

PySCeS 0.4.x Input File Guide

<http://pysces.sourceforge.net>

Brett G. Olivier, Johann M. Rohwer and Jan-Hendrik S. Hofmeyr

Triple-J Group for Molecular Cell Physiology
Biochemistry Department, Stellenbosch University

Contents

| | | |
|----------|---|----------|
| 1 | Introduction | 3 |
| 2 | Basic format | 4 |
| 3 | Detailed format | 4 |
| 3.1 | Fixed species | 4 |
| 3.2 | Reaction stoichiometry and rate equations | 4 |
| 3.3 | Initialization | 6 |
| 4 | An example PySCeS input file | 8 |
| 5 | Formal PySCeS input file syntax | 9 |

© The PySCeS development team – Stellenbosch (September 20, 2007)

1 Introduction

PySCeS: the Python Simulator for Cellular Systems is an extendable toolkit for the analysis and investigation of cellular systems. It is available for download from: <http://pysces.sourceforge.net>

PySCeS uses an ASCII text ‘input file’ to describe a cellular system in terms of it’s stoichiometry, kinetics and parameters. Input files may have any filename with the single restriction that, for cross platform compatibility, they must end with the extension `.psc`. The PySCeS Model Description Format (MDF) is based on the input file format defined by Jannie Hofmeyr and his original language specification is included as part of this guide.

PySCeS is distributed under the GNU General Public Licence (GPL) and is made freely available as Open Source software.

We hope that you will enjoy using our software. If, however, you find any unexpected features (bugs) or have any suggestions on how we can improve PySCeS please let us know.

Brett G. Olivier
Johann M. Rohwer
Jan-Hendrik S. Hofmeyr
Stellenbosch (September 20, 2007)
<http://www.jjj.sun.ac.za>

2 Basic format

In order to describe a basic PySCeS model, five distinct types of information are required, the:

- fixed or boundary species.
- reaction network stoichiometry.
- rate equations for for each reaction step.
- system parameter and external species values.
- initial values of the variable species.

Once you have this information it can be organised and written as a formatted PySCeS input file.

3 Detailed format

There is no specific header section but model details can be placed at the start of an input file using comments. Generally, comments are inserted into the input file using a Python single line comment for example:

```
# Title Sucrose metabolism in sugar cane  
# Johann M. Rohwer, Biochem. J. (2001) 358, 437-445
```

This type of comment can also be used anywhere in the input file.

3.1 Fixed species

First boundary species need to defined and set at a specific value. Boundary species, also called fixed or external species, are usually necessary for an open system to achieve a steady state. In a PySCeS input file fixed species need to be declared using the **FIX** keyword. The declaration line must begin with the keyword **FIX:** and followed by a space separated list of species names.

```
FIX: Fru_ex Glc_ex ATP ADP UDP phos glycolysis Suc_vac
```

3.2 Reaction stoichiometry and rate equations

The reaction stoichiometry and rate equations are grouped together per reaction step. Each reaction step in the system is defined as having a name (identifier), a stoichiometry (substrates are converted to products) and rate equation (the catalytic activity, described in terms of species and parameters). All reaction definitions should be separated by an empty line.

Reaction name

Each reaction is given a unique name, for a detailed description of the allowed name format see the formal model description in Section 5 of this guide, the name is declared and followed by a :

R1:

R2:

Reaction stoichiometry

On the line following the reaction name the reaction stoichiometry is defined with reaction substrates on the left and products on the right hand side of an identifier which describes the reaction as either reversible (=) or irreversible (>). Each reagent's stoichiometric coefficient (as an int or float) can be included in brackets () immediately preceding the reagent name, these can be omitted if the coefficient is one.

```
{2}Hex_P = Suc6P + UDP    # reversible reaction
```

```
Fru_ex > Fru              # irreversible reaction
```

Reaction rate equation

Rate equations should be written as valid Python expression and may fall across more than one line. Standard Python operators (+ - * /) are allowed, including the Python power operator (**), for example 2^4 is written as `2**4`. There is no shorthand for multiplication in Python so $-2(a+b)^h$ would be written as `-2*(a+b)**h` and the normal Python operator precedence applies as summarised in Table 1. A number of functions

| | |
|--------|--------------------------|
| +, - | Addition, subtraction |
| *, / | Multiplication, division |
| +x, -x | Positive, negative |
| ** | Exponentiation |

Table 1: Operator precedence increases from top to bottom and left to right, adapted from the *Python Reference Manual*.

can also be included in the rate equations including:

- log, log, log10

- pow, abs, fabs, factorial
- cos, cosh, sin, sinh, tan, tanh
- arccos, arccosh, arcsin, arcsinh, arctan, arctanh

An example of a complete reaction step including the name, stoichiometry and rate equation:

R5:

```
Fru + ATP = Hex_P + ADP
Fru/Ki5_Fru)*(Fru/Km5_Fru)*(ATP/Km5_ATP)/(1 +
Vmax5/(1 + Fru/Ki5_Fru)*(Fru/Km5_Fru)*(ATP/Km5_ATP)/(1 +
Fru/Km5_Fru + ATP/Km5_ATP + Fru*ATP/(Km5_Fru*Km5_ATP) + ADP/Ki5_ADP)
```

3.3 Initialization

A model property is declared and initialized using the form:

```
property = value
```

Initializations can be written in any order but should neither use shorthand floating point (1.) nor shorthand exponential (1.e-3) syntax. Instead, full exponential (1.0e-3), decimal (0.001) and floating point syntax (1.0) is required. It is generally a good idea to end the input file with at least a single blank line to avoid potential cross platform errors.

Parameters, external species

Once the reactions are defined the system's external species and parameters should now be initialised. Although generally speaking these parameters should also be present in your rate equations, they are not required to be. If such parameter initialisations are detected a harmless warning is generated when the model is parsed. If, on the other hand, an uninitialised parameter is detected a warning is generated and the model will not function properly.

```
# InitExt
X0 = 10.0
X4 = 1.0

# InitPar
Vf1 = 10.0
Ks1 = 1.0
```

Variable species initial values

Although the variable species concentrations are determined by the parameters of the system, it is important to give them an initial value. These initial values are used for calculating total moiety concentrations (if present) and initial values for certain analysis methods. If you want the species pools to start empty it is not recommended to initialise these values to zero (in order to prevent potential zero division errors) but rather to a small value (e.g. 10^{-6}).

```
# InitVar  
S1 = 1.0e-03  
S2 = 2.0e+02
```

4 An example PySCeS input file

```
# PySCeS input file
# Simple linear pathway (2004)
```

```
FIX: x0 x3
```

```
R1:
    x0 = s0
    k1*x0 - k2*s0
```

```
R2:
    s0 = s1
    k3*s0 - k4*s1
```

```
R3:
    s1 = s2
    k5*s1 - k6*s2
```

```
R4:
    s2 = x3
    k7*s2 - k8*x3
```

```
# InitExt
x0 = 10.0
x3 = 1.0
```

```
# InitPar
k1 = 10.0
k2 = 1.0
k3 = 5.0
k4 = 1.0
k5 = 3.0
k6 = 1.0
k7 = 2.0
k8 = 1.0
```

```
# InitVar
s0 = 1.0
s1 = 1.0
s2 = 1.0
```


5 Formal PySCeS input file syntax

Extract from `lexparse.py` (c) Copyright Jannie Hofmeyr, 2002

Lexical analyser and parser for a string or file containing the set of reaction equations defining a model. There are three possible inputs:

1. DECLARATION OF FIXED REAGENTS (optional)

FIX: Reagent_1 Reagent_2 ... Reagent_n (list can be empty)

Spaces between reagents; no other separators allowed.

2. REACTION LINE (required)

ReactionID: {StoichCoef}Reagent + ... + {StoichCoef}Reagent
 = | > {StoichCoef}Reagent + ... + {StoichCoef}Reagent
 Rate equation (optional)

ReactionID: name given to the reaction, e.g., R1 or ATPase
 (must begin with a letter, otherwise any sequence of [a-zA-Z0-9_],
 no spaces allowed)

StoichCoef: stoichiometric coefficient
 (integer or decimal number) enclosed in curly brackets

Reagent: name of a reactant or product
 (must begin with a letter, otherwise any sequence of [a-zA-Z0-9_],
 no spaces allowed)

= denotes reversible reaction
> denotes irreversible reaction

Rate equation: valid Python expression (required for timecourse and steady
state,
 but not for elementary modes)

NOTES:

- If StoichCoef = 1, it can be omitted
- The rate equation, if present, must follow the reaction, but need not be on the same line

3. PARAMETER INITIALISATION (optional)

Parameter = Value

Can be on same or separate lines, no separators needed.