TRUE

TEMPORAL REASONING UNIVERSAL ELABORATION

True System dynamics software

MANUAL Part 01

Concept

last release 2014/03/17

www.true-world.com

Contents

I - PRESENTATION

A) Description

TRUE = TEMPORAL REASONING UNIVERSAL ELABORATION

The TRUE software allows to model, to simulate, to optimize, to memorize and to restore the evolution of multidisciplinary systems calculated according to data, instructions and reasoning by :

- ❑ transparent and implicit mathematical modelling :
  - coupled differential equations of order n^n
- ❑ systems dynamic modeling in discrete time and continuous time (Euler)
- ❑ dynamic optimization with retro-calculation functions

Wysiwyg graphic interface which allows to create and to dynamically display models in 2D, 3D and 4D using a 3D modeler.

B) General principle

The evolution of a system can be regarded as a succession of states with regular time intervals times (discretization) which differences, since always explained, can be controlled.

Modeling consists in defining rules of evolution, for interval of time defined over given period, then to calculate and reproduce this evolution.
Rules are individually planned actions through seven temporal parameters.
They define the model temporality:

- **Rate** : the parameter Rate is only effective if the flow links two stocks.
  - If the parameter **Rate** is enable, the stocks will be updated by the result of the action (result = y / TimeStepI) at the end of the time unit, before the transition to the next time as in Vensim.
  - If the parameter **Rate** is not enable, the stocks will be updated by the result of the action (result = y) immediatly after that all the actions with the same **chronology** parameter has been calculated.

- **chronology**, **start**, **repeat**, **interval**, **type of cycle** and **filtered cycles**

It is possible to individually define a **cycle type** for each action : **cycle types** are:

- **default cycle**
- **float default cycle**, can be filtered by the current **default cycle**
- **float filtered cycle**, can bo be filtered by itself

The model is calculated for the asked **default cycles** number.

For each time unit, the actions, filtered and sorted by their temporal parameters, compute its code.
If the flow that contains actions links two stocks, their return values will be transfered between the source stock(subtract the return value) and the target stock (add the return value), according the Rate parameter.
The sum of all stocks of the model is constant for all units of time.

Stocks and flows values are memorized in the model data base for every time unit.

II - DETAILED PRINCIPLE

A) Principle of the Vensim software

Vensim software is a notable system dynamics software.

1. Calculation phases

Only for t=0

- **Phase 1**
  - Stocks are initialized with their initial value defined in their equation.
  - The stocks values are stored for the flows(t=0), graphs and tables.
- **End of phase 1**

- **Phase 2**
  - Flow calculations in an order automatically defined by the causalities.
  - Flow values stored for the other flows(t=0), graphs and tables.
- **End of phase 2**

- **Phase 3**
  The source and the target stocks of the flows (**Rate or Valve**) are updated with the flow values multiplied by the parameter **TIME STEP**
  Source stock = source stock - (value * TIME STEP)
  Target stock = target stock + (value * TIME STEP)
- **End of phase 3**

For t>0

For each unit of time (t)

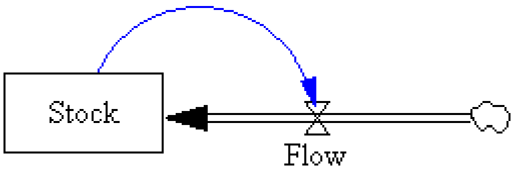
- **Phase 1**
  Stocks are initialized with their values at **t-1**: **stock(t)=stock(t-1)**
  - The stocks values are stored for the flows(t), graphs and tables.
- **End of phase 1**

- **Phase 2**
  Flow calculations in an order automatically defined by the causalities.
  Flows values are stored for other flows(t), graphs and tables
- **End of phase 2**

- **Phase 3**
  The source and the target stocks of the flows (**Rate or Valve**) are updated with the flow values multiplied by the parameter **TIME STEP**
  Source stock = source stock - (value * TIME STEP)
  Target stock = target stock + (value * TIME STEP)
- **End of phase 3**

End of for the current unit of time

Vensim example

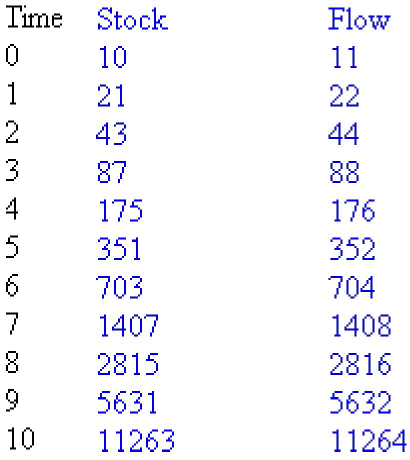


```
Initial value of the Stock = 10
Equation of the flow : Flow = Stock + 1
Equation of the Stock : Stock += Flow
```

Phases for units of time from 0 to n

	Phase 1	Phase2	Table of results
time	**Stocks calculation**	**Flows calculation**	
0	Stock(t0) = Initial value or calculation	Flow(t0) = Stock(t0) + 1	Flow(t0) & Stock(t0)
1	Stock(t1) = Stock(t0) + Flow(t0)	Flow(t1) = Stock(t1) + 1	Flow(t1) & Stock(t1)
2	Stock(t2) = Stock(t1) + Flow(t1)	Flow(t2) = Stock(t2) + 1	Flow(t2) & Stock(t2)
3	Stock(t3) = Stock(t2) + Flow(t2)	Flow(t3) = Stock(t3) + 1	Flow(t3) & Stock(t3)
n	Stock(tn) = Stock(tn) + Flow(tn)	Flow(tn) = Stock(tn) + 1	Flow(tn) & Stock(tn)

Results table



Time	Stock	Flow
0	10	11
1	21	22
2	43	44
3	87	88
4	175	176
5	351	352
6	703	704
7	1407	1408
8	2815	2816
9	5631	5632
10	11263	11264

2. Operation in continuous time

To get intermediate values and better accuracy, the model can run in continuous time.
When working in continuous time, the time unit number is increased and values that change the stock levels are decreased.

The TIMESTEP parameter defines the new time unit number:

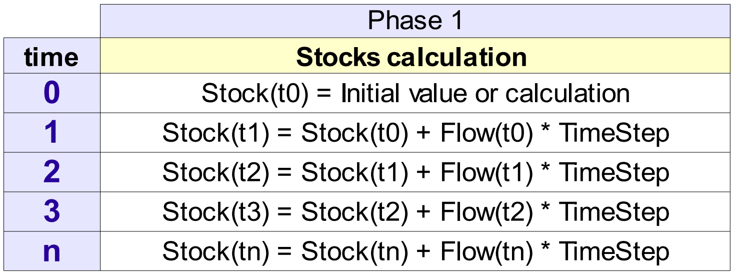
- 1 = 1 (discrete mode, the time unit number is not modified)
- 0,5 = 1/2 (continuous mode, the time unit number is multiplied by 2)
- 0,25 = 1/4 (continuous mode, the time unit number is multiplied by 4)
- 0,125 = 1/8 (continuous mode, the time unit number is multiplied by 8)
- 0,0625 = 1/16 (continuous mode, the time unit number is multiplied by 16)
- 0,03125 = 1/32 (continuous mode, the time unit number is multiplied by 32)
- 0,015625= 1/64 (continuous mode, the time unit number is multiplied by 64)
- 0,0078125 = 1/128 (continuous mode, the time unit number is multiplied by 128)

Values that change stock levels can be multiplied by the TIMESTEP, this is the simplest of the Euler method, or calculated with the Runge-Kutta method

Example: 20 units of time (years)

- discrete time: TIMESTEP=1, time unit number = 20 (years)
- continuous time: TIMESTEP=0,25, time unit number = (20*4)=80 (trimesters)

Phase 1 of the previous example, in continuous time



	Phase 1
time	**Stocks calculation**
0	Stock(t0) = Initial value or calculation
1	Stock(t1) = Stock(t0) + Flow(t0) * TimeStep
2	Stock(t2) = Stock(t1) + Flow(t1) * TimeStep
3	Stock(t3) = Stock(t2) + Flow(t2) * TimeStep
n	Stock(tn) = Stock(tn) + Flow(tn) * TimeStep

B) Principle of the True software

1. Calculation phases

In the True software :

- ❑ each action in a flow is individually planned with 7 temporal parameters
- ❑ one or several actions can be contained in one flow

see also www.true-world.com/htm/en/algorithm.php

Only for t=0

- ❑ **Phase 1**

  Stocks are initialized with their initial value:
  Stocks(t) = Stocks(initial value)

  Mirror stocks are initialized with the sum of their source stocks:
  Mirror stocks(t) = sum(source stocks(t))

  Flow values are cleared:
  Flows(t) = 0

  The stock, mirror stock and flow values are stored in the database for graphs and tables.
  Theses values represent the previous state of the system before the model starts calculation.
- ❑ **End of phase 1**

End of only for t=0

For time unit t= 1 to max time units

- **Phase 1**
  Stock values are initialized with their values at **t-1**: **stocks(t) = stocks(t-1)**
  Mirror stock values are initialized with their values at t-1: **Mirror stocks(t) = mirrors stocks(t-1)**
  Flow values are cleared: **flow(t) = 0**
- **End of phase 1**

- **Phase 2**
  The actions to be performed are:
  - filtered by their temporal parameters: start, interval,repeat, cycle type, filtered cycle,
  - added to several **sets of action** that will contain actions with same chronology parameters
  - sets of action are sorted by chronology
- **End of phase 2**

- **Phase 3, for set of actions = 1 to max set number**

  - **Phase3.1, for each action:**
    Action is computed
    Its **return value** is stored in its **buffer**
  - **End of phase 3.1**

  - **Phase 3.2, for each action**
    If the flow that contains the action links two stocks:
    - if the **Rate** parameter is **disable**:
    -- source stock is updated: **stock value(t) = stock value(t) - return value**
    -- target stock is updated: **stock value(t) = stock value(t) + return value**
    -- the stock values are stored in the database

    - if the **Rate** parameter is **enable**:
    -- the return value of the action is stored in its **RateBuffer**

    Flow value is updated: **flow value(t) = flow value(t) +return value**
    Flow value is stored in the database
  - **End of phase 3.2**

  - **Phase 3.3, for each mirror stock:**
    The mirror stock value is updated:
    - **mirror stock value(t) = sum(source stock values(t)**
  - **End of phase 3.3**

- **End of phase 3, for each set**

- **Phase 4, for each RateBuffer**

  Source and targed stock values of actions that have a RateBuffer (= **return value**) are updated:
  Source stock(t) = source stock(t) - (value / TimeStepI)
  Target stock(t) = target stock(t) + (value / TimeStepI)

  Note: **TimeStepI** = **1 / TIME STEP**

  Stock values are stored in the database
- **End of phase 4**

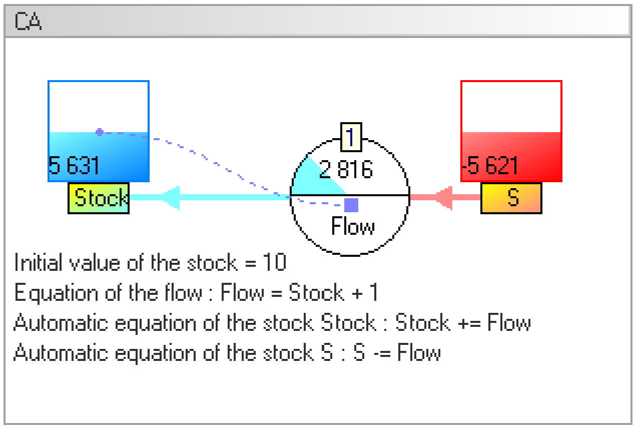
- **Phase 5, for each mirror stock:**

  Mirror stock values are updated with their source stocks value:
  - **mirror stock value(t) = sum (source stock values(t))**

  Mirror stocks values are stored in the database
- **End of phase5**

End of current time unit

True example in discrete time, parameter Rate not enable

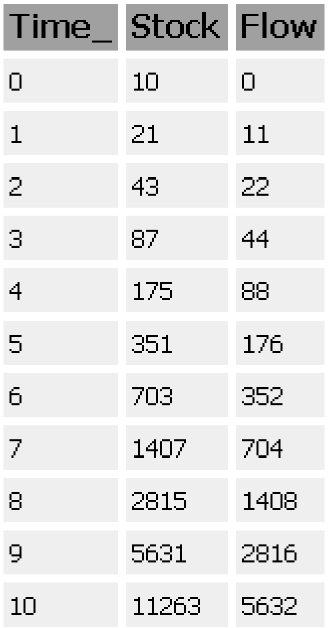


Initial value of the stock = 10
Equation of the flow : Flow = Stock + 1
Automatic equation of the stock Stock : Stock += Flow
Automatic equation of the stock S : S -= Flow

Phases for units of time from 0 to n

	Phase 1	Phase3	Phase4	Table of results
time	**Stocks calculation**	**Flows calculation**	**Stocks calculation**	
0	Stock(t0) = Initial value			0 & Stock(0)
1	Stock(t1) = Stock(t0)	Flow(t1) = Stock(t1) + 1	Stock(t1) += Flow(t1)	Flow(1) & Stock(1)
2	Stock(t2) = Stock(t1)	Flow(t2) = Stock(t2) + 1	Stock(t2) += Flow(t2)	Flow(2) & Stock(2)
3	Stock(t3) = Stock(t2)	Flow(t3) = Stock(t31) + 1	Stock(t3) += Flow(t3)	Flow(3) & Stock(3)
n	Stock(tn) = Stock(tn-1)	Flow(tn) = Stock(tn) + 1	Stock(tn) += Flow(tn)	Flow(n) & Stock(n)

Results table



Time_	Stock	Flow
0	10	0
1	21	11
2	43	22
3	87	44
4	175	88
5	351	176
6	703	352
7	1407	704
8	2815	1408
9	5631	2816
10	11263	5632

2. Operation in continuous time

True software can reproduce operation in continuous time according the Vensim software simplified Euler method.

Note : True uses the parameter **TimeStepI** (inverse of parameter **TIME STEP** in Vensim)
Example : if in Vensim **TIME STEP** = 0,5, in True **TimeStepI** = 2.

Once the parameter **TimeStepI** is initialized, because increasing the time unit number and dividing the values that affect stocks is not automatic, it is necessary to :

- ❑ To change the temporality of the model to increase the time unit number, change the ***Repeat*** parameter for the concerned actions

- ❑ To change the values updating the stocks in continous time, enable the parameter **Rate** of the action

➢ Note :

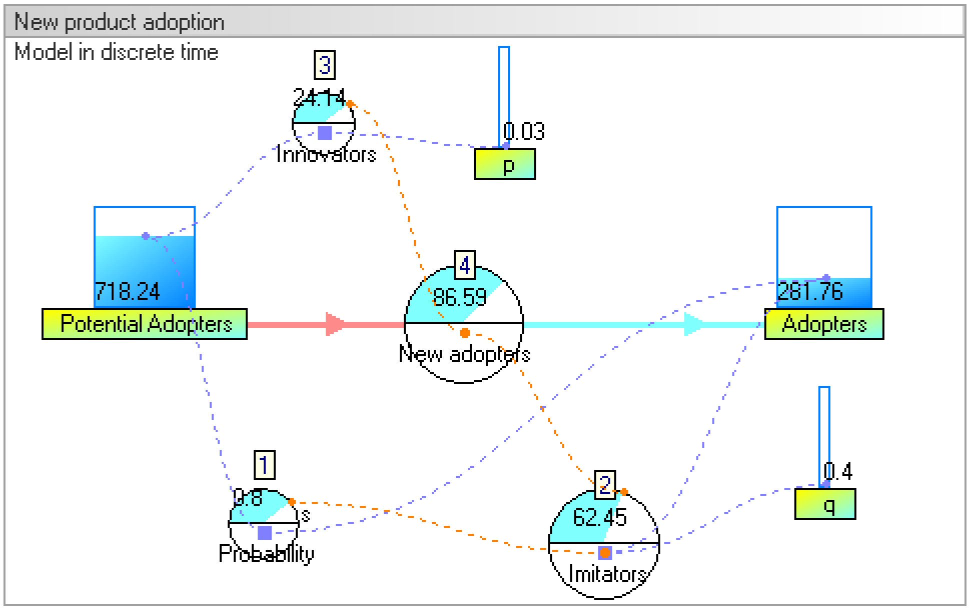
- Because continuous time is not automatic, it is possible to build models that work both in continuous and discrete mode

Calculation phases of the example in continuous time

	Phase 1	Phase3	Phase4
time	**Stocks calculation**	**Flows calculation**	**Stocks calculation**
0	Stock(t0) = Initial value		
1	Stock(t1) = Stock(t0)	Flow(t1) = Stock(t1) + 1	Stock(t1) += Flow(t1) / TimeStepI
2	Stock(t2) = Stock(t1)	Flow(t2) = Stock(t2) + 1	Stock(t2) += Flow(t2) / TimeStepI
3	Stock(t3) = Stock(t2)	Flow(t3) = Stock(t31) + 1	Stock(t3) += Flow(t3) / TimeStepI
n	Stock(tn) = Stock(tn-1)	Flow(tn) = Stock(tn) + 1	Stock(tn) += Flow(tn) / TimeStepI

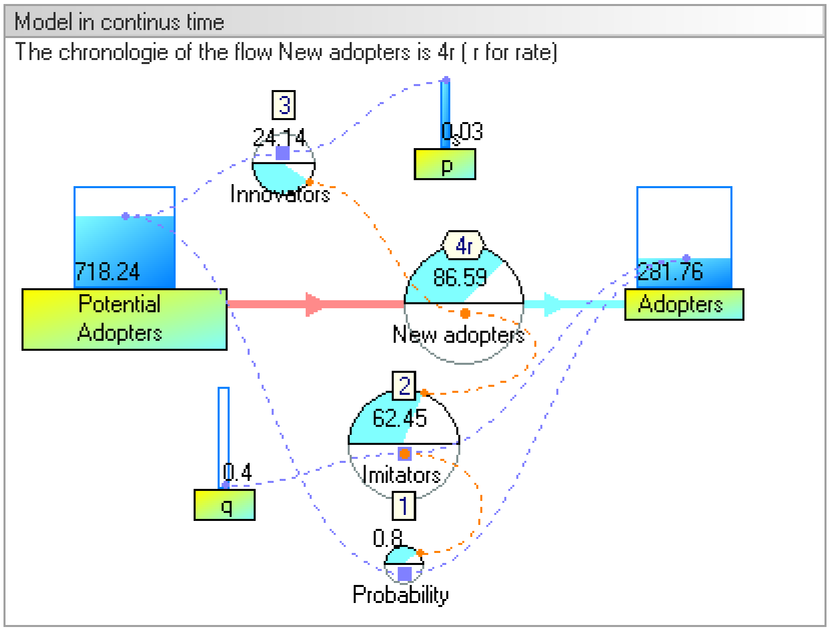
Transform a model in continuous time

Model in discrete mode



- ❑ The temporality of this model is 15 years : one cycle with 15 time units, but it is possible to set 15 cycles with 1 unit of time

Model in continuous time



Model in continus time

The chronologie of the flow New adopters is 4r (r for rate)

We want to transform this model in continuous time with a **TimeStepI** = 12

Adjustment of temporality :

- The temporality of this model will be 15 years (cycles) of 12 months (units of time)
- Initialize the *Repeat* parameters of the actions to 12
- Initialize the asked computed cycles to 15 (the model will be computed for 15 cycles)
- Enable the parameter **Rate** of the flow **New adopters**

III - TRUE / VENSIM DIFFERENTIATION - MODEL IMPORT

See http://www.true-world.com/htm/en/import.php

IV - ADVICE

See http://www.true-world.com/htm/en/advice.php

V - WORKING WITH TRUE FUNCTIONS

See http://www.true-world.com/htm/en/workingwith.php

VI - EXERCISES

See http://www.true-world.com/htm/en/exercises.php

See manual **Man60-Exercises.pdf**