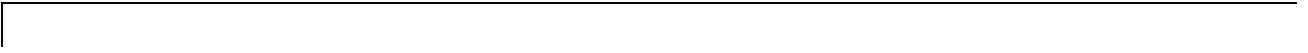# System Identification Toolbox Release Notes

Note The System Identification Toolbox 5.0.2 does not include any significant changes for Release 13.

If you are upgrading from a release earlier than Release 12, you should see "System Identification Toolbox 5.0 Release Notes" on page 1-1.

Printing the Release Notes

If you would like to print the Release Notes, you can link to a PDF version.

Contents

System Identification Toolbox 5.0 Release Notes

1

1

System Identification Toolbox 5.0 Release Notes

New Features

This section introduces the new features and enhancements added in the System Identification Toolbox 5.0 since the System Identification Toolbox 4.0.5 (Release 11.0).

Object-Based Design

Based on MATLAB object technology, the System Identification Toolbox 5.0 provides functions for creating objects directly associated with your models and data. Some quick examples illustrating this feature are

```
z = iddata(y,u,Ts);
sys = pem(z);
```

The new object-based syntax makes it much easier to perform analysis activities beyond what the System Identification Toolbox visual interfaces support. The use of an object-based design by the System Identification Toolbox 5.0 makes it much easier to work with Control System Toolbox objects seamlessly, including converting back and forth between the two toolbox's objects and applying the relevant analysis tools to objects from both toolboxes.

For an overview of the features included in this new object-based approach, type

```
help idhelp
```

You do not need to rewrite any code you wrote using an earlier version of the System Identification Toolbox; the earlier command-line syntax is still supported in the System Identification Toolbox 5.0.

Advanced Feature Enhancements

The System Identification Toolbox 5.0 includes several enhancements to some of the toolbox's more advanced features:

• Free parameterization for state-space models is now supported. For example, you can simply use

```
m = pem(z,4)
```

to obtain a fourth order state-space model with a well-conditioned parameterization.

- You can now add initial filter conditions. This yields much better performance for slow dynamics. See the `'InitialState'` property of `idmodel` objects for further information.

- You can now use the `SearchDirection` and `Advanced` properties of `idmodel` objects to access several variants of iterative search algorithms. For more information, type

  ```
  idprops algorithm
  ```

- You can now focus the model approximation inherent in system identification to various frequency regions, by using the `Focus` property. The values for the `Focus` property include `'Prediction'`, `'Simulation'`, or any idmodel or LTI object that uses the frequency weighting of that system.

Upgrading from an Earlier Release

This section describes the upgrade issues involved in moving from the System Identification Toolbox 4.0.5 to the System Identification Toolbox 5.0.

Theta Models No Longer Supported

Theta models (matrices) are no longer supported in the System Identification Toolbox 5.0. Existing code that uses functions such as th2par and th2ss to access the theta model data will continue to work in the System Identification Toolbox 5.0. However, if you have code that directly indexes into the theta matrix (e.g., th(1,3)), that code will no longer work.