PSim Installation and Release Notes

Release 1.7.0

The Polyswift Team

May 08, 2018

Contents

1 Introduction

Welcome to the PSim, the self-consistent field theory code for simulating mesoscale structure of complex block copolymer mixtures. PSim consists of two major components:

- PolySwift++ computational engine
- PSimComposer graphical user interface

PSimComposer provides an interface that allows you to edit and validate your simulation input files, run PolySwift++ simulations, and visualize results using the VisIt-based Visualization pane embedded within PSimComposer. You can also edit PolySwift++ input files in the text editor of your choice, perform calculations with the easy-to-run, command-line-driven PolySwift++ executable, and then run the visualization tool of your choice.

A serial version of PolySwift++ is available for running on single processor workstations. A parallel version of PolySwift++ is provided for multi-core systems that support the Message Passing Interface (MPI).

More information about PSim can be found at the PSim Product Website. Send questions about installing or running PSim to Tech-X Customer Support at support@txcorp.com. For technical questions about the physics, users may wish to email the PolySwift++ discussion list (polyswift-users List). Extensive assistance in the use of PolySwift++ engine or PSim simulations in general is available from Tech-X Professional Services. Please contact Tech-X directly for sales, consulting, and other questions at sales@txcorp.com.

2 Installation

For all operating systems, installation places

- PolySwift++ computational engine (polyswiftser and polyswift)
- PSimComposer, the graphical user interface to PolySwift++
- PSim documentation (.pdf files)

on your computer.

2.1 PSim Installation Instructions

Windows PSim Software Installation

The PSim distribution package for Windows, both 32-bit and 64-bit, is a self-extracting executable. Invoke the installer by double clicking on it. The default installation path is:

For 32-bit application on 32-bit Windows:

```
C:\Program Files\Tech-X\PSim 1.0
```

For 32-bit application on 64-bit Windows:

```
C:\Program Files (x86)\Tech-X\PSim 1.0
```

For 64-bit application on 64-bit Windows:

```
C:\Program Files\Tech-X (Win64)\PSim 1.0
```

To open the PSim software, go to the Start Menu, click on the Tech-X folder, click on PSim 1.0, then click on PSimComposer. See *Dialogs for Windows installation.*.

Mac OS X PSim Software Installation

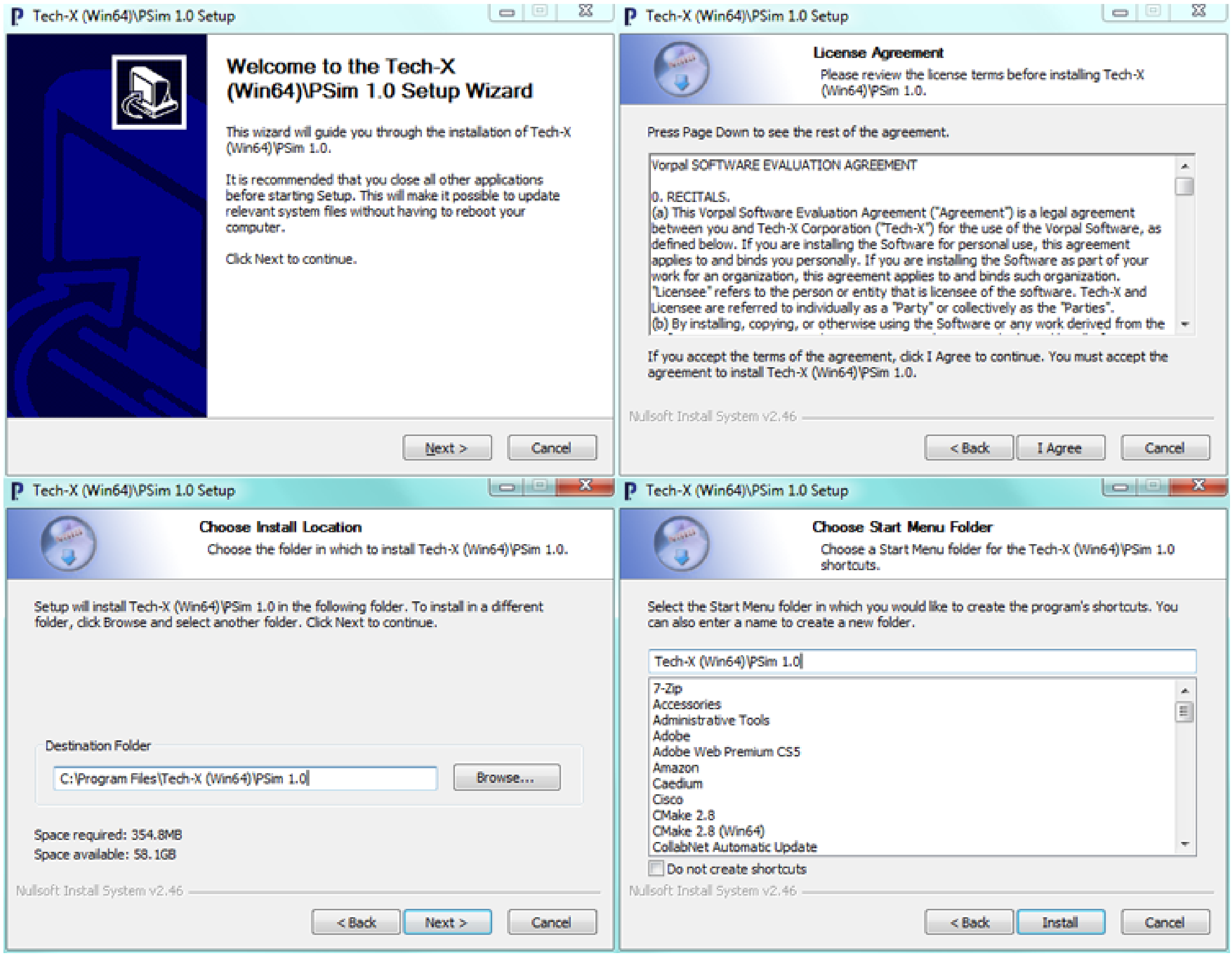
MacOS El Capitan or previous Users:

The PSim distribution package for Mac OS X is a .dmg installer. Invoke the installer by double clicking on it. Drag the PSim folder into your Applications folder (visible in the installer window). From the Application folder, double click on the PSimComposer icon in the PSim folder. See *Dialog for the Mac installer.*. This default installation path is:

```
/Applications/PSim-'version'
```

MacOS Sierra or later Users:

For MacOS Sierra or later, special steps must be taken because of the newer security features.



Welcome to the Tech-X (Win64)\PSim 1.0 Setup Wizard

This wizard will guide you through the installation of Tech-X (Win64)\PSim 1.0.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

License Agreement

Please review the license terms before installing Tech-X (Win64)\PSim 1.0.

Press Page Down to see the rest of the agreement.

Vorpal SOFTWARE EVALUATION AGREEMENT

0. RECITALS.
(a) This Vorpal Software Evaluation Agreement ("Agreement") is a legal agreement between you and Tech-X Corporation ("Tech-X") for the use of the Vorpal Software, as defined below. If you are installing the Software for personal use, this agreement applies to and binds you personally. If you are installing the Software as part of your work for an organization, this agreement applies to and binds such organization. "Licensee" refers to the person or entity that is licensee of the software. Tech-X and Licensee are referred to individually as a "Party" or collectively as the "Parties".
(b) By installing, copying, or otherwise using the Software or any work derived from the

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Tech-X (Win64)\PSim 1.0.

Choose Install Location

Choose the folder in which to install Tech-X (Win64)\PSim 1.0.

Setup will install Tech-X (Win64)\PSim 1.0 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder

C:\Program Files\Tech-X (Win64)\PSim 1.0

Space required: 354.8MB
Space available: 58.1GB

Choose Start Menu Folder

Choose a Start Menu folder for the Tech-X (Win64)\PSim 1.0 shortcuts.

Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

Tech-X (Win64)\PSim 1.0

7-Zip
Accessories
Administrative Tools
Adobe
Adobe Web Premium CS5
Amazon
Caedium
Cisco
CMake 2.8
CMake 2.8 (Win64)
CollabNet Automatic Update

□ Do not create shortcuts

Fig. 1: Dialogs for Windows installation.

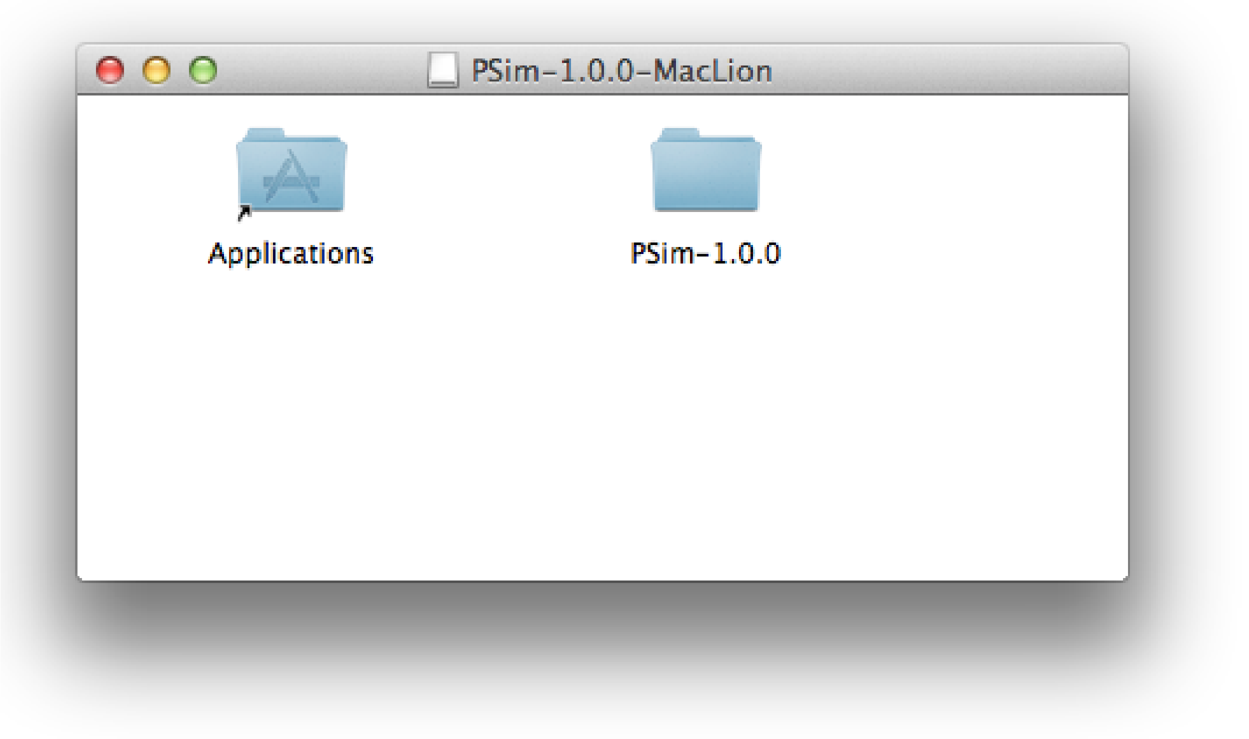


Fig. 2: Dialog for the Mac installer.

3 First Method

- Make sure you have deleted ALL instances of the version of PSim that you are about to install: from the Applications folder, from Documents/txcorp, and from your Desktop. Empty these folders from Trash as well.
- Unmount any PSim installer disk image. DO NOT CLICK ON IT!
- In a terminal window, run:

```
sudo spctl --master-disable
```

- Upon doing so, in System Preferences… Security & Privacy, you should see that the radio button for *Allow apps downloaded from:* is set to *Anywhere*.
- Open the disk image in your download folder.
- A window should open that shows your Application folder and a PSim-1.X folder.
- Drag the PSim-1.X folder into the Applications folder, then run PSimComposer from inside PSim-1.X folder that is inside the Applications folder.
- Open PSim-1.X from the Applications folder. If you get the message "PSimComposer can't be opened because it is from an unidentified developer" go to Apple menu > System Preferences > Security & Privacy > General. In the lower part of the GUI select "App Store and identified developers" and a message will appear below: "PSimComposer was blocked from opening because it is not from an identified developer". Click on "Open Anyway".
- Now you can open PSim-1.X from the Applications folder and add a license.
- AFTER adding the license, in a terminal, you may run

```
sudo spctl --master-enable
```

- To restore default security preferences.

4 Second Method

- Make sure you have deleted ALL instances of the version of PSim that you are about to install, including desktop icons.
- Unmount any PSim installer disk image.
- Double click on the PSim installer disk image to mount it.
- A window should open up that shows your Application folder and a PSim-1.X folder.
- Drag the PSim-1.X folder into the Applications folder.
- In a terminal window, run:

```
sudo spctl --add /Applications/PSim-1.X/PSimComposer.app
```

- Run PSimComposer from inside that PSim folder.
- Open PSim-1.X from the Applications folder. If you get the message "PSimComposer can't be opened because it is from an unidentified developer" go to Apple menu > System Preferences > Security & Privacy > General. In the lower part of the GUI select "App Store and identified developers" and a message will appear below: "PSimComposer was blocked from opening because it is not from an identified developer". Click on "Open Anyway".
- You will now be able to add a PSim license.

If you open PSim and get a message saying "License File Not Found", make sure that the path for your license starts with:

```
/Applications/PSim-1.X/...
```

If it does not, please provide full information to support@txcorp.com.

The PSim distribution package for Linux is a gzipped tarball. Unpack the gzipped tarball into the directory in which you wish to install PSim. A typical location would be

```
/usr/local/PSim
```

The unzip and untar command is

```
$ tar -zxvf PSim-'version'-Linux64.tar.gz
```

To run PSimComposer, execute the command

```
$ cd PSim-'version'-Linux64
$ ./PSimComposer.sh
```

from within the installation directory.

4.1 Included Software

Upon completing the installation process, the following software will be installed on your computer.

PSimComposer

PSimComposer is a graphical user interface for

- Creating and editing PolySwift++ input files
- Executing PolySwift++ engine
- Analyzing PolySwift++ generated data
- Visualizing PolySwift++ generated data
- Viewing the documentation.

The PSimComposer editor and validator have built-in functions and graphical components that help you to create well-formed input files. Example input files, ranging in complexity from beginning to advanced, are included with PSimComposer. New PolySwift++ users can use these examples as templates.

The PSimComposer Run pane invokes PolySwift++ with user definable settings for number of steps, number of data dumps, and restart file, if any. It also allows selection of serial versus parallel PolySwift++.

PSimComposer now allows selection of analysis programs, either supplied with PolySwift++ or user written.

The visualization in PSimComposer is provided by the graphical analysis tool VisIt. VisIt is embedded within PSimComposer. Data generated by PolySwift++ or by analysis programs automatically appears in the Visualization pane.

All documentation can be seen from within PSimComposer, fully cross-referenced.

PolySwift++

PolySwift++, the computational engine, runs both as a serial and parallel code for multi-processor / multi-core systems that support MPI. PolySwift++ now comes in the specialized PSim packages. PolySwift++ is embedded within PSimComposer.

Python

Python is an open-source, interpreted scripting language managed by the Python Software Foundation. For more information about Python, see: http://www.python.org/

PSim uses Python to process input files, allowing users to set up simulations with math functions, variable substitutions, and macros.

PolySwift++ uses its own embedded version of the Python interpreter to pre-process input files and execute any Python code in an input file. Python is also included within PSimComposer.

MPI

The Message Passing Interface (MPI) is an application programming interface (API) for communicating between processes that execute in parallel. There are many implementations. The Linux and Mac versions come with the OpenMPI (http://www.open-mpi.org/) implementation of MPI. The Windows versions come with the Microsoft MPI implementation (http://www.microsoft.com/). The appropriate MPI implemention is embedded within PSimComposer.

4.2 PSim Documentation

All documentation is accessible from within the PSimComposer interface, as well as online at the PSim Support Website.

5 Release Notes

The release notes describes new features for the PSim computational engine (Polyswift++) and the PSimComposer graphical user interface.

5.1 New and Updated Features

PSim Computational Engine (PolySwift++)

PSimComposer GUI Features

The PSimComposer graphical user interface (GUI) allows the user to edit and validate simulation input files, run simulations in either serial or parallel (thereby utilizing multiple cores or even computational nodes that do not share memory), and visualize results.