What's New

Morpheus 2.0 is the 2nd major release that improves modeling flexibility, usability, and allowing extensibility.

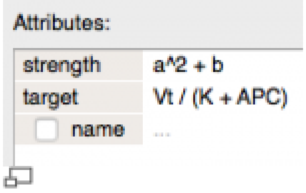
Open source

- Morpheus 2 is now **open source**.
    - Source code is available in the Git repository.
    - Released under a permissive BSD license
- Morpheus 2 is an **extensible framework**
    - Modeling and analysis tools can be customized or added as plugins
    - A stable C++ API has been developed for plugin development
        - Simple integration of XML and implementation
        - Automated symbol dependency tracking and scheduling

Simulator

Modeling

- **Expressions everywhere**
    - In addition to numerical values, plugin can now take mathematical expressions as inputs
    - Greatly improves modeling flexibility by coupling of submodels
    - CPM temperature can now also be function of time, e.g. to simulate annealing.



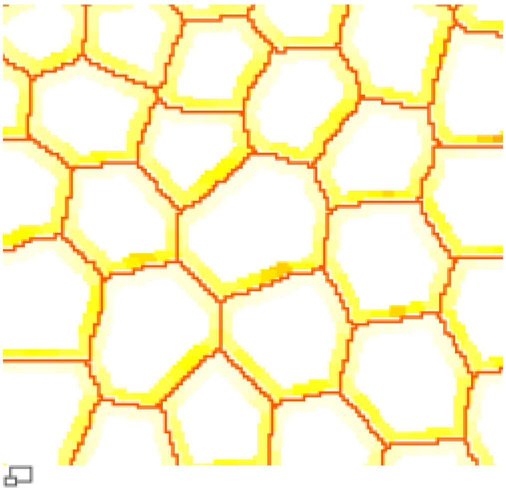
Attributes:

strength	a^2 + b
target	Vt / (K + APC)
☐ name	...

Parameters can given as math expressions, interpreted at run-time.

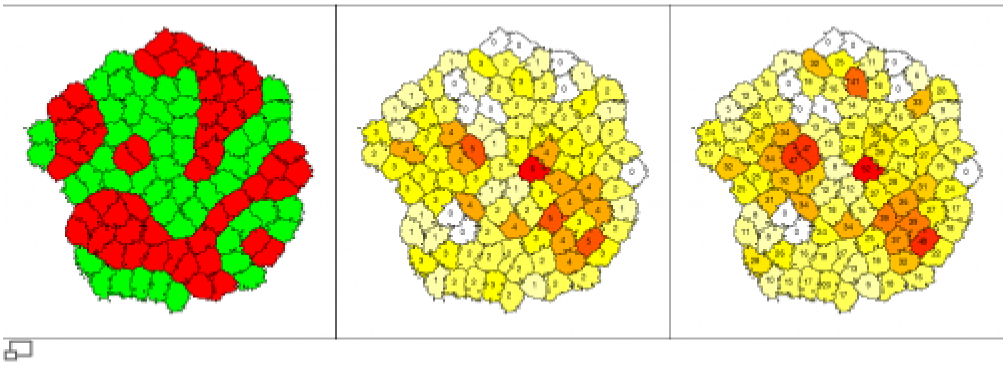
- **CPM surface scaling**
    - New correction terms for scaling of surface of CPM cells
    - Now calculated in accordance to the scaling rules as specified in Magno et al., BMC Biophysics, 2015

- **MembraneProperties**
    - Variables with spatial resolution, mapped to cell membrane
    - Uses circular (2D) or spherical (3D) approximation of cell shape
    - Allows spatially resolved registration of (neighboring) cell properties
    - Enables reaction-diffusion of properties on membrane



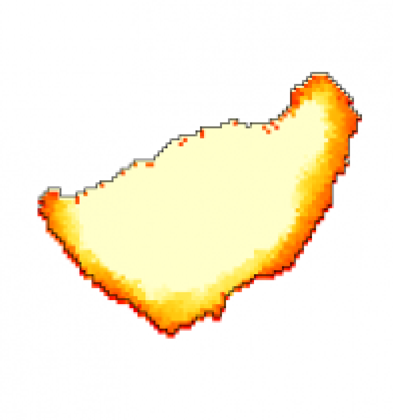
MembraneProperties provide a scalar field mapped to the cell membrane on which reaction-diffusion systems can be simulated.

- **Reporters**
  - All `Reporters`, plugins that generate statistics and mappings, have been revised and generalized.
  - `NeighborhoodReporter` provides statistics (e.g.sum, mean) on properties of the cell's directly adjacent microenvironment
  - `CellReporter` provides statistics on properties of the cellular environment such as including the concentration of `MembraneProperties` and `Fields`.



NeighborhoodReporter counts the number of neighbors of the other cell type. Left: colors show two cell types. Middle: colors and labels show number of non-identical neighboring cells. Right: colors and labels show length of interface to non-identical neighboring cells

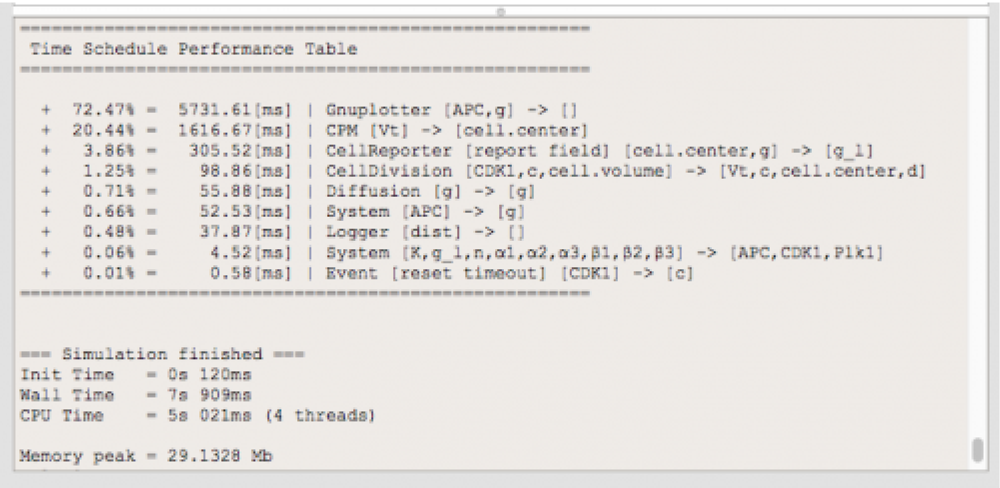
- A number of **new plugins** introduce new modeling features, including
  - `AddCell`: add CPM cells during simulation based on a condition and a location specified by a probability density function.
  - `ChangeCelltype`: change the cell type of a cell based on a condition.
  - `Protrusion`: actin-inspired feedback model that control shape and motility of CPM cells, as proposed by Niculescu et al., 2015.



The new Protrusion plugin models positive feedback of active membrane regions, inspired by actin dynamics.

Performance

- **Performance optimization**
  - Automatic scheduling has been revised and optimized by subdividing according to spatial contexts.
  - Tracking of CPM object properties has been moved to an adaptive tracker, tracking only the required information
  - All expressions are now in thread-safe containers to optimize use of multithreading.
- **Performance measurements and profiling**
  - Information on execution time (wall time, CPU time) and peak memory usage
  - Execution times (wall time) are listed per plugin

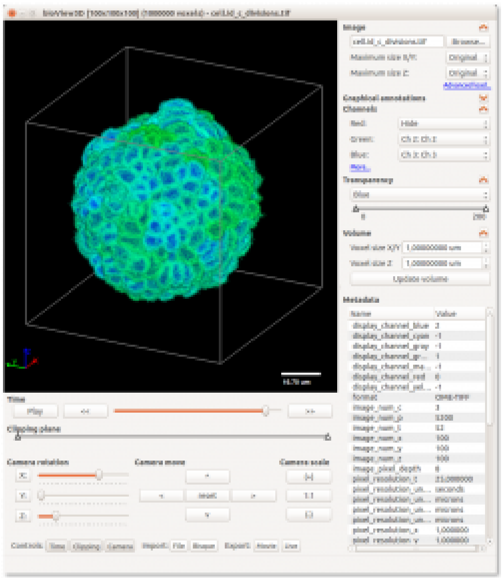


```
=======================================================
 Time Schedule Performance Table
=======================================================

  +  72.47% =  5731.61[ms] | Gnuplotter [APC,g] -> []
  +  20.44% =  1616.67[ms] | CPM [Vt] -> [cell.center]
  +   3.86% =   305.52[ms] | CellReporter [report field] [cell.center,g] -> [g_l]
  +   1.25% =    98.86[ms] | CellDivision [CDK1,c,cell.volume] -> [Vt,c,cell.center,d]
  +   0.71% =    55.88[ms] | Diffusion [g] -> [g]
  +   0.66% =    52.53[ms] | System [APC] -> [g]
  +   0.48% =    37.87[ms] | Logger [dist] -> []
  +   0.06% =     4.52[ms] | System [K,g_l,n,α1,α2,α3,β1,β2,β3] -> [APC,CDK1,Plk1]
  +   0.01% =     0.58[ms] | Event [reset timeout] [CDK1] -> [c]
=======================================================


=== Simulation finished ===
Init Time    = 0s 120ms
Wall Time    = 7s 909ms
CPU Time     = 5s 021ms (4 threads)

Memory peak = 29.1328 Mb
```

A performance profile is generated after each simulation.

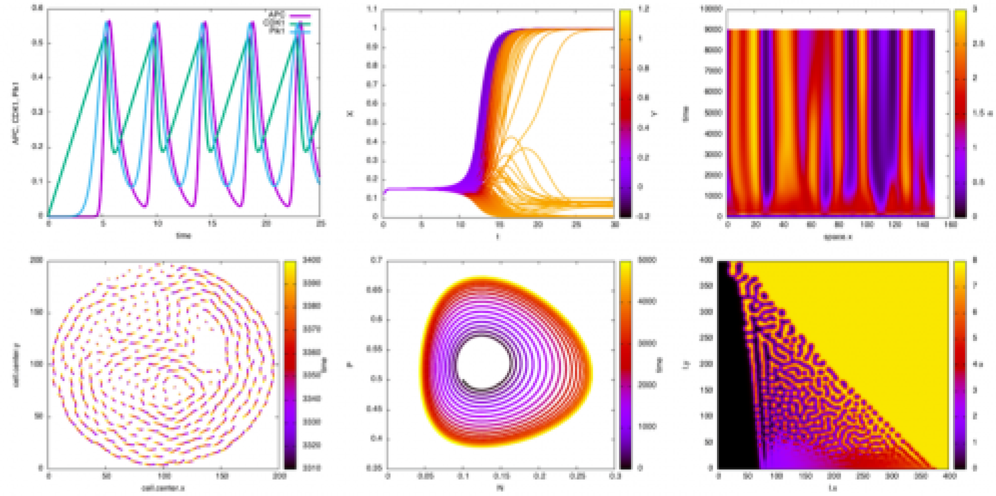
Analysis

- **TIFFPlotter**
  - Now writes OME-TIFF headers (Open Microscopy Environment) to facilitate visualization of 3D-5D image data with external such as Fiji or BioView3D.



Morpheus' TIFFPlotter exports time-lapse multichannel TIFF stacks with OME-TIFF header can be easily visualized with external tools such as BioView3D.

- **Logger**
  - The completely revised `Logger` plugin provides generic tool for data export.
  - Data export: Periodically writes any combination of variables (Globals, Properties, Fields, etc.) to file in CSV or Matrix format.
  - Plotting: Logger also provides a versatile interface for generating data visualizations including time plots, phase diagrams, cell trajectory plot, spacetime plot, profile plots, surface plots etc.



The new `Logger` plugin provides a versatile interface for data export and data visualization.

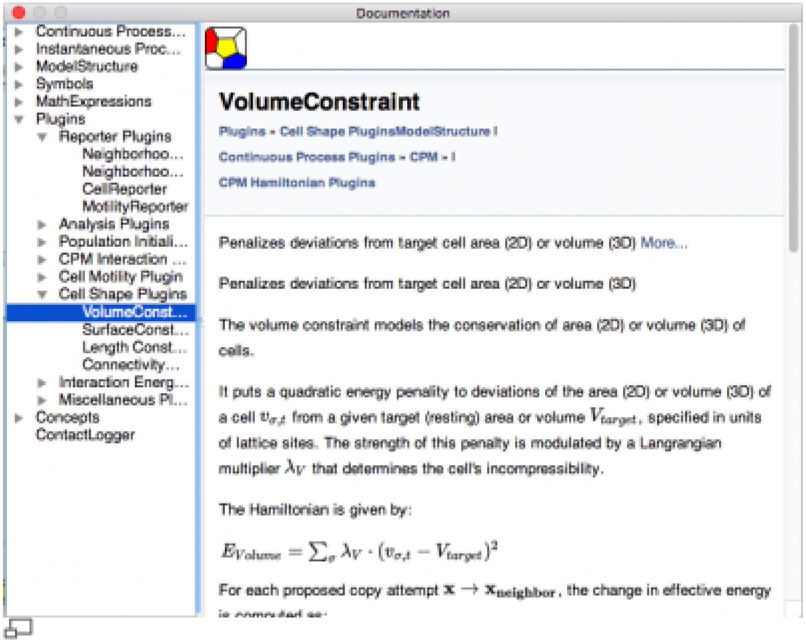
MorpheusML

- **Scoping**
  - Lexical scoping was introduced: symbols are only valid on the scope in which they are declared.
  - Improves intuitive character and consistency of model description language
  - Allows multiple usages of symbols, setting global defaults, etc.

- **Globals**
    - New Globals section for global constants, variables and `Fields`
    - Scalar field previously PDE/Layer are moved into Globals/Fields
- **MorpheusML 2.0**
    - Restructured MorpheusML markup language
    - Allows better future integration with other formats such as SBML
    - Fixboard can partially convert Morpheus 1.x models into new format.

GUI

- **Documentation**
    - Doxygen-generated in-app docs appear in HTML (using QtWebKit)
    - Docs are both context-sensitive and can be searched (using QtHelp framework)
    - Docs can include latex-based maths (displayed using MathJax)



VolumeConstraint

Plugins » Cell Shape PluginsModelStructure |

Continuous Process Plugins » CPM » |

CPM Hamiltonian Plugins

Penalizes deviations from target cell area (2D) or volume (3D) More...

Penalizes deviations from target cell area (2D) or volume (3D)

The volume constraint models the conservation of area (2D) or volume (3D) of cells.

It puts a quadratic energy penalty to deviations of the area (2D) or volume (3D) of a cell $v_{\sigma,t}$ from a given target (resting) area or volume V_{target}, specified in units of lattice sites. The strength of this penalty is modulated by a Langrangian multiplier λ_V that determines the cell's incompressibility.

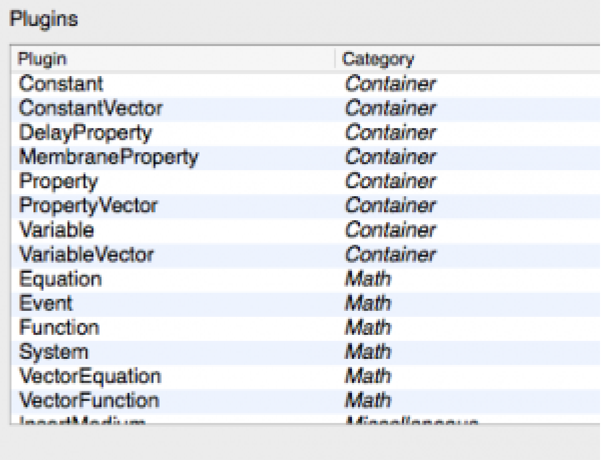
The Hamiltonian is given by:

$$E_{Volume} = \sum_{\sigma} \lambda_V \cdot (v_{\sigma,t} - V_{target})^2$$

For each proposed copy attempt $\mathbf{x} \rightarrow \mathbf{x}_{\mathbf{neighbor}}$, the change in effective energy is computed as:

The new in-app documentation is context-sensitive, browsable, features Latex equations and XML examples.

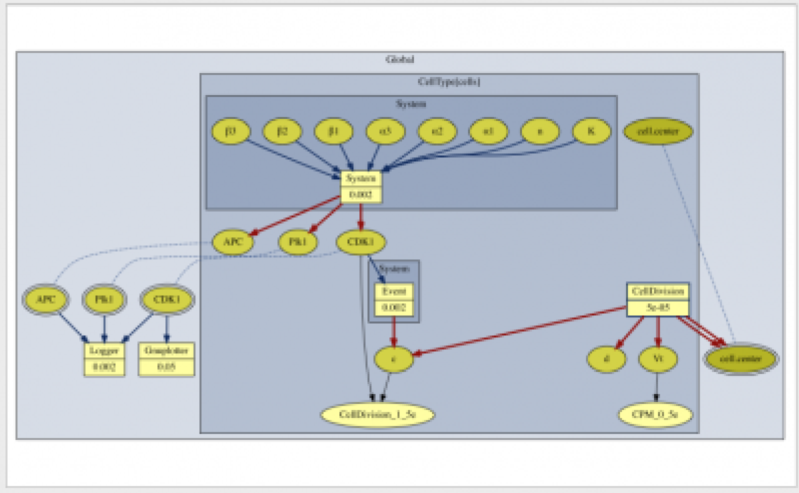
- **Panel with addable plugins**
    - New context-sensitive overview of addable plugins shows what plugins can be added


Plugins

Plugin	Category
Constant	*Container*
ConstantVector	*Container*
DelayProperty	*Container*
MembraneProperty	*Container*
Property	*Container*
PropertyVector	*Container*
Variable	*Container*
VariableVector	*Container*
Equation	*Math*
Event	*Math*
Function	*Math*
System	*Math*
VectorEquation	*Math*
VectorFunction	*Math*
InsertMedium	*Miscellaneous*

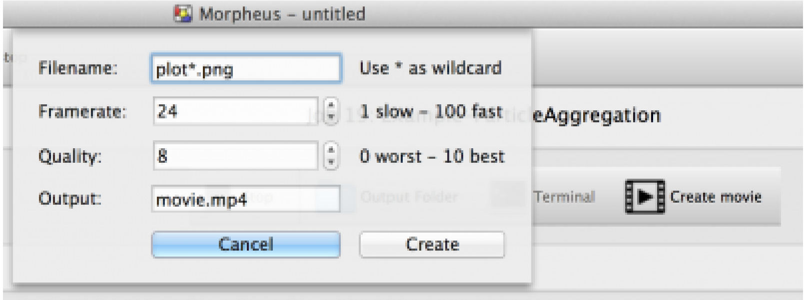
Plugins can now be added using a new panel that shows what can be added at the selected position.

- **Generate symbol graph**
    - Shows interdependencies between symbols and scopes



Symbol graphs show how Morpheus interprets the model. This can be used for model testing and tracing bugs.

- **Generate movies**
    - Images can be generatedfrom image sequences within GUI
    - Also generates multiple movies from `ParamSweep`
    - Requires `ffmpeg` or `avconv` (download from here)



Filename: plot*.png Use * as wildcard

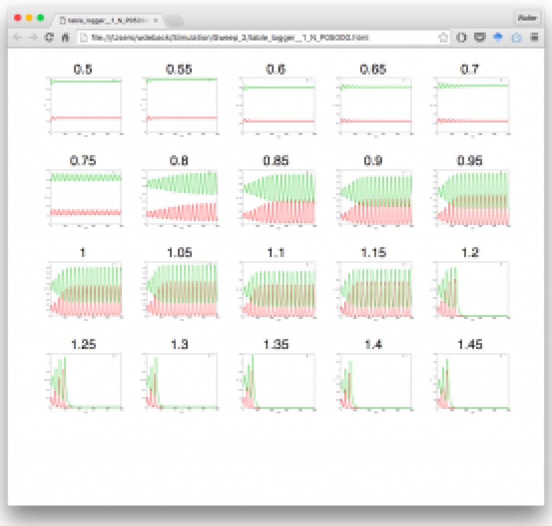
Framerate: 24 1 slow – 100 fast

Quality: 8 0 worst – 10 best

Output: movie.mp4

Create movies from images.

- **ImageTable**
    - Generate visual tables of `ParamSweep` results including images or movies (viewable in HTML5-enabled web browsers)



The new ImageTable feature gives a quick overview of the results of a parameter sweep from a web browser.

Version 1.2.1

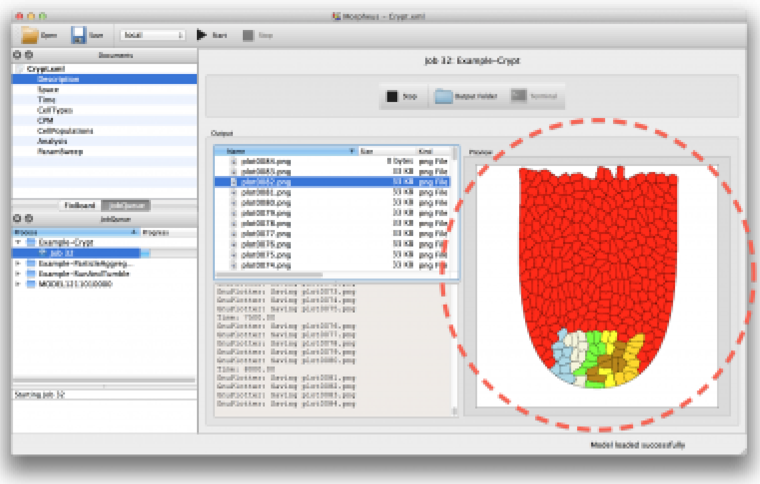
Changes

- SBML support is built-in for Ubuntu 14.04+ as libSBML was removed from repositories.
- Fixed rare GUI crashes on Mac OSX when moving model elements.

Version 1.2

The changes in this version focus on improving modeling flexibility, but also include several improvements in automatic scheduling as well as a new preview panel in the user interface. Download version 1.2 here.

Major changes



New panel to preview images and text files in result browser.

- New **preview panel** in GUI in the result browser for images and plain text files.
- New `ChildID` option for `Proliferation`. This supports the representation of **asymmetric cell divisions** by providing a symbolic handle to treat daughter cells differently after cell division.
    - See new example Crypt.xml
- New `ChangeCellType` plugin to conditionally **change the cell type** of a cell. By default, cell properties are copied, but may be overridden using `Triggers`.
    - See new example Crypt.xml
- New `VectorRule` to manipulate `PropertyVector`, either in x,y,z or phi,theta,radius (spherical coordinates). This facilitates **vector computations** to e.g. control directed cell motility.
    - See new example RunAndTumble.xml
- New `FlipCells` plugin to represent **cell motility** in CA-like models. This plugin enables the conditional switching of positions with a random neighboring site.
    - See new example ParticleAggregation.xml
- New `PopulationReporter` to collect **population-level statistics** on cell `Properties`. This allows individual cell behavior to depend on population-level properties, or to collect population statistics to log and plot.
    - See new examples ParticleAggregation.xml and Crypt.xml
- New `StopCondition` to **terminate a simulation** upon a `Condition` before reaching `StopTime`. This can significantly improve efficiency in e.g. parameter explorations or sensitivity analysis.
    - See new example ParticleAggregation.xml
- New `trigger` option for `Event`. This provides a choice to **trigger events** (1) whenever the condition changes from false to true (as in SBML) or (2) whenever the condition is true.
- Several fixes and many improvements to **automatic scheduling**.
    - Analysis tools are now scheduled according to input symbols if `interval=0`

Minor changes

- New `InitCA` plugin to populate CA-like models.
    - See examples ParticleAggregation.xml, LateralSignaling.xml, GameOfLife.xml and FrenchFlag.xml
- Fix `InitRectangle` on `hexagonal` lattices.
- Support for α-synchronous simulation.
- Fixed issue with `Delay` properties.
- `HistogramLogger` now supports logarithmic binning and plotting

Acknowledgments

Many of these new features are the result of requests from users. In particular, we thank *Maria*

Herberg (Medical Faculty, TU Dresden), *Angelo Torelli* (Mannheim University of Applied Sciences) and *Ovidiu Pârvu* (Brunel University) for useful suggestions.

Version 1.1.1

Minor changes

- Fix issue in Windows with SBML import feature.
- Fix issue in GUI with displaying newly created elements.
- Updated version for math expression parser (muparser).

Version 1.1

Major changes

- Added **SBML Import** feature that converts SBML-based models into equivalent intracellular models of differential equations in Morpheus description language. An example model is added.

- Added `DelayProperty` that returns assigned values after a specified time lag which enables the simulation of **delay differential equations**. An example model is added.

- A distinction is made between **`Equations` and `Rules`** whereby the latter are equations within a `System`. Unlike `Equations`, `Rules` may have recurrency and be part algebraic loops.

- **`Events`** are now triggered whenever the condition change from false to true, for SBML compatibility.

- Sequential processes are now **scheduled** only as often as their input symbol can change. This avoids redundant processing and optimizes computational performance.

Minor changes

- Fixed issues with file dialogs and output folder button under Windows.
- Fixed libfreetype problems with Gnuplot.app under Mac OSX.
- Reduced verbosity in standard output (`model.xml.out`).
- Moved Examples menu to toolbar menu.
- Various minor fixes.
