Modeling in DBM
Version 1.2

Adam Streck

Discrete Biomathematics, FU Berlin

June 14, 2013

1 Modeling

Models are described using an internal modeling language, based on the XML syntax [**?**], called *DBM* (Discrete biological model). The model is provided within a single DBM file that holds specification for the model together with static and dynamic constrains of the parametrization space. Such a file has to have a `.dbm` suffix, and to abide by the overall rules for XML files.

1.1 Model example

Every model must be enclosed within a pair `MODEL` tag. This tag has a single mandatory attribute `ver`, which must contain a floating point number specifying the employed version of the DBM language. A detailed description of the whole language is provided later in this section, here we present, as an example, model file for the network depicted in Figure **??**. This example model has a quite non-uniform syntax, which has been chosen on purpose to present different possibilities of model description.

```
<MODEL ver="1.0">
    <STRUCTURE>
        <SPECIE undef="basal" name="cA">
            <REGUL source="B" threshold ="1" label="Observable"/>
            <REGUL source="cA" />
        </SPECIE>
        <SPECIE>
            <REGUL source="cA" />
            <REGUL source="B" label="+" />
            <PARAM context="" value="?" />
            <PARAM context="B" value="1" />
            <PARAM context="cA,B" value="0,1" />
```

```
        </SPECIE>
    </STRUCTURE>
    ... specification of a property ...
</MODEL>
```

As can be seen, the model is a structure with two species, both being affected by two regulations. For the component B, the possible parametrizations space is reduced by requirements that the regulation from cA must be observable and that the effect of its self-regulation must be positive, if any. Also, the logical parameter of self-regulation of the component B must always be 1. As a result, the parametrization space is reduced to four possibilities.

1.2   Model property

The main purpose of the tool is picking parametrizations that satisfy some property. The description of this property can be given in one of two possible ways - either as Büchi automaton or as a time series, whose specification is inserted between **</STRUCTURE>** and **</MODEL>** tags. A time series is merely a Büchi automaton specialization, but as will be explained later the Parsybone is optimized for its usage and provides additional features if the time series is employed. To demonstrate the difference between the two, we present a single property described using each formalism. This property assures that the model in Figure **??** is able to reproduce a time series composed of the following three measurements:

1. $cA = 0 \wedge (B = 0 \vee B = 1)$

2. $cA \Leftrightarrow B$

3. $cA = 1 \wedge B = 0$

Only two out of four parametrizations allow for reproduction of this time series. To obtain them, we can describe the time series either using the Büchi automaton:

```
<AUTOMATON>
    <STATE final="0">
        <EDGE target="0" label="tt" />
        <EDGE target="1" label="cA=0" />
    </STATE>
    <STATE>
        <EDGE target="1" label="tt" />
        <EDGE target="last" label="((cA=0 &amp; B=0) | (cA=1 &amp; B=1))" />
    </STATE>
    <STATE name="last">
        <EDGE target="last" label="tt" />
        <EDGE target="3" label="(cA=1 &amp; B=0)" />
    </STATE>
```

```
        <STATE final="1">
            <EDGE target="3" label="tt" />
        </STATE>
    </AUTOMATON>
```

Or using the time series directly:

```
<SERIES>
    <EXPR values="cA=0" />
    <EXPR values="((cA=0 &amp; B=0) | (cA=1 &amp; B=1))" />
    <EXPR values="(cA=1 &amp; B=0)" />
</SERIES>
```

As can be seen, the second method makes the model quite shorter and shuld be used for description of time series.

1.3 Model holder description

- MODEL

    - Occurrence: single, mandatory.
    - Type: pair.
    - Parent: none.
    - Description: encloses the whole model.
    - Attributes:

        1. *ver*
            * Occurrence: mandatory.
            * Value: must be 1.0 in the current version of the tool.

1.4 Regulatory network description

- STRUCTURE

    - Occurrence: single, mandatory.
    - Type: pair.
    - Parent: MODEL.
    - Description: encloses the definition of a regulatory network.
    - Attributes: none.

- SPECIE

    - Occurrence: multiple, mandatory.
    - Type: pair.
    - Parent: STRUCTURE.
    - Description: defines a single specie.
    - Attributes:

1. *name*
   * Occurrence: optional.
   * Value: string containing letters and numbers.
   * Default: Capital letter, starting from `A`.
   * Description: name of the specie under which it will be further addressed.

2. *undef*
   * Occurrence: optional.
   * Value: basal/param/error.
   * Default: param.
   * Description: tells the system how it should handle values of regulatory contexts that are not specified. Basal means using a basal value, param means using all possible values and error causes error in case there are unspecified parameters.

3. *max*
   * Occurrence: optional.
   * Value: natural number.
   * Default: 1.
   * Description: maximal activation level this specie can occur in.

4. *basal*
   * Occurrence: optional.
   * Value: positive integer.
   * Default: 0.
   * Description: basal activation level of this specie - the value towards which the specie tends if not specified otherwise.

- REGUL
  - Occurrence: multiple, mandatory.
  - Type: solo.
  - Parent: SPECIE.
  - Description: defines a single incoming regulation of the parent specie.
  - Attributes:
    1. *source*
       * Occurrence: mandatory.
       * Value: name or the ordinal number of a specie.
       * Description: name of the specie that regulates this one.

    2. *threshold*
       * Occurrence: optional.
       * Value: natural number.
       * Default: 1.
       * Description: lowest activation level of the source specie that activates the regulation.

3. *label*
   * Occurrence: optional.
   * Value: a string, see Sec. 1.7.
   * Default: Free.
   * Description: describes nature of the regulation.

- PARAM

  - Occurrence: multiple.
  - Type: solo.
  - Parent: SPECIE.
  - Description: defines a single kinetic parameter.
  - Attributes:

    1. *context*
       * Occurrence: mandatory.
       * Value: comma separated list of active regulations, given by a name or an ordinal number of a regulator.
       * Description: defines the exact regulatory context in which this kinetic parameter is applied.

    2. *value*
       * Occurrence: optional.
       * Value: positive integer or ?.
       * Default: ?.
       * Description: specifies target value of the specie in this regulatory context, which must be one of possible activation levels of the specie. Character ? means that an exact number is unknown and a parametrization for each value between 0 and the maximal activation level of the specie is created.

1.5 Büchi automaton description

- AUTOMATON

  - Occurrence: single, present if and only if there is no sibling SERIES tag.
  - Type: pair.
  - Parent: MODEL.
  - Description: encloses decription of a Büchi automaton.
  - Attributes: none.

- STATE

  - Occurrence: multiple, mandatory.
  - Type: solo.
  - Parent: AUTOMATON.

- Description: defines a single state of the automaton. The first state in the description is also considered to be the initial state of the automaton.
- Attributes: none.

  1. *name*
     - Occurrence: optional.
     - Value: string containing letters and numbers.
     - Default: ordinal number of the tag, counting from zero.
     - Description: name of the specie under which it will be further addressed. System also uses its ordinal number (so the first state can be addressed using the name *0*).

  2. *final*
     - Occurrence: optional.
     - Value: Boolean.
     - Default: 0.
     - Description: specifies if the state is final (1) or not (0).

- **EDGE**
  - Occurrence: multiple, mandatory.
  - Type: solo.
  - Parent: STATE.
  - Description: defines an edge leading from the parent state.
  - Attributes:
    1. *target*
       * Occurrence: mandatory.
       * Value: name or the ordinal number of the target state.
    2. *label*
       * Occurrence: mandatory.
       * Value: logical formula (see Section 1.8), variables are atomic propositions (see Section 1.9).
       * Description: conditions that must be met for the edge to be transitive.

1.6   Time series description

- **SERIES**
  - Occurrence: single, present if and only if there is no sibling AUTOMATON tag.
  - Type: pair.
  - Parent: MODEL.
  - Description: encloses definition of a time series.
  - Attributes: none.

- **EXPR**

- Occurrence: multiple, mandatory.
- Type: solo.
- Parent: SERIES.
- Description: a single measurement in the time series.
- Attributes:
  1. *values*
     * Occurrence: mandatory.
     * Value: logical formula (see Section 1.8), variables of the formula must be atomic propositions (see Section 1.9).
     * Description: conditions that must be met for the measurement to be reproduced.

1.7   Edge label

There are two basic labels:

+ Meaning that the there must be a regulation whose parameter value increases if we add this regulator.

- Has an opposite meaning.

One can compose these labels using a logical formula over $+$ and $-$ or use one of the following predefined descriptions:

- Activating: $+$

- ActivatingOnly: $(+ \land \lnot -)$

- Inhibiting: $-$

- InhibitingOnly $(- \land \lnot +)$

- NotActivating $\lnot +$;

- NotInhibiting $\lnot -$

- Observable: $(+ \lor -)$

- NotObservable: $(\lnot + \land \lnot -)$

- Free: *true*

1.8 Formula construction

A formula is constructed using the following set of recursive rules:

1. *tt* and *ff* are formulas representing true or false respectively,

2. any variable is a formula,

3. for every formula A is $\neg A$ a formula,

4. for formulas A, B are $(A|B)$ and $(A\&B)$ formulas representing logical disjunction or conjunction respectively,

5. nothing else is a formula.

Note that in the model $\neg$ is denoted using `!` and $\&$ is denoted using `&`.

1.9 Atomic propositions

An atomic proposition is a string of the form: $specie * value$, where:

- *specie* denotes the name or ordinal number of a specie,

- $*$ denotes comparison operator from the set $\{<, >, =\}$,

- *value* denotes a positive integer with which the value of the specie is compared.

Note that in the model $<$ is denoted using `<` and $>$ is denoted using `>`.