
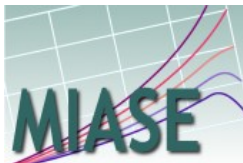





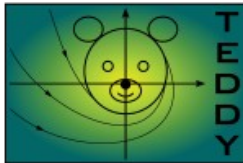


# KiSAO

Kinetic Simulation  
Algorithm Ontology

	Models	Simulation	Results
Minimal requirements			
Data-models	 		SBRML
Ontologies			

# Simulation approach

```
<listOfSimulations>  
  <uniformTimeCourse id="simulation1"  
    initialTime="0"  
    outputStartTime="0"  
    outputEndTime="140"  
    numberOfPoints="1000">  
    <algorithm kisaoID="KiSAO:0000030"/>  
  </uniformTimeCourse>  
</listOfSimulations>
```

# Simulation approach

```
<listOfSimulations>  
  <uniformTimeCourse id="simulation1"  
    initialTime="0"  
    outputStartTime="0"  
    outputEndTime="140"  
    numberOfPoints="1000">  
    <algorithm kisaoID="KISA0:0000030"/>  
  </uniformTimeCourse>  
</listOfSimulations>
```

The screenshot displays the KISA0:0000071 interface, which is used for defining and managing simulation algorithms. The main window is titled "KISA0:0000071" and shows the "Definition" tab for the "Livermore solver for ordinary differential equations".

**Definition Tab:**

- Text:** "LSODE solves explicitly given ODE systems. [and] [...] is based on the GEAR and GEARB packages. It solves ODE systems given explicitly as  $dy/dt = f(t, y)$ ". Hindmarsh AC. LSODE and LSODI, two new initial value ordinary differential equation solvers. SIGNUM Newsletter, Volume 15 (4), pages 10-11 (1980). Radhakrishnan K, Hindmarsh AC. Description and Use of LSODE, the Livermore Solver for Ordinary Differential Equations. Lawrence Livermore National Laboratory Report, Vol. UCRL-ID-113855 (1993).
- Dbxrefs:** dk:16NOV2007, doi:1218052.12180
- Synonyms:** LSODE. Select a synonym from the list to edit it, or press add to create a new synonym.

**DAG Viewer:**

- Classes
  - kinetic simulation algorithm
    - is\_a algorithm using non-spatial description
      - is\_a Livermore solver
        - is\_a Livermore solver for ordinary differential equations
    - is\_a algorithm using continuous variables
      - is\_a Livermore solver
        - is\_a Livermore solver for ordinary differential equations
    - is\_a algorithm using deterministic rules
      - is\_a Livermore solver
        - is\_a Livermore solver for ordinary differential equations
    - is\_a algorithm using adaptive timesteps
      - is\_a Livermore solver
        - is\_a Livermore solver for ordinary differential equations

4 paths loaded. ☐ Multi-select ☐ Collapse ☐ Local

# Kinetic Simulation Algorithm Ontology Version Sat Jan 24 18:58:35 2009 UTC

kinetic simulation algorithm | [Link Here](#) | [Subscribe](#)

## View Ontology Summary

Details Visualization Notes (0) Mappings (0) Resource Index

Jump To:  Go

Legend

- kinetic simulation algorithm
  - algorithm using adaptive timesteps
    - Bortz-Kalos-Liebowitz method
    - code value ordinary differential equat
  - Gillespie-like stochastic simulation met
    - Gillespie-like approximate simulatio
    - Gillespie-like exact stochastic simu
    - particle-based spatial stochastic m
    - sub-volume stochastic reaction-dif
  - Green's function reaction dynamics
  - Livermore solver
  - algorithm using continuous variables
  - algorithm using deterministic rules
  - algorithm using discrete variables
  - algorithm using fixed timesteps
  - algorithm using non-spatial description
  - algorithm using spatial description
  - algorithm using stochastic rules

ID: KISAO:0000000

Full Id: [http://purl.bioontology.org/ontology/KISAO/KISAO\\_0000000](http://purl.bioontology.org/ontology/KISAO/KISAO_0000000)

Definitions: Algorithm used to instantiate a simulation from a mathematical model, where the variable values evolve over time.

Xref Definition: dk:260508

# Old KiSAO: subsumptions, multiple inheritance

The image shows a screenshot of the Old KiSAO ontology editor. On the left is a class hierarchy tree, and on the right is a detailed view of a specific class.

**Class Hierarchy (Left Panel):**

- Thing
  - 'kinetic simulation algorithm'
    - 'algorithm using adaptive timesteps'
    - 'algorithm using continuous variables'
    - 'algorithm using deterministic rules'
    - 'algorithm using discrete variables'
    - 'algorithm using fixed timesteps'
      - 'Euler backward method'
      - 'Euler forward method'
      - 'Runge-Kutta based method'
      - 'StochSim nearest-neighbour algorithm'
      - 'brownian diffusion Smoluchovski method'
      - 'deterministic cellular automata update algorithm'
      - 'multi-state agent-based simulation method'
    - 'algorithm using non-spatial description'
    - 'algorithm using spatial description'
      - 'Green's function reaction dynamics'
      - 'StochSim nearest-neighbour algorithm'
      - 'binomial tau-leap spatial stochastic simulation algorithm'
      - 'brownian diffusion Smoluchovski method'
      - 'deterministic cellular automata update algorithm'
      - 'partial differential equation method'
      - 'particle-based spatial stochastic method'
      - 'sub-volume stochastic reaction-diffusion algorithm'
    - 'algorithm using stochastic rules'
      - 'Bortz-Kalos-Liebowitz method'
      - 'Gillespie-like stochastic simulation method'
      - 'Smoluchowski equation based method'
        - 'Green's function reaction dynamics'
        - 'brownian diffusion Smoluchovski method'

**Class Definition (Right Panel):**

**def**

"\nIn the Brownian diffusion Smoluchowski method, \\\n"each molecule is treated as a point-like particle that diffuses freely in three-dimensional space. When a pair of reactive molecules collide, such as an enzyme and its substrate, a reaction occurs and the simulated reactants are replaced by products. [...] Analytic solutions are presented for some simulation parameters while others are calculated using look-up tables.\n"

Supported chemical processes include molecular diffusion, treatment of surfaces, zeroth-order-, unimolecular-,

**Description: 'brownian diffusion Smoluchovski method'**

Equivalent classes +

Superclasses +

- 'Smoluchowski equation based method'
- 'algorithm using discrete variables'
- 'algorithm using fixed timesteps'
- 'algorithm using spatial description'

Inherited anonymous classes

Members +

Keys +



# Old KiSAO: subsumptions, multiple inheritance

The screenshot displays the Old KiSAO software interface, which is used for defining and managing simulation algorithms. The interface is divided into two main panels.

**Left Panel: Class Hierarchy**

- Thing
  - 'kinetic simulation algorithm'
    - 'algorithm using adaptive timesteps'
    - 'algorithm using continuous variables'
    - 'algorithm using deterministic rules'
    - 'algorithm using discrete variables'
    - 'algorithm using fixed timesteps'
      - 'Euler backward method'
      - 'Euler forward method'
      - 'Runge-Kutta based method'
      - 'StochSim nearest-neighbour algorithm'
      - 'brownian diffusion Smoluchovski method'
      - 'deterministic cellular automata update algorithm'
      - 'multi-state agent-based simulation method'
    - 'algorithm using non-spatial description'
    - 'algorithm using spatial description'
      - 'Green's function reaction dynamics'
      - 'StochSim nearest-neighbour algorithm'
      - 'binomial tau-leap spatial stochastic simulation algorithm'
      - 'brownian diffusion Smoluchovski method'
      - 'deterministic cellular automata update algorithm'
      - 'partial differential equation method'
      - 'particle-based spatial stochastic method'
      - 'sub-volume stochastic reaction-diffusion algorithm'
    - 'algorithm using stochastic rules'
      - 'Bortz-Kalos-Liebowitz method'
      - 'Gillespie-like stochastic simulation method'
      - 'Smoluchowski equation based method'
        - 'Green's function reaction dynamics'
        - 'brownian diffusion Smoluchovski method'

**Right Panel: Definition Window**

**def**

"\nIn the Brownian diffusion Smoluchowski method, \\\n"each molecule is treated as a point-like particle that diffuses freely in three-dimensional space. When a pair of reactive molecules collide, such as an enzyme and its substrate, a reaction occurs and the simulated reactants are replaced by products. [...] Analytic solutions are presented for some simulation parameters while others are calculated using look-up tables.\\\nSupported chemical processes include molecular diffusion, treatment of surfaces, zeroth-order-, unimolecular-,

**Description: 'brownian diffusion Smoluchovski method'**

Equivalent classes +

Superclasses +

- 'Smoluchowski equation based method'
- 'algorithm using discrete variables'
- 'algorithm using fixed timesteps'
- 'algorithm using spatial description'

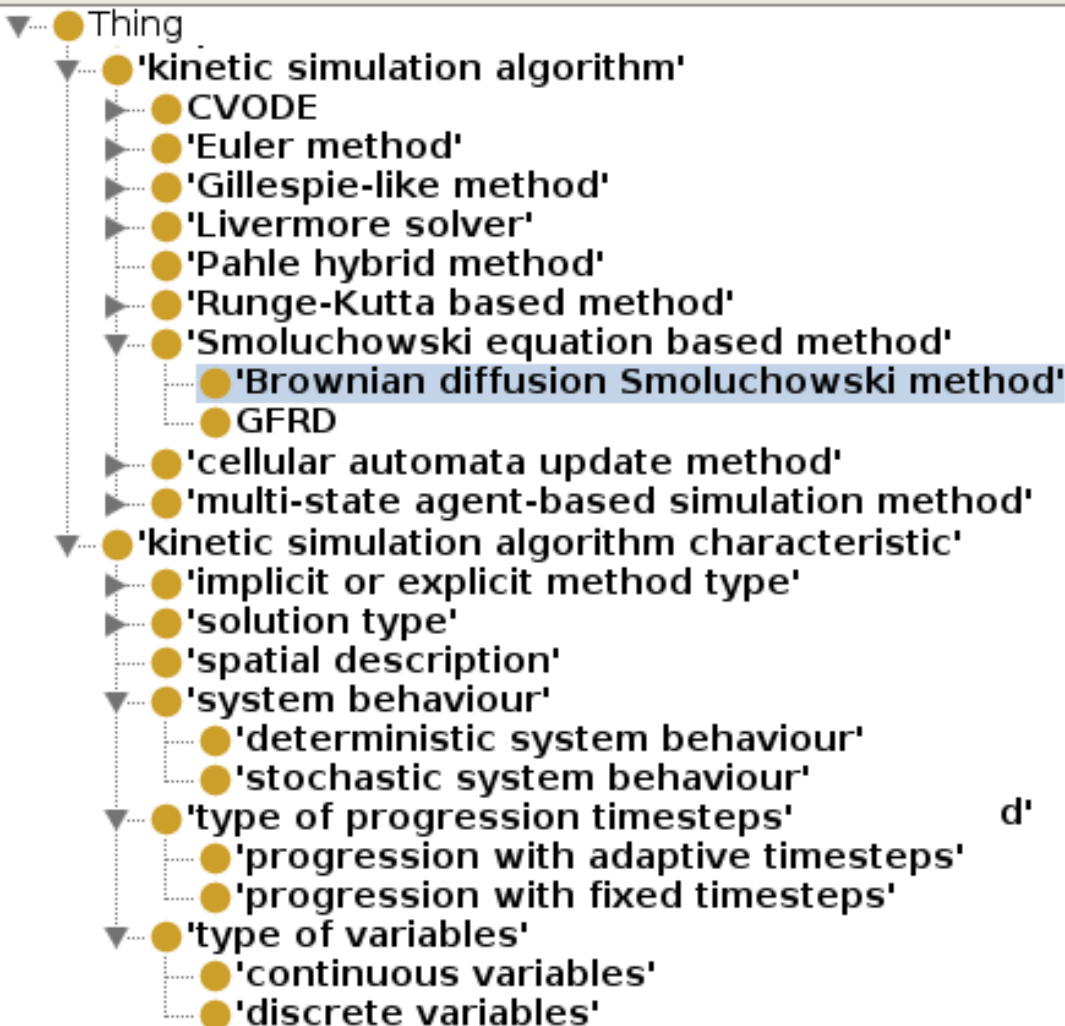
Inherited anonymous classes

Members +

Keys +

# New KiSAO: 'algorithm characteristic' branch, 'hasProperty' relation

Class hierarchy: 'Brownian diffusion Smoluchowski method'



comment

"In the Brownian diffusion Smoluchowski method, each molecule is treated as a point-like particle that diffuses freely in three-dimensional space. When a pair of reactive molecules collide, such as an enzyme and its substrate, a reaction occurs and the simulated reactants are replaced by products. [...] Analytic solutions are presented for some simulation parameters while others are calculated using look-up tables." Supported chemical processes include molecular diffusion, treatment of surfaces, zeroth-order-, unimolecular-, and bimolecular reactions."

label

"Brownian diffusion Smoluchowski method"^^string

Description: 'Brownian diffusion Smoluchowski method'

Equivalent classes +

Superclasses +

'Smoluchowski equation based method'

hasProperty some 'discrete variables'

hasProperty some 'progression with fixed timesteps'

hasProperty some 'spatial description'

Inherited anonymous classes

hasProperty some 'stochastic system behaviour'



# New KiSAO: 'algorithm characteristic' branch, 'hasProperty' relation

Class hierarchy: 'Brownian diffusion Smoluchowski method'

**Thing**

- 'kinetic simulation algorithm'**
  - CVODE
  - 'Euler method'
  - 'Gillespie-like method'
  - 'Livermore solver'
  - 'Pahle hybrid method'
  - 'Runge-Kutta based method'
  - 'Smoluchowski equation based method'**
    - 'Brownian diffusion Smoluchowski method'**
    - GFRD
  - 'cellular automata update method'
  - 'multi-state agent-based simulation method'
- 'kinetic simulation algorithm characteristic'**
  - 'implicit or explicit method type'
  - 'solution type'
  - 'spatial description'
  - 'system behaviour'**
    - 'deterministic system behaviour'
    - 'stochastic system behaviour'**
  - 'type of progression timesteps'**
    - 'progression with adaptive timesteps'
    - 'progression with fixed timesteps'**
  - 'type of variables'**
    - 'continuous variables'
    - 'discrete variables'**

**comment**

"In the Brownian diffusion Smoluchowski method, "each molecule is treated as a point-like particle that diffuses freely in three-dimensional space. When a pair of reactive molecules collide, such as an enzyme and its substrate, a reaction occurs and the simulated reactants are replaced by products. [...] Analytic solutions are presented for some simulation parameters while others are calculated using look-up tables." Supported chemical processes include molecular diffusion, treatment of surfaces, zeroth-order-, unimolecular-, and bimolecular reactions."

**label**

"Brownian diffusion Smoluchowski method"^^string

**Description: 'Brownian diffusion Smoluchowski method'**

Equivalent classes +

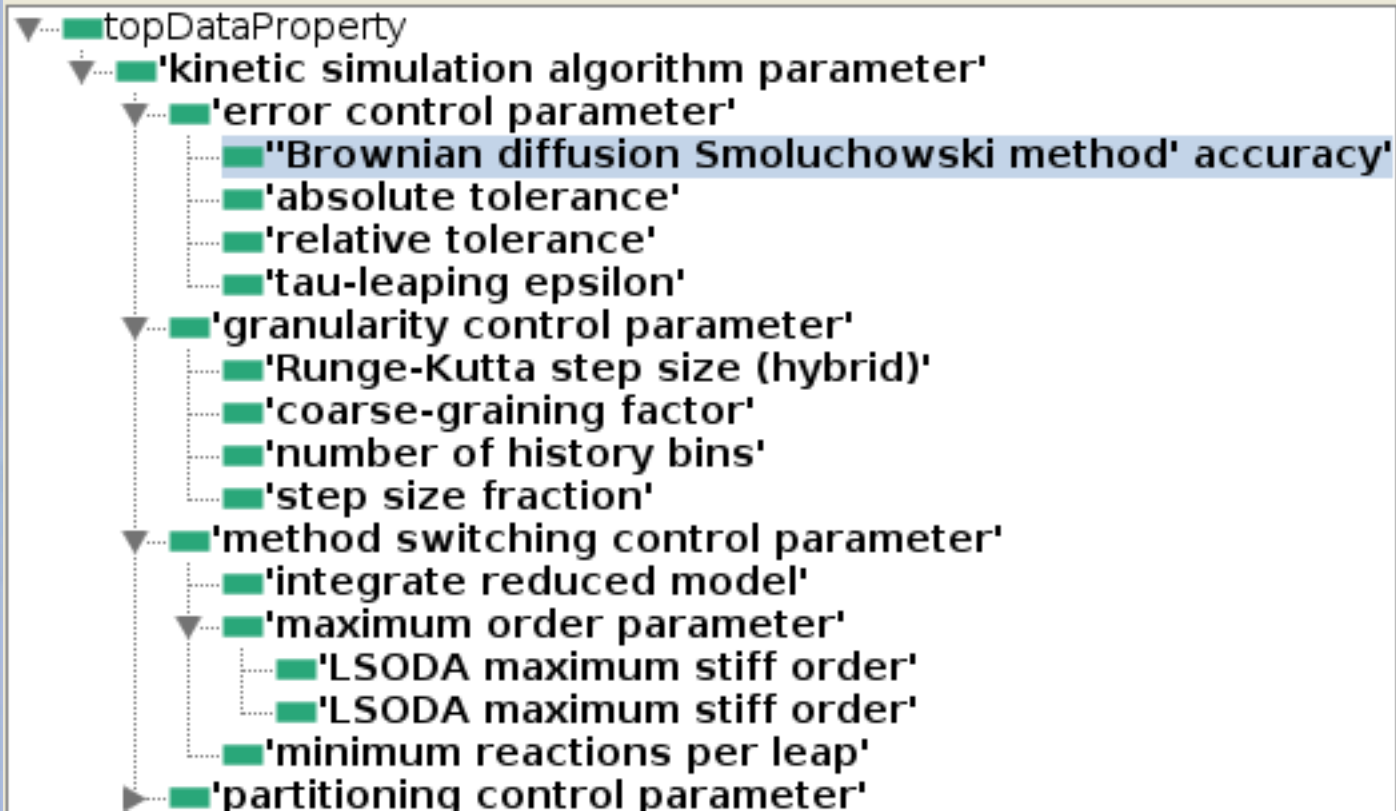
Superclasses +

- 'Smoluchowski equation based method'
- hasProperty some 'discrete variables'
- hasProperty some 'progression with fixed timesteps'
- hasProperty some 'spatial description'

Inherited anonymous classes

- hasProperty some 'stochastic system behaviour'

# New KiSAO: algorithm parameters



Consider the reaction  $A + B \rightarrow$  I suppose that A and B are within binding radius of each other. The reaction will always be performed and B are in the same virtual box accuracy is set to at least 3, then also occur if A and B are in

Description: "Brownian diffusion Smoluchowski method"

Domains (intersection) +

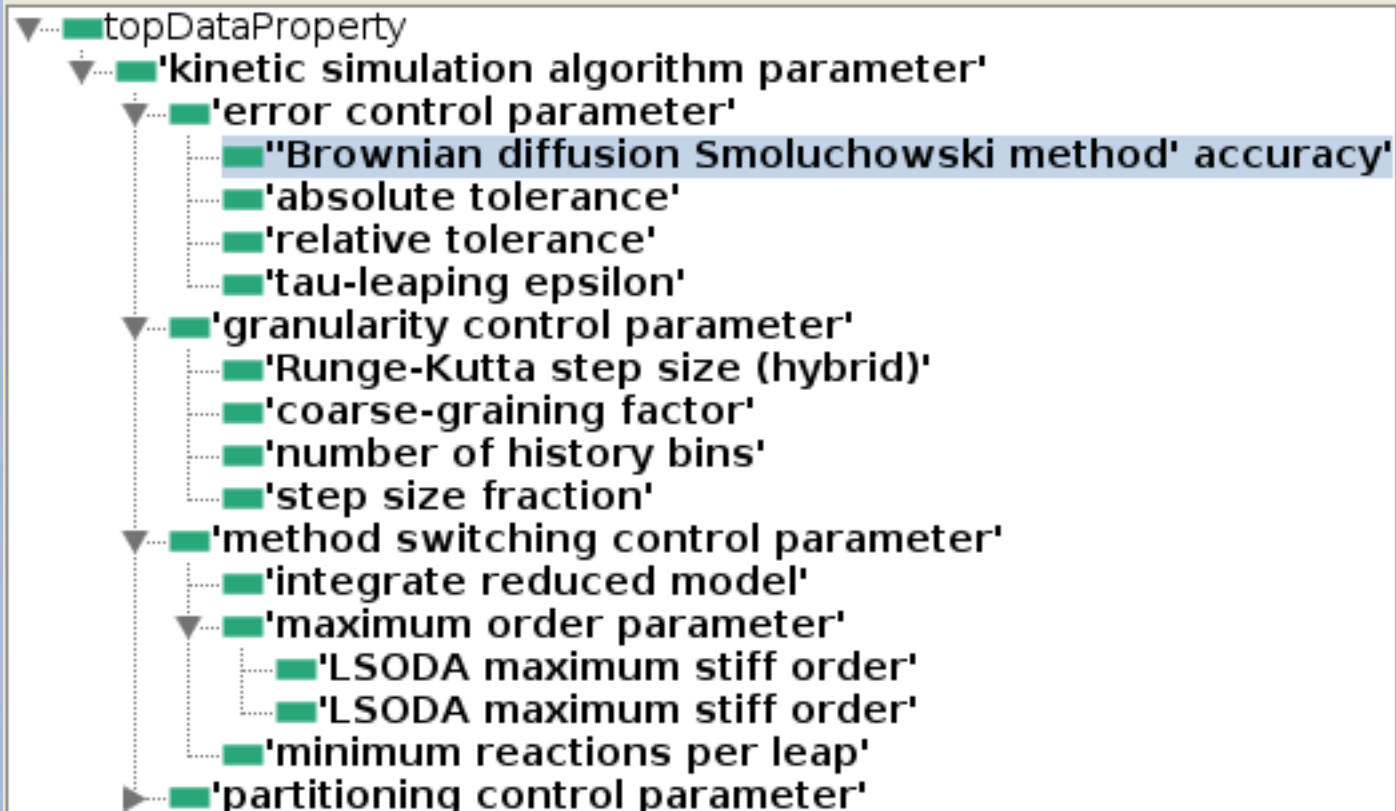
● "Brownian diffusion Smoluchowski method"

Ranges +

● int

Equivalent properties +

# New KiSAO: algorithm parameters



Consider the reaction  $A + B \rightarrow$  I suppose that A and B are within binding radius of each other. The reaction will always be performed and B are in the same virtual box accuracy is set to at least 3, then also occur if A and B are in

Description: "Brownian diffusion Smoluchowski method"

Domains (intersection) +

● 'Brownian diffusion Smoluchowski method'

Ranges +

● int

Equivalent properties +

# Algorithm Parameter Survey

[https://www.surveymonkey.com/  
s/kisao\\_parameters](https://www.surveymonkey.com/s/kisao_parameters)

# Acknowledgements



**Dagmar Waltemath**



**Anna Zhukova**