Limitations

Below is a necessarily incomplete list of current limitations of Morpheus. Many of these are on our to-do list or under development.

Lattice

Morpheus uses regular lattice discretization for all spatial processes. It does not support irregular lattices of continuous space models.

Solvers

The solvers in Morpheus use finite difference schemes with fixed time-steps. Adaptive time-stepping schemes are not available. It does not support solving of stiff ODE systems.

PDE

Morpheus can simulate reaction-diffusion systems, but does not support processes that depend on higher-order derivatives such as advection or chemotaxis.

Cell motility

Morpheus supports the cellular Potts model to represent cell shape, motility and adhesion. Depending on the scale at which cell-based models are used, motility of cells may be more suitably modeled in more parsimonious formalisms, such as interacting particle systems. These are currently not available in Morpheus.

Extensibility

Morpheus is an extensible framework and provides a plugin interface to extend its features. However, this requires building from source, which is currently restricted to developers and collaborators. Therefore, in its current distribution as an application (binaries), extensibility is not supported.

Performance

Morpheus heavily relies on muparser for parsing mathematical expressions. Although this provides fast parsing by translating into bytecode, its performance cannot compete with compiled C++ code. Thus, the flexibility gained by using parsed expression invokes a penalty on computational performance. The amount of penalty depends on the number and complexity of parsed expressions and is highly model-dependent. Therefore, in some cases, Morpheus may not be best-suited for large-scale simulations.

See the results of performance and scalability tests for more detailed and quantitative information.

Parallelization

Morpheus uses openMP multithreading to concurrently execute several data-parallel processes. For instance, intracellular ODEs are computed concurrently for each cell, and the reaction-step in PDEs are computed concurrently per y-line. Although this is a good first approximation, it does not fully exploit parallel computing. Scalability therefore heavily depends on the specific simulation model. CPM (sub)models, for instance, are not parallelized.

See the results of performance and scalability tests for more detailed and quantitative information.