Toymodel & Full FBA model

https://github.com/dagwa/wholecell-metabolism.git

Toymodel

https://github.com/dagwa/wholecell-metabolism/tree/master/mkoenig/python/metabolism/toymodel

toy_comp.xml

- toy_ode_model.xml
- toy_ode_update.xml
- toy_ode_bounds.xml

toy_fba.xml

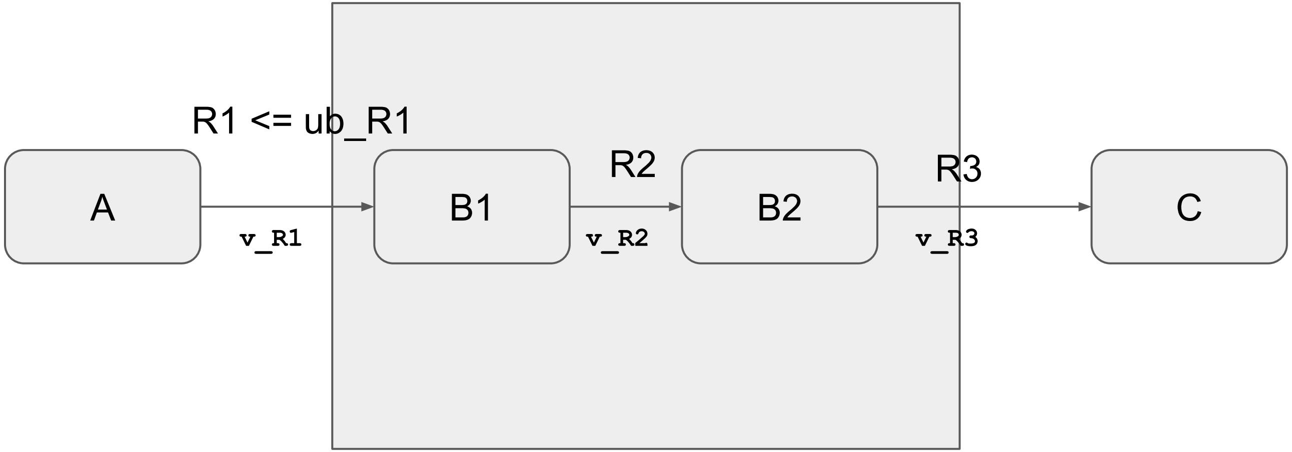
Full FBA model

https://github.com/dagwa/wholecell-metabolism/tree/master/mkoenig/results

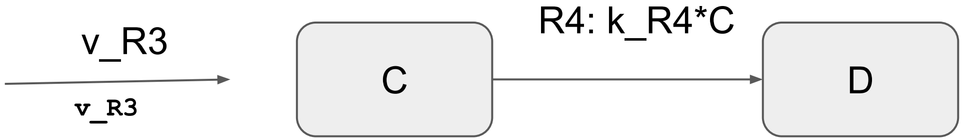
Metabolism_matrices_08_L3V1.xml

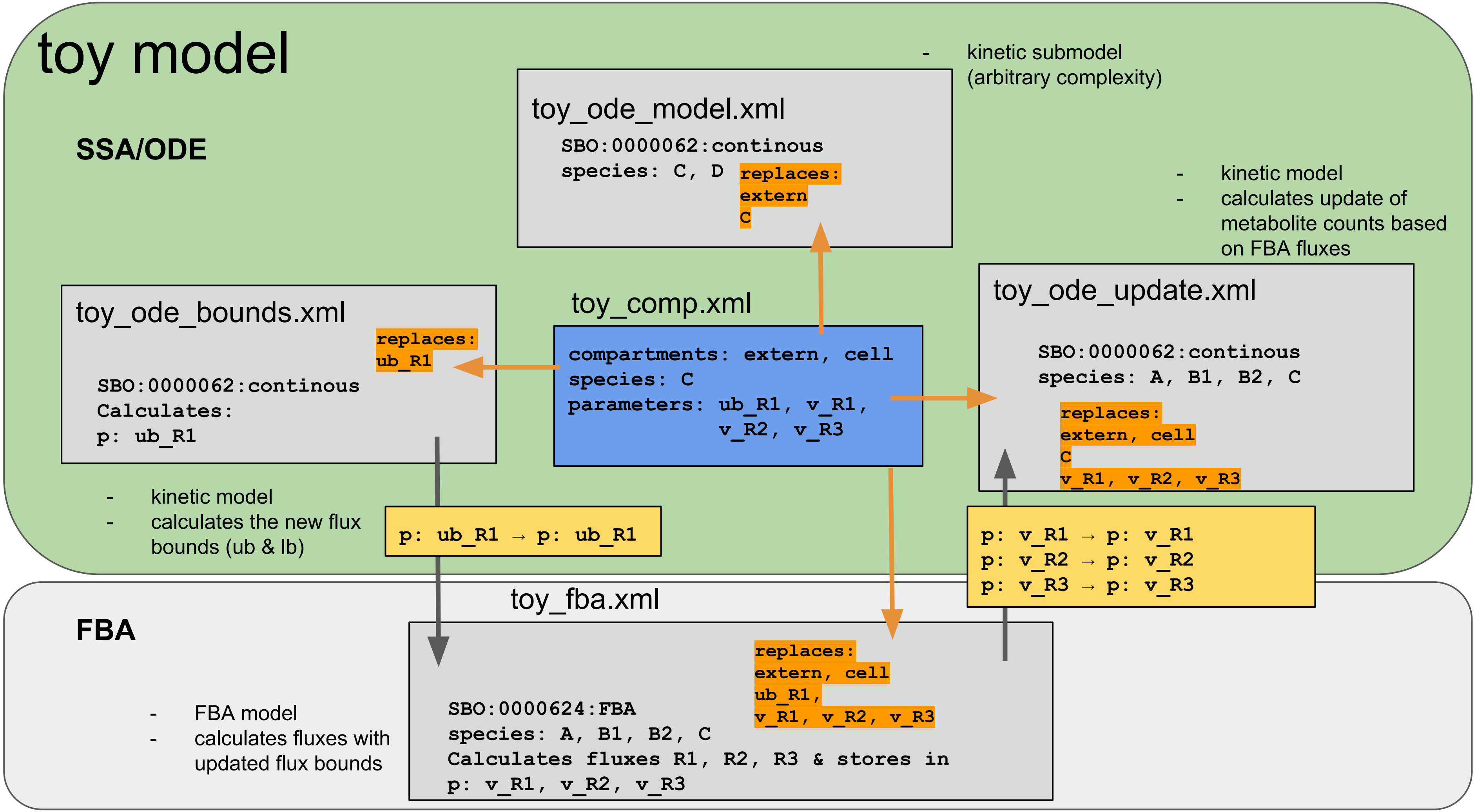
Metabolism_matrices_annotated_08_L3V1.xml

FBA Submodel



Kinetic Submodel





toy model

SSA/ODE

- kinetic submodel (arbitrary complexity)

toy_ode_model.xml

```
SBO:0000062:continous
species: C, D
replaces:
extern
C
```

toy_ode_bounds.xml

```
replaces:
ub_R1
SBO:0000062:continous
Calculates:
p: ub_R1
```

- kinetic model
- calculates the new flux bounds (ub & lb)

toy_comp.xml

```
compartments: extern, cell
species: C
parameters: ub_R1, v_R1,
            v_R2, v_R3
```

- kinetic model
- calculates update of metabolite counts based on FBA fluxes

toy_ode_update.xml

```
SBO:0000062:continous
species: A, B1, B2, C
replaces:
extern, cell
C
v_R1, v_R2, v_R3
```

```
p: ub_R1 → p: ub_R1
```

```
p: v_R1 → p: v_R1
p: v_R2 → p: v_R2
p: v_R3 → p: v_R3
```

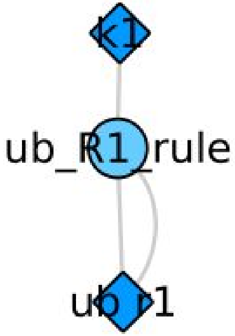
FBA

toy_fba.xml

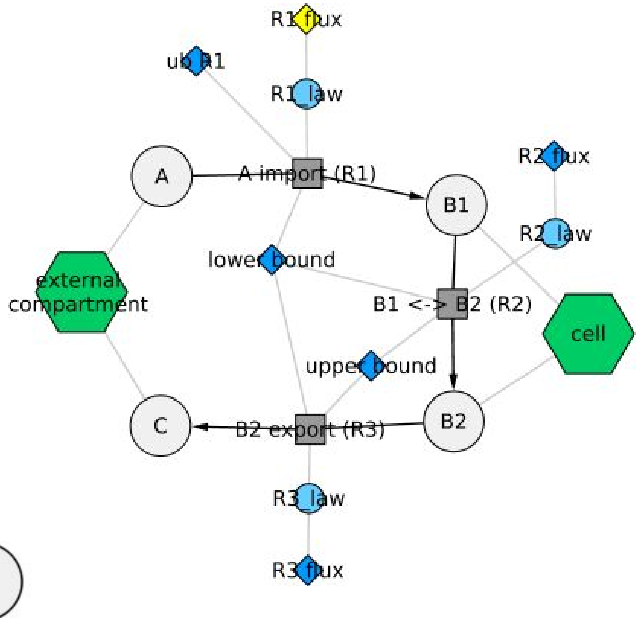
```
replaces:
extern, cell
ub_R1,
v_R1, v_R2, v_R3
SBO:0000624:FBA
species: A, B1, B2, C
Calculates fluxes R1, R2, R3 & stores in
p: v_R1, v_R2, v_R3
```

- FBA model
- calculates fluxes with updated flux bounds

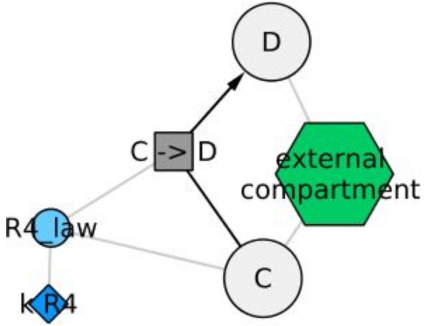
toy_bounds.xml



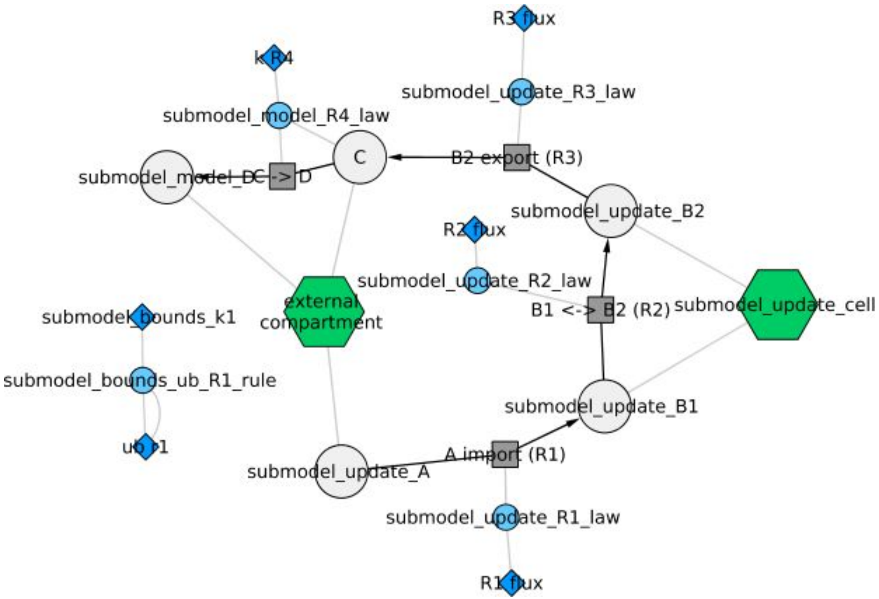
toy_fba.xml & toy_ode_update.xml

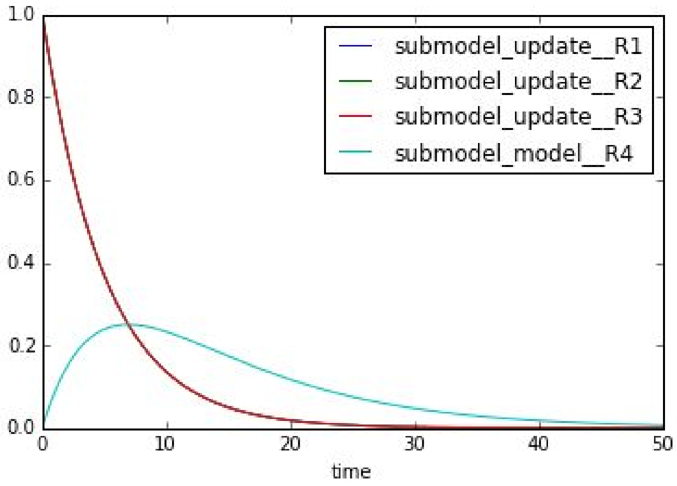


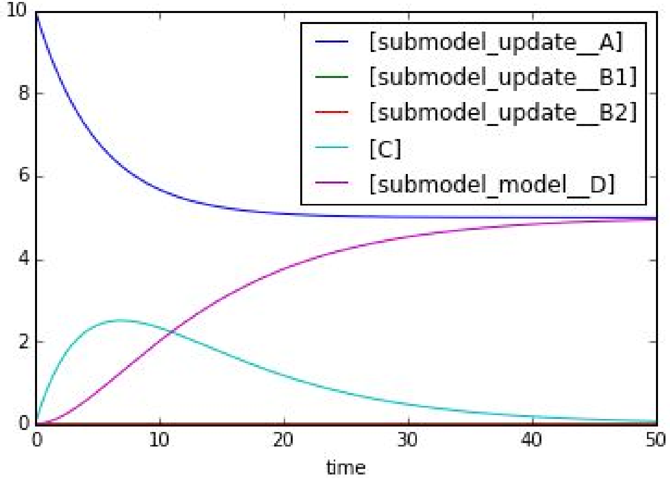
toy_model.xml



toy_comp.xml (flattened)







Version: 61004733ca7

Solution for
simulate(tend=50.0, step_size=0.01)

Initial Values

Parameters

ub_R1 = 1.0
ub = 1000
lb = 0

Species

A   = 10.0
B1 =   0.0
B2 =   0.0
C   =   0.0
D   =   0.0

Compartments

ext = 1.0
cell = 1.0

FluxBounds

lb <= R1 <= r1
lb <= R2 <= ub
lb <= R3 <= ub

Objective

maximize : 1.0*R3