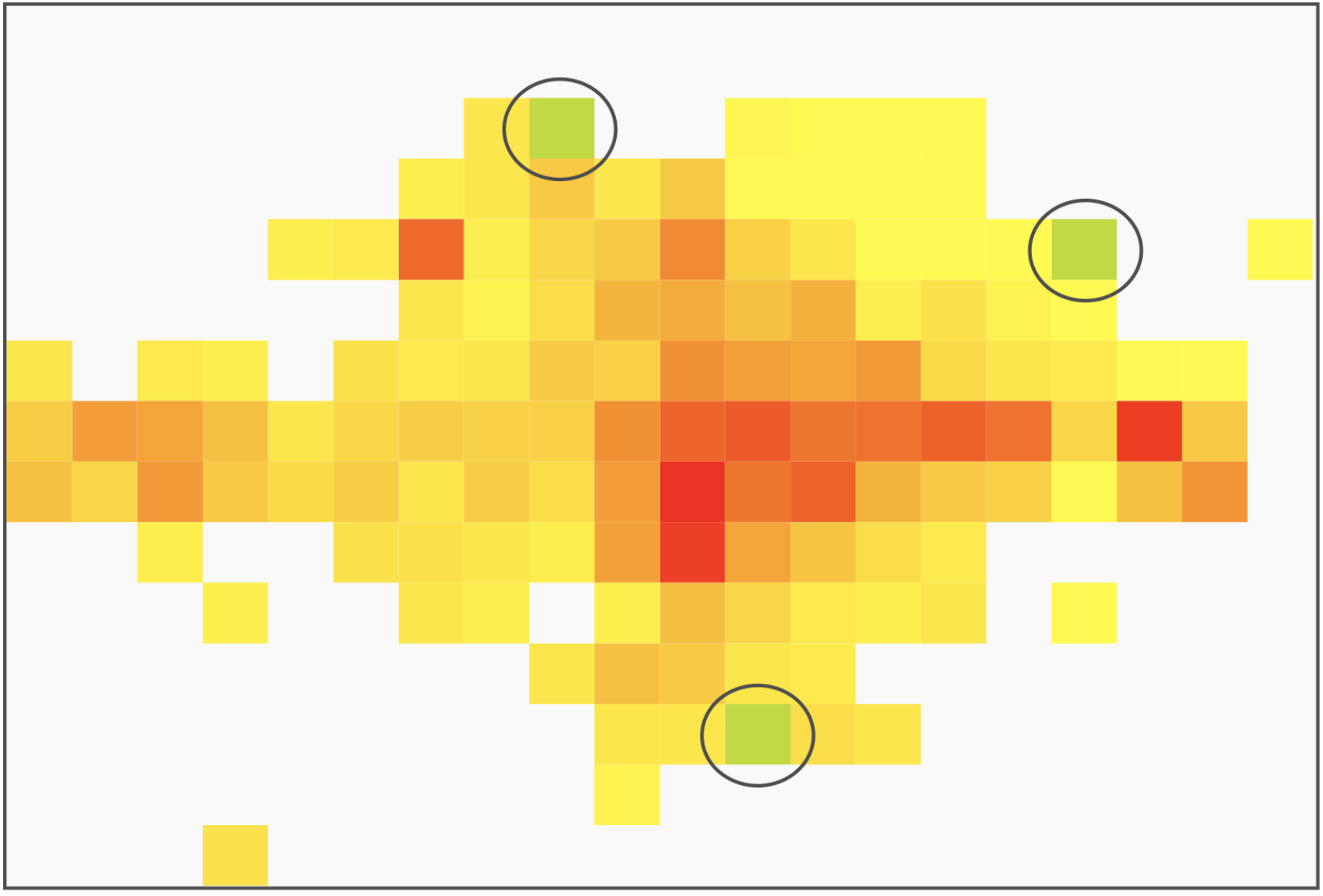


Novelty search [Chérel et al., 2015]

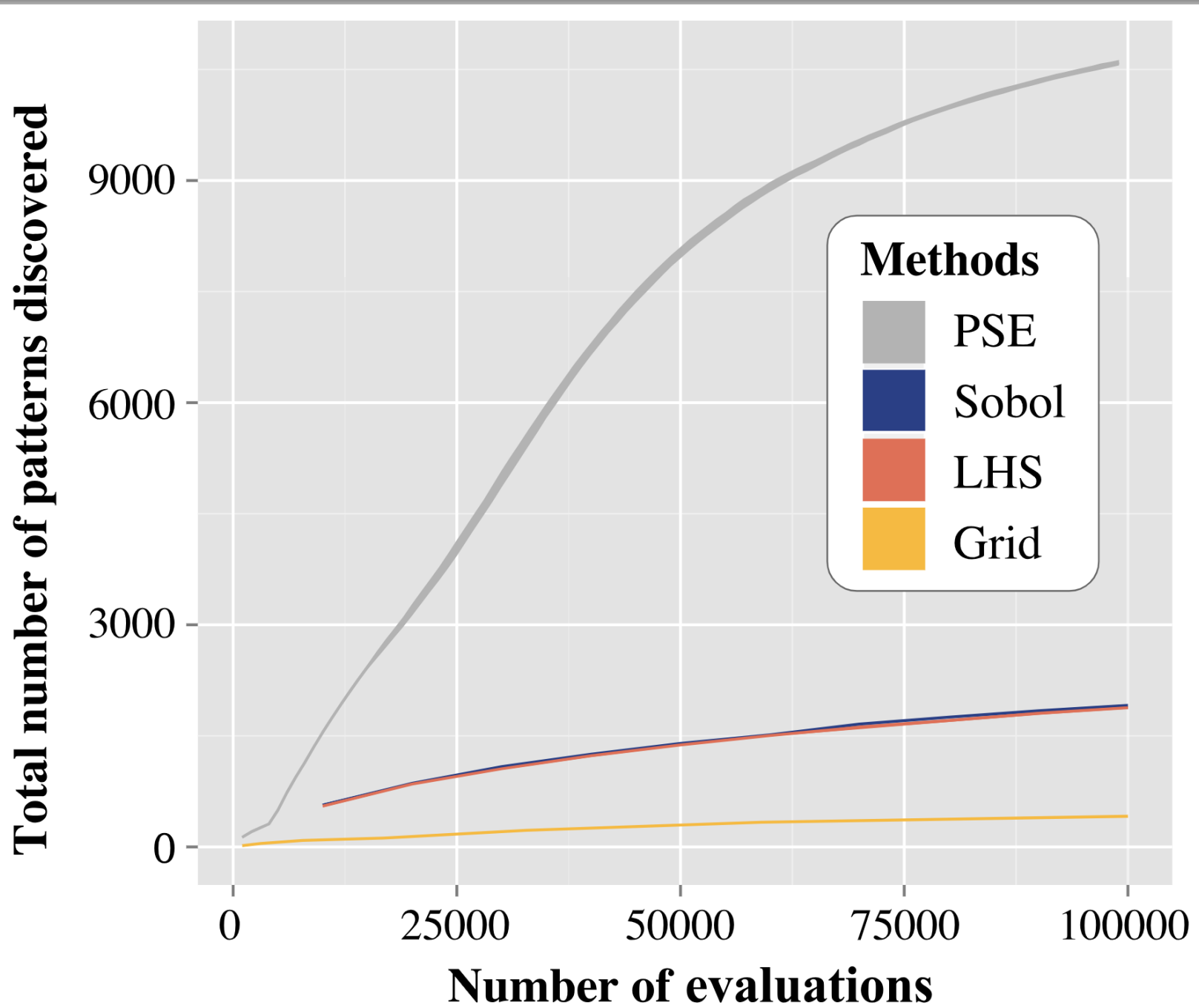


Novelty search

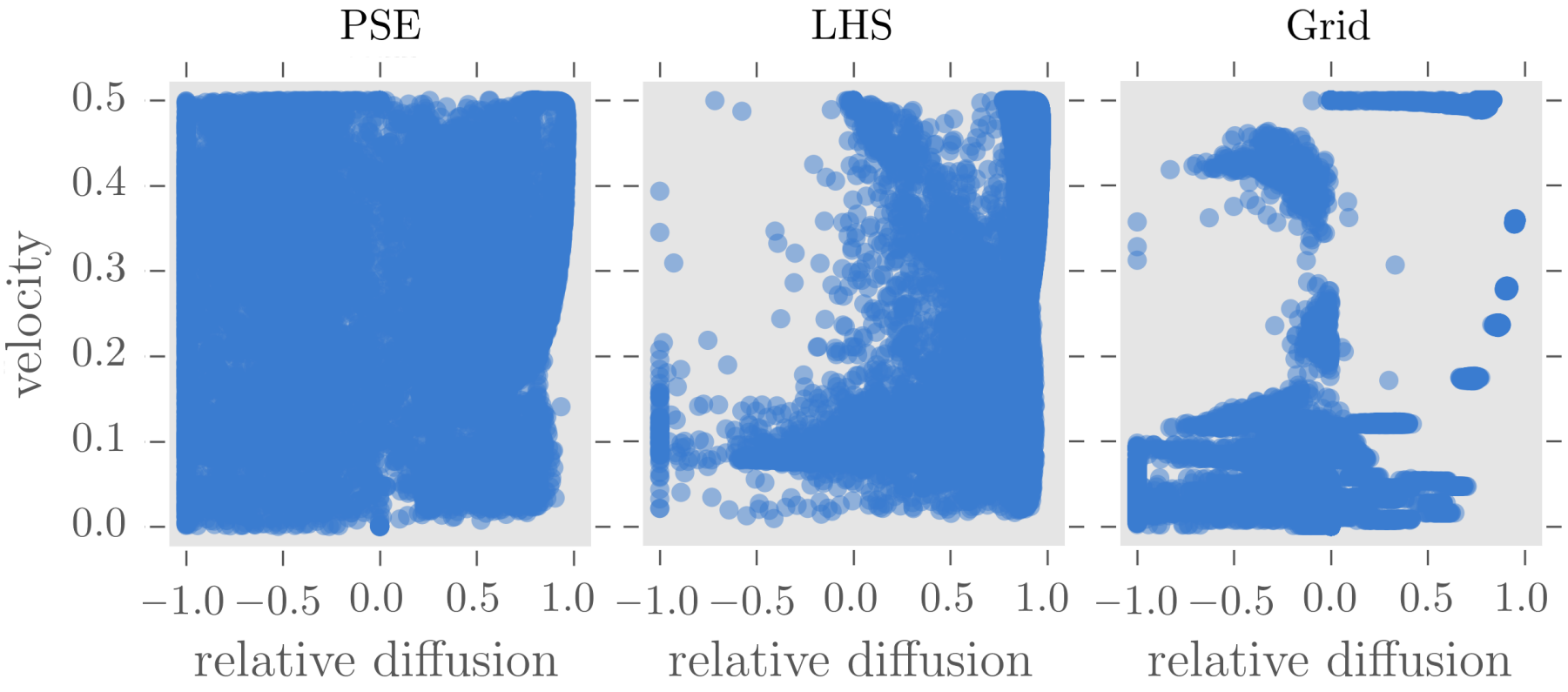
The inputs producing rare patterns have high fitness values.



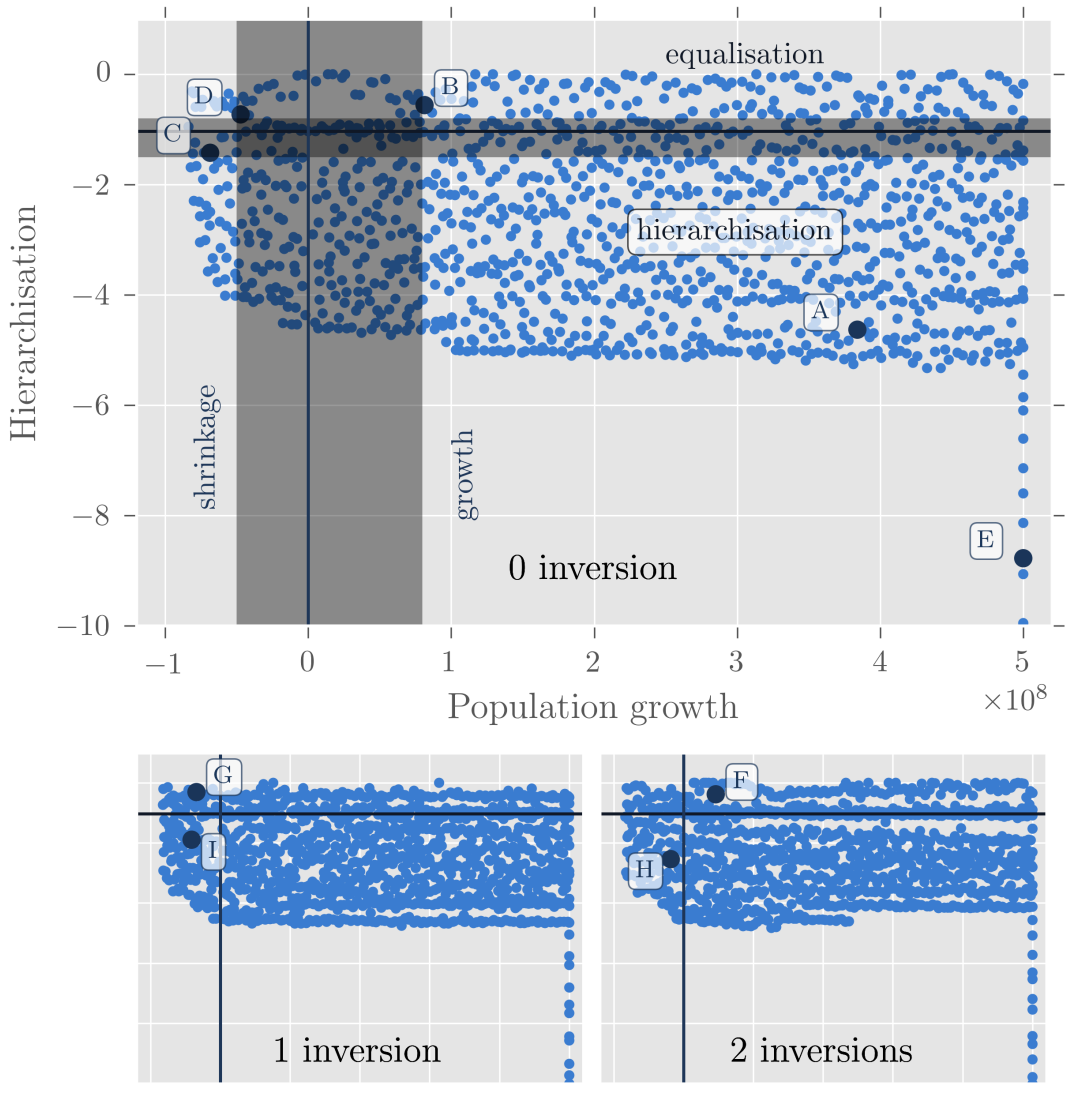
Results


Total number of patterns discovered
9000
6000
3000
0
0
25000
50000
75000
100000
Number of evaluations
Methods
PSE
Sobol
LHS
Grid

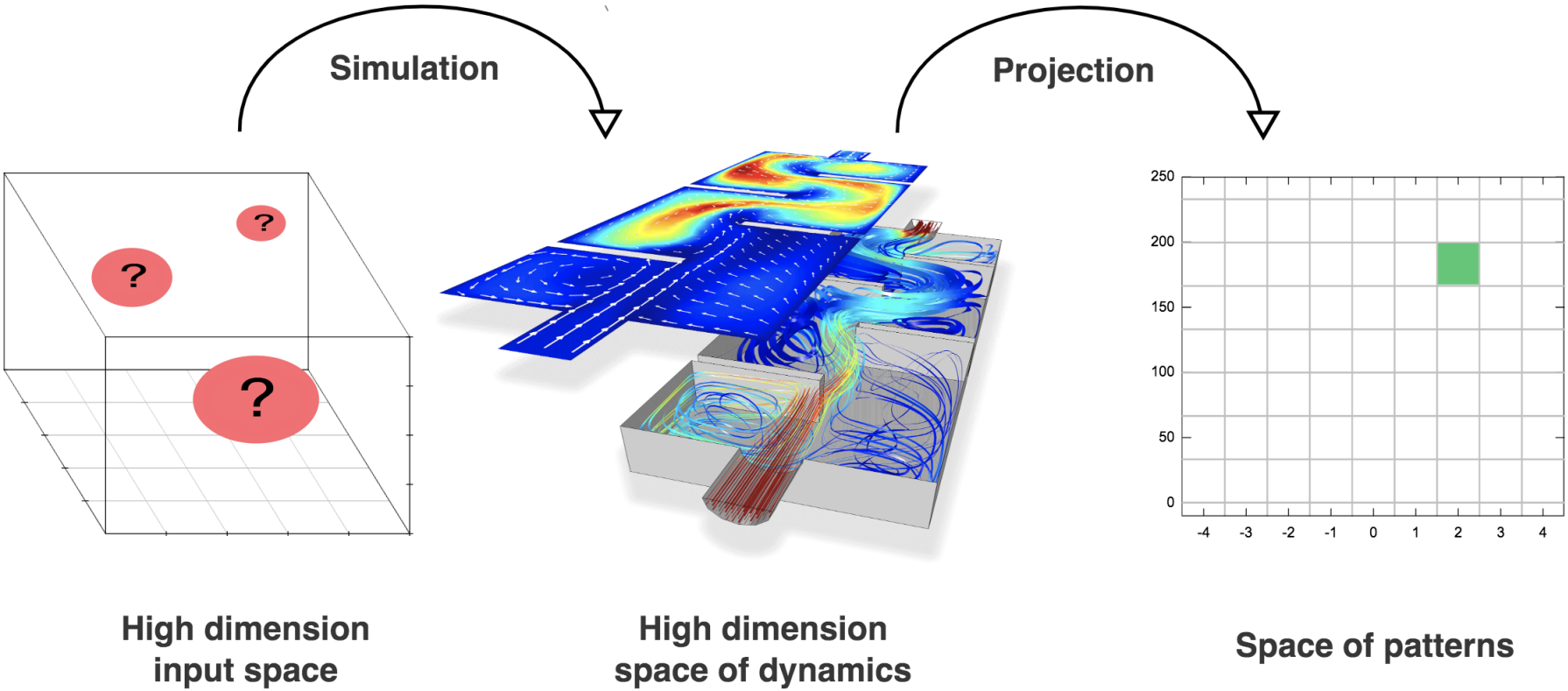
Results


PSE
LHS
Grid
velocity
0.5
0.4
0.3
0.2
0.1
0.0
−1.0 −0.5 0.0 0.5 1.0
relative diffusion

Results

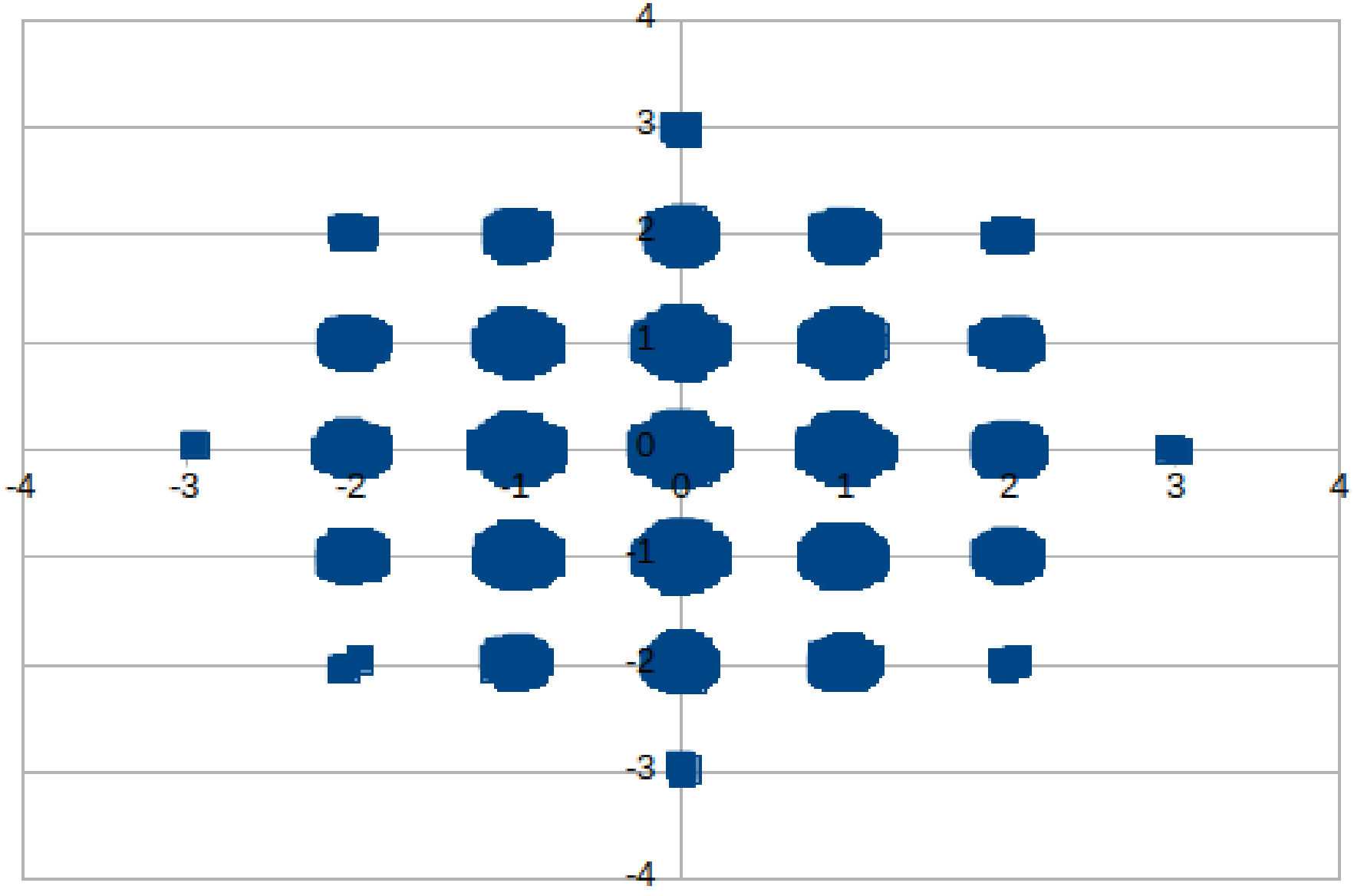

equalisation
hierarchisation
shrinkage
growth
0 inversion
1 inversion
2 inversions
A
B
C
D
E
F
G
H
I
Hierarchisation
Population growth
0
−2
−4
−6
−8
−10
−1
0
1
2
3
4
5
×10^8

Inverse problem

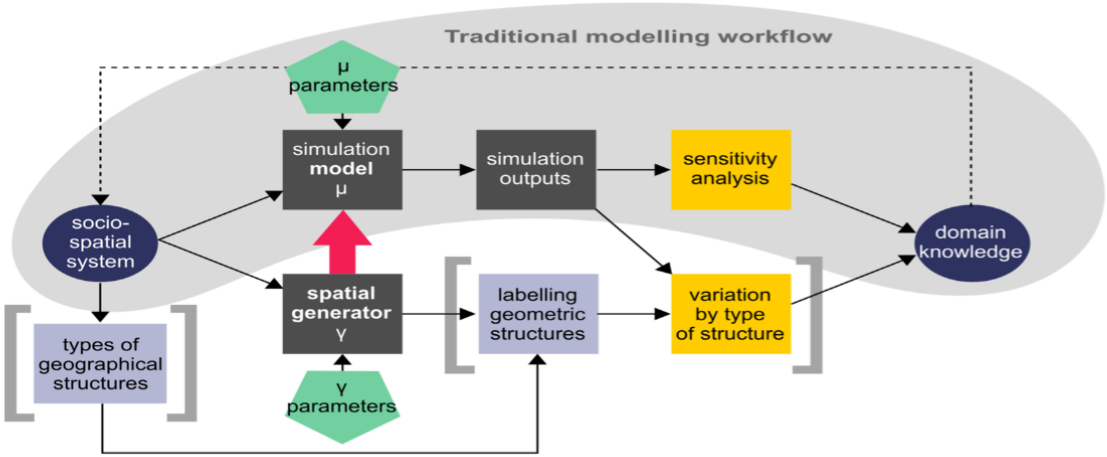


Results : minimising a Rastrigin function

Δ pattern $< \varepsilon$

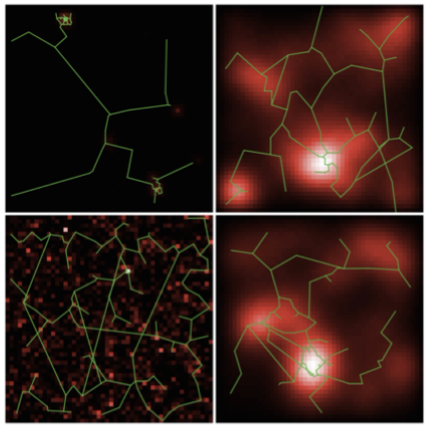


Spatial sensitivity analysis

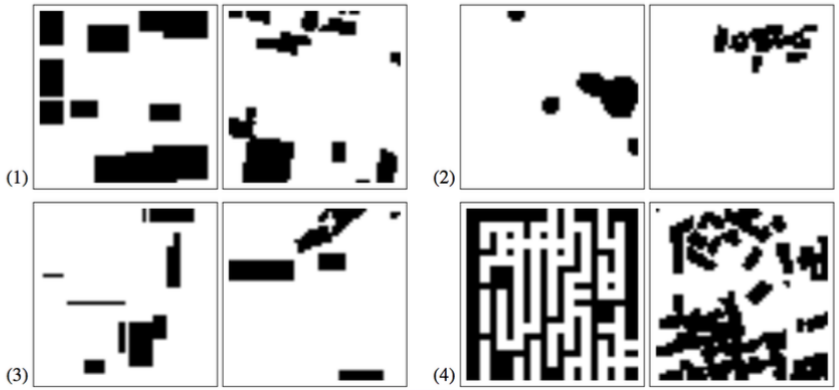


Raimbault, J., Cottineau, C., Le Texier, M., Le Nechet, F., Reuillon, R. (2019). Space Matters: Extending Sensitivity Analysis to Initial Spatial Conditions in Geosimulation Models. *Journal of Artificial Societies and Social Simulation*, *22*(4).

Raimbault, J., Perret, J., & Reuillon, R. (2020). A scala library for spatial sensitivity analysis. *GISRUK 2020 Proceedings, 32.*



Raimbault, J. (2019). Second-order control of complex systems with correlated synthetic data. *Complex Adaptive Systems Modeling*, *7*(1), 1-19.



Raimbault, J., Perret, J. (2019). Generating urban morphologies at large scales. In *Artificial Life Conference Proceedings* (pp. 179-186).

Model coupling

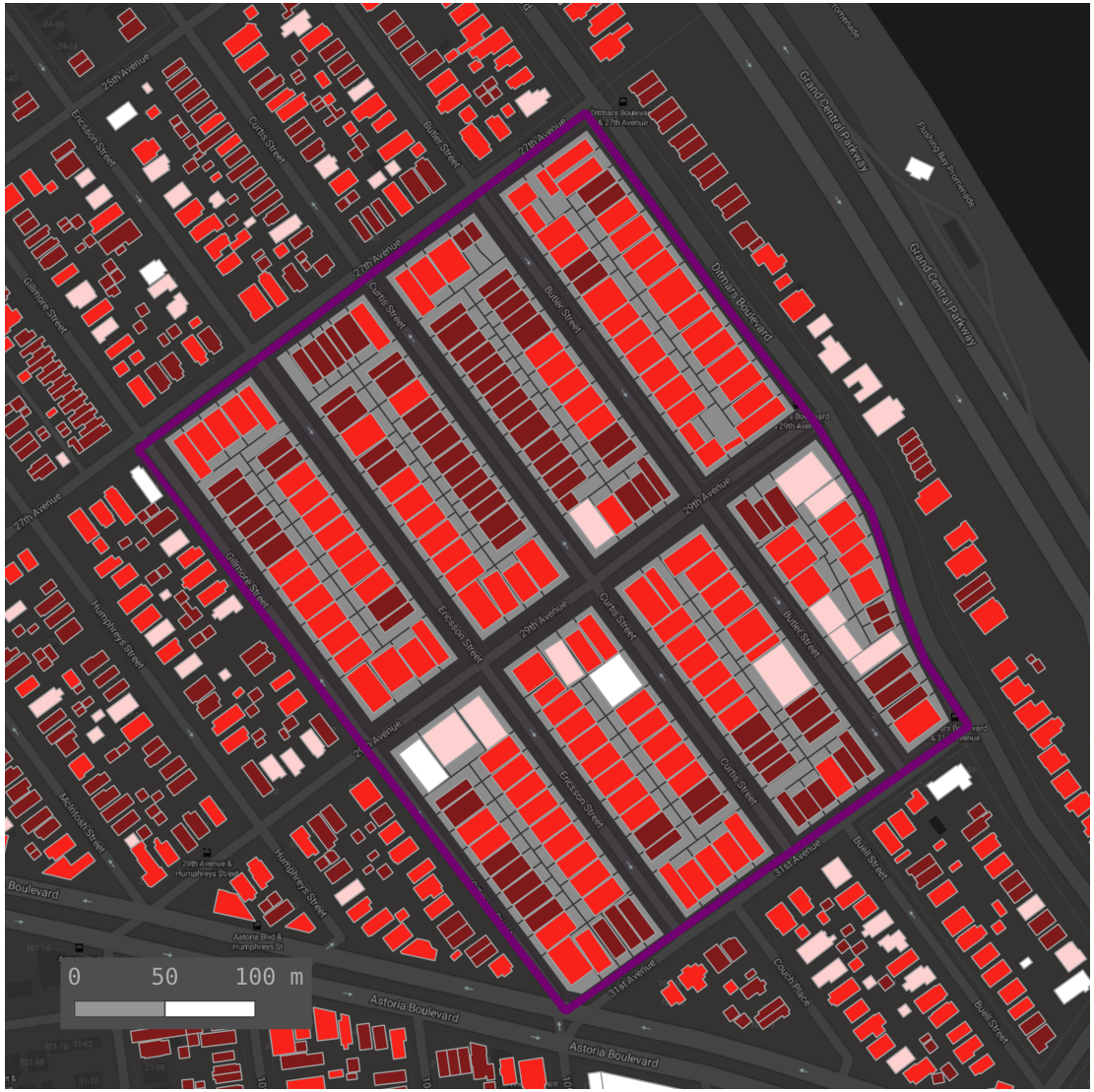
Transport model built using modular and open models

Modèles intégrés :

- MATSim (MATSim Community) for transport [W Axhausen et al., 2016]
- SPENSER (University of Leeds) for synthetic population [Spooner et al., 2021]
- QUANT (CASA, University College London) for spatial interactions [Batty and Milton, 2021]
- spatialdata library (OpenMOLE community) for spatial data [Raimbault et al., 2020]

Raimbault, J., & Batty, M. (2021). Estimating public transport congestion in UK urban areas with open transport models. GISRUK 2021 Proceedings.

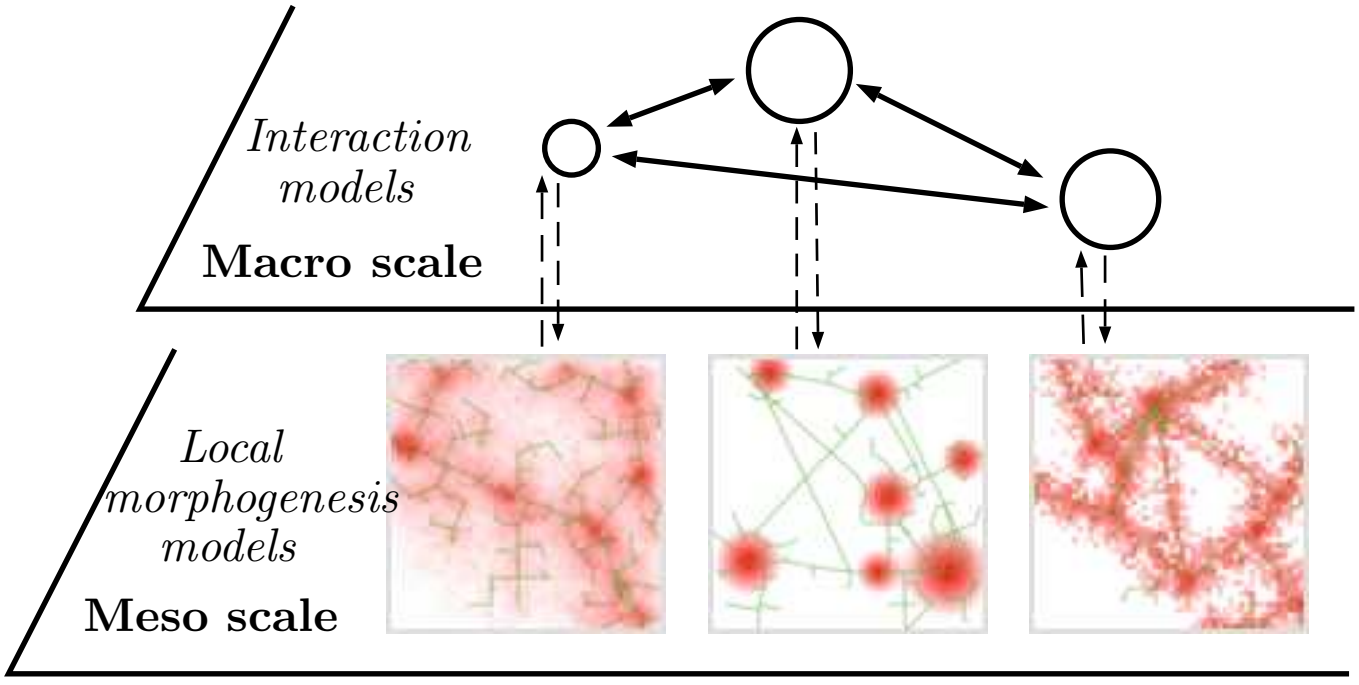
Model coupling: urbanism and heat island effect



SURe project (ASTIG, ISC-PIF, EPIDAPO collaboration)

$\rightarrow$ coupling the SimPLU3D model [Brasebin et al., 2017] with an UHI model.

Model coupling: towards multi-scale models



Processes specific to scales, coupling requires a dedicated ontology.

Raimbault, J. (2021). Strong coupling between scales in a multi-scalar model of urban dynamics. arXiv preprint arXiv:2101.12725.

Raimbault, J. (2021). A multiscale model of urban morphogenesis. arXiv preprint arXiv:2103.17241.

Raimbault, J. and Pumain, D. (2023). Innovation dynamics in multi-scalar systems of cities. *ALIFE 2023*.

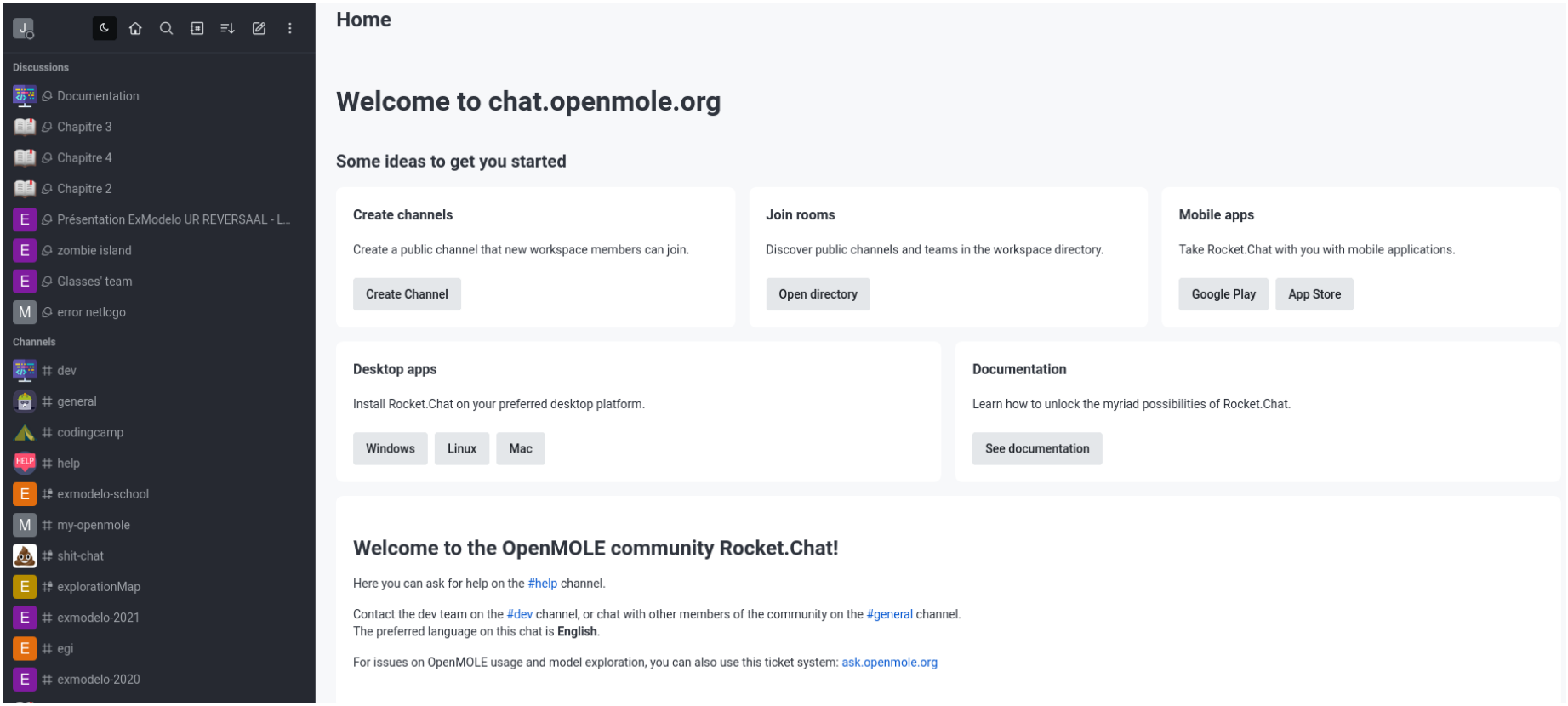
Synthesis: OpenMOLE benchmark

Type	Critères	Apache Taverna	Spark	Jupyter	R	Dakota	OpenTURNS	PEST++	OpenMOLE
Mo	Appel d'executable	✓	✓	✓	✓	✓	✓	✓	✓
Mo	Execution de containers	✗	✗	✗	✗	✗	✗	✗	✓
Mo	Transmission transparente de données structurées	✓	✗	✓	✗	✗	✗	✗	✓
Me	Méthodes d'exploration	✗	✗	✗	✓	✓	✓	✓	✓
Me	Échantillonage adaptatif	✗	✗	✗	✓	✓	✓	✓	✓
Me	Optimisation globale	✗	✗	✗	✓	✓	✓	✓	✓
Me	Recherche de diversité	✗	✗	✗	✗	✗	✗	✗	✓
E	Calcul distribué	✓	✓	✗	✓	✓	✗	✓	✓
E	Zero-deploiement	✗	✗	✗	✗	✗	✗	✗	✓
C	Communauté exploration de modèles	✗	✗	✗	~	✓	✓	✓	✓
I	Logiciel installable	✓	✓	✓	✓	✓	✓	✓	✓
I	Service en ligne	✓	✓	✓	✓	✗	✗	✗	✓
I	Langage généraliste	✗	✓	✓	✓	✗	✓	✗	✓
I	Système de workflow	✓	✗	✗	✗	✗	✓	✗	✓
I	Programation Graphique	✓	✗	✗	✗	✗	✓	✗	✗

Use OpenMOLE

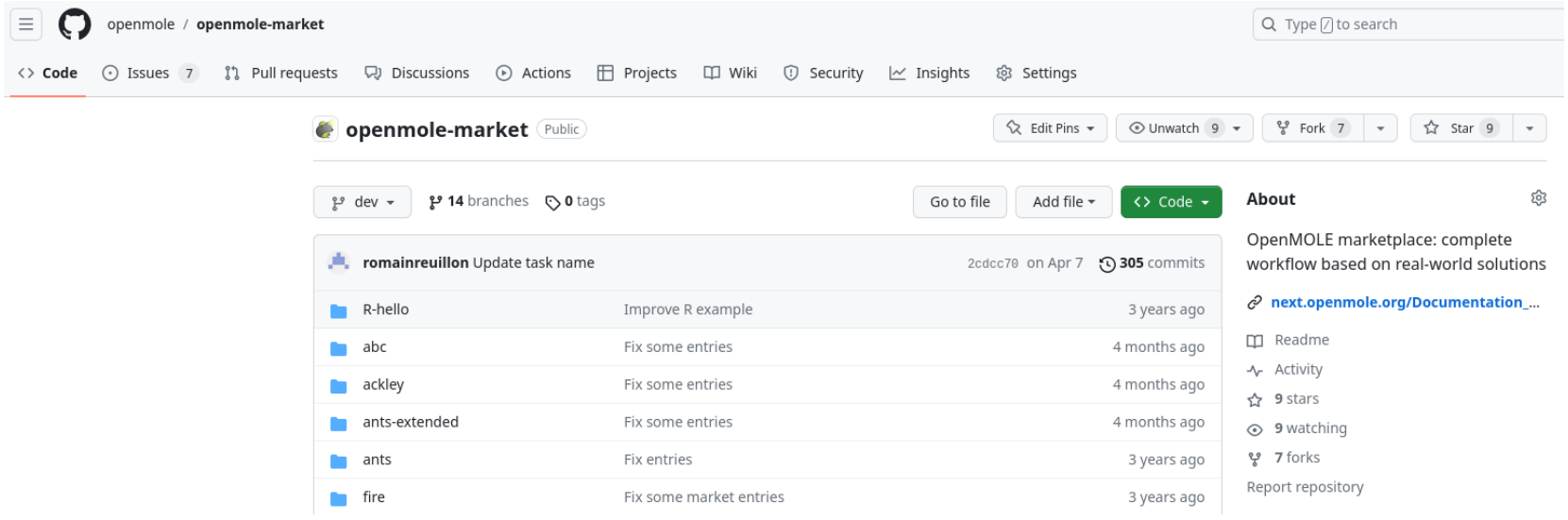
→ Java executable without installation at `https://openmole.org`, code source to compile at `https://github.com/openmole/openmole`
→ Setup an online instance

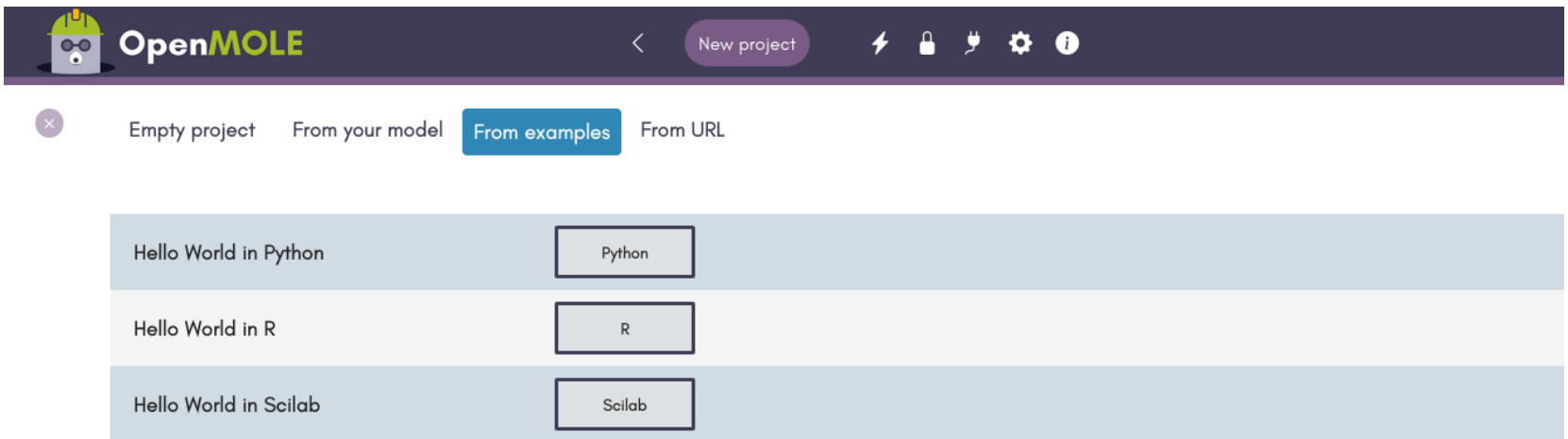
Help and contribution



Very reactive chat: `https://chat.openmole.org`

Contribution





Add thematic example to the script market

`https://github.com/openmole/openmole-market`

Learning

Learning two days workshop (ISC-PIF, November)
`https://workshop.exmodelo.org/`: applications open

R&D and scientific consulting

https://trempoline.io/

Summary of OpenMOLE positioning

A qualitative shift in knowledge that can be extracted from a simulation model with model exploration methods.

Application in different disciplines: Geography [Schmitt et al., 2015][Chérel et al., 2015], Ecological modeling [Lavallée et al., 2018], epidemiology [Arduin, 2018], etc.

Main characteristics:

→ Complementary role of the three axis: methods, computing environments, model embedding.

→ Iterative and integrated construction of models and theories.

→ Model coupling and reproducibility made easy through the scripting language [Passerat-Palmbach et al., 2017].

References I

Arduin, H. (2018).
Modélisation mathématique des interactions entre pathogènes chez l'hôte humain: Application aux virus de la grippe et au pneumocoque.
PhD thesis, Université Paris-Saclay.

Batty, M. and Milton, R. (2021).
A new framework for very large-scale urban modelling.
Urban Studies, 58(15):3071–3094.

Brasebin, M., Chapron, P., Chérel, G., Leclaire, M., Lokhat, I., Perret, J., and Reuillon, R. (2017).
Apports des méthodes d'exploration et de distribution appliquées à la simulation des droits à bâtir.
In *Spatial Analysis and GEOmatics 2017*.

References II


Chérel, G., Cottineau, C., and Reuillon, R. (2015).
Beyond corroboration: Strengthening model validation by looking for unexpected patterns.
PLoS ONE, 10(9):e0138212.


Cottineau, C. (2014).
L'évolution des villes dans l'espace post-soviétique. Observation et modélisations.
PhD thesis, Université Paris 1 Panthéon-Sorbonne.


Lavallée, F., Alvarez, I., Dommanget, F., Martin, S., Reineking, B., and Smadi, C. (2018).
A dynamical model for the growth of a stand of japanese knotweed including mowing as a management technique.
In *Conference on Complex Systems 2018*.

References III


Passerat-Palmbach, J., Reuillon, R., Leclaire, M., Makropoulos, A., Robinson, E. C., Parisot, S., and Rueckert, D. (2017).
Reproducible large-scale neuroimaging studies with the openmole workflow management system.
Frontiers in neuroinformatics, 11:21.


Pumain, D. (2018).
An evolutionary theory of urban systems.
In *International and Transnational Perspectives on Urban Systems*, pages 3–18. Springer.


Raimbault, J. (2018).
Extracting knowledge from simulation models: trends and perspectives from the viewpoint of quantitative geography.
In *Conference on Complex Systems 2018*.

References IV


Raimbault, J. (2019).
Second-order control of complex systems with correlated synthetic data.
Complex Adaptive Systems Modeling, 7(1):1–19.


Raimbault, J. (2021a).
A multiscale model of urban morphogenesis.
arXiv preprint arXiv:2103.17241.


Raimbault, J. (2021b).
Strong coupling between scales in a multi-scalar model of urban dynamics.
arXiv preprint arXiv:2101.12725.


Raimbault, J. and Batty, M. (2021).
Estimating public transport congestion in uk urban areas with open transport models.
GIS Research UK (GISRUK).

References V


Raimbault, J., Cottineau, C., Le Texier, M., Le Nechet, F., and Reuillon, R. (2019).
Space matters: Extending sensitivity analysis to initial spatial conditions in geosimulation models.
Journal of Artificial Societies and Social Simulation, 22(4).


Raimbault, J. and Perret, J. (2019).
Generating urban morphologies at large scales.
In *Artificial Life Conference Proceedings*, pages 179–186. MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info . . . .


Raimbault, J., Perret, J., and Reuillon, R. (2020).
A scala library for spatial sensitivity analysis.
GISRUK.

References VI


Reuillon, R., Schmitt, C., De Aldama, R., and Mouret, J.-B. (2015).
A new method to evaluate simulation models: The calibration profile (cp) algorithm.
Journal of Artificial Societies and Social Simulation, 18(1):12.


Schmitt, C., Rey-Coyrehourcq, S., Reuillon, R., and Pumain, D. (2015).
Half a billion simulations: Evolutionary algorithms and distributed computing for calibrating the simpoplocal geographical model.
Environment and Planning B: Planning and Design, 42(2):300–315.


Spooner, F., Abrams, J. F., Morrissey, K., Shaddick, G., Batty, M., Milton, R., Dennett, A., Lomax, N., Malleson, N., Nelissen, N., et al. (2021).
A dynamic microsimulation model for epidemics.
Social Science & Medicine, 291:114461.

References VII


W Axhausen, K., Horni, A., and Nagel, K. (2016).
The multi-agent transport simulation MATSim.
Ubiquity Press.