

# Application for Generating Transition Images after Its Normalization Using Operator G1

**Dimitar Tomov**

Faculty of Computer Science and Engineering, Software Engineering  
Burgas Free University  
62 San Stefano Street, Burgas - 8001, Bulgaria  
email: xdtomov@gmail.com

**Abstract:** The article explains the process of developing an algorithm for the normalizing operator G1. There is a generalized net model of the algorithm as well as a flow diagram. It also illustrates an application developed that generates the final net image.

**Keywords:** Generalized nets, Operator G1, Automation.

**AMS Classification:** 68Q85, 94A08.

## 1 Introduction

When it comes to generalized nets, automating the work that is needed when developing and comparing different models is crucial to future development. When it comes to comparing two different models that perform the same task we can use complexity operators. But sometimes two models can be very different from one another – for example one might have more places and less transitions while the other – the opposite. It is useful before comparing the two models to unify them. One such unifying method is normalizing models using operator G1 [1, 4]. Using standard algorithm developing techniques I have created a simple but effective solution to the task of G1 normalization and an application using that algorithm to generate images of normalized transitions.

## 2 Algorithm for normalization of a transition using operator G1

Manual solution to the normalization task [2, 3]:

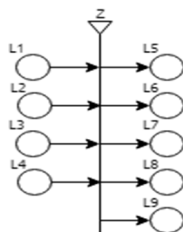


Figure 1. Input transition

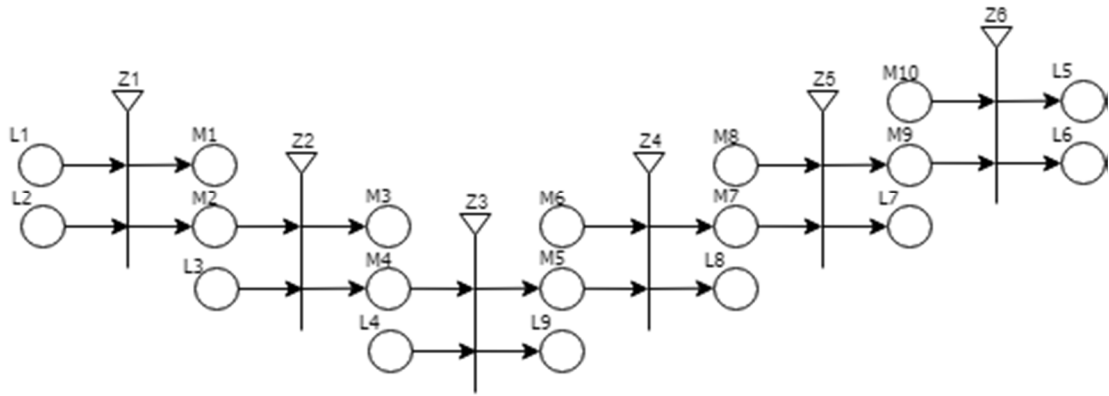


Figure 2. Output net

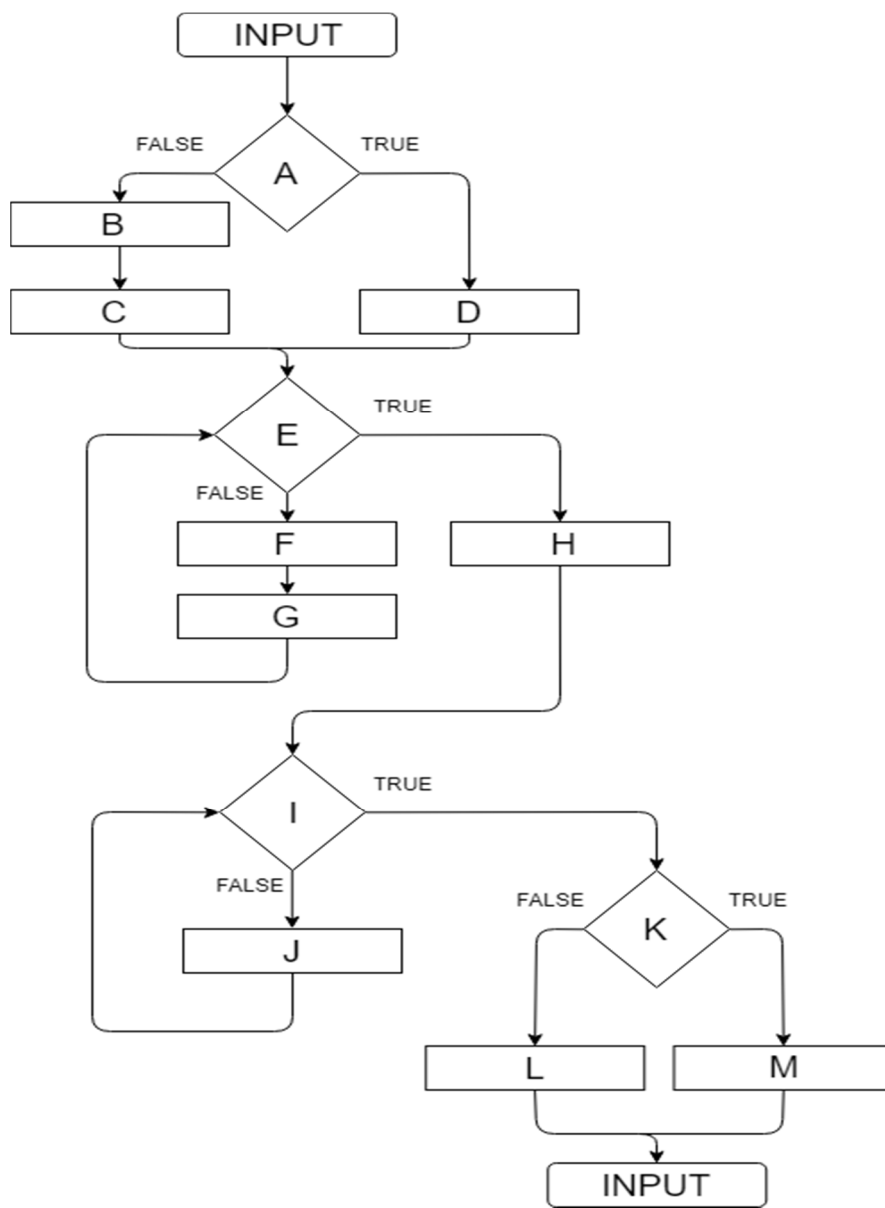


Figure 3. Traditional flowchart of normalization algorithm of operator G1

- A – Does the transition have at least 2 input places?
- B – Take one input place.
- C – Add one fictitious place.
- D – Take two input places.
- E– Are there no more input places?
- F – Add two fictitious places.
- G – Take one input place.
- H – Reverse building direction.
- I – Does the transition have two or less output places?
- J – Take one output place and add one fictitious place.
- K – Does the transition have only one output places?
- L – Take two output places.
- M – Take 1 output place and add one fictitious place.
- INPUT – Transition object.
- OUTPUT – Generalized Net object.

### 3 Example execution of the application

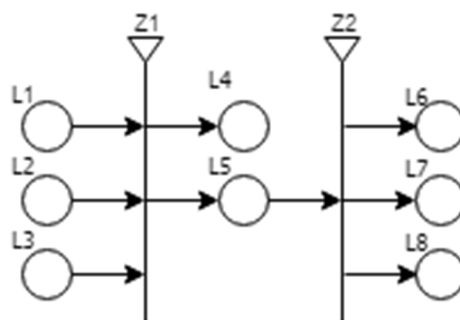


Figure 4. Input

Enter the number of transitions in the net and the name of the file for the output images:

The screenshot shows a window titled "Generalized Nets - G1". Inside the window, there is a form with the following elements:

- A label "How many transitions:" followed by a text input field containing the number "2".
- A label "File name:" followed by a text input field containing the text "MyFirstGenNet".
- A button labeled "NEXT" located below the file name input field.

Enter number of input and output places:

Generalized Nets - G1

Z1 Nº Inputs: 3 Nº Outputs: 2

Z2 Nº Inputs: 1 Nº Outputs: 3

NEXT

Enter the names of input and output place for each transition:

Generalized Nets - ...

Transition Z1

Inputs: I1 I2 I3

Outputs: I4 I5

NEXT

Generalized Nets - ...

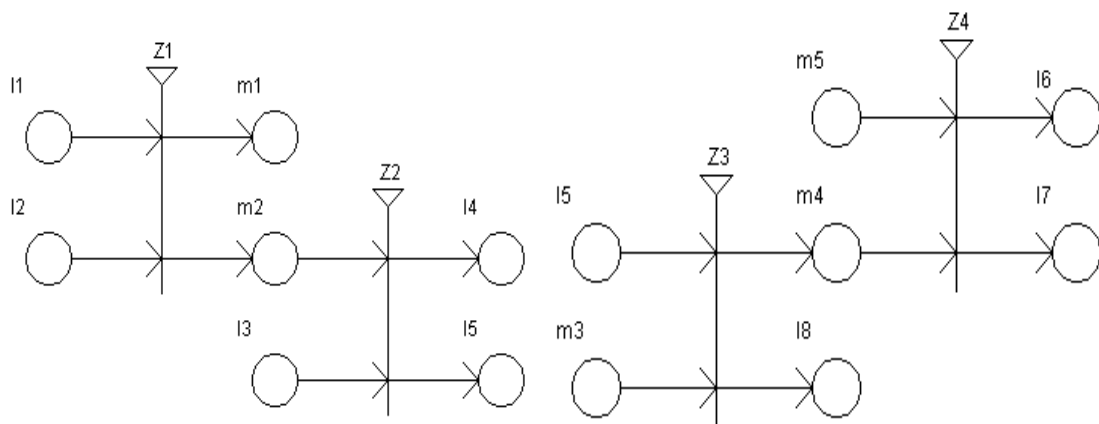
Transition Z2

Inputs: I5

Outputs: I6 I7 I8

NEXT

Results from application's execution (automatically generated images):



Two separate images were generated that cannot be stitched together automatically because of all the diverse ways two transitions can be connected to one another.

Examples of larger transitions generated by the application:

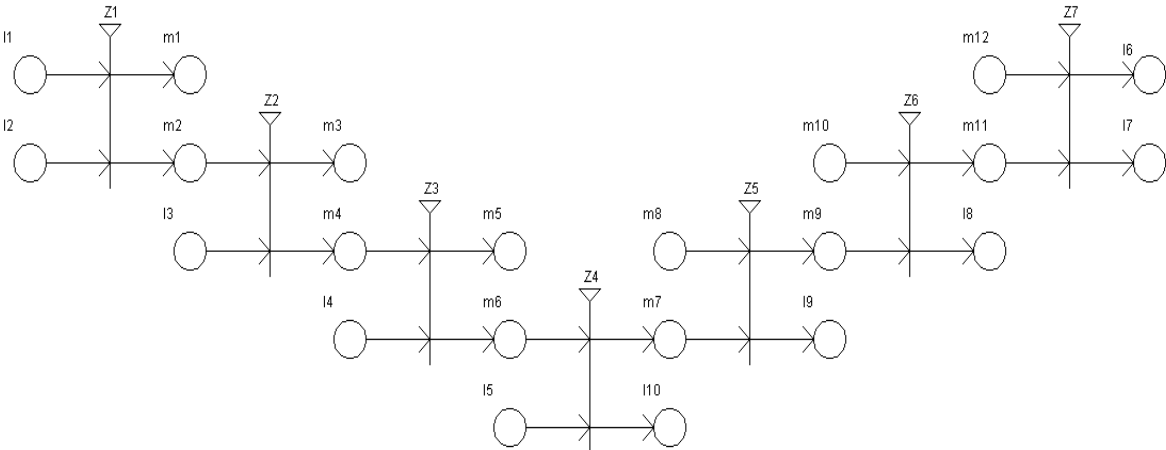


Figure 5. Example 1

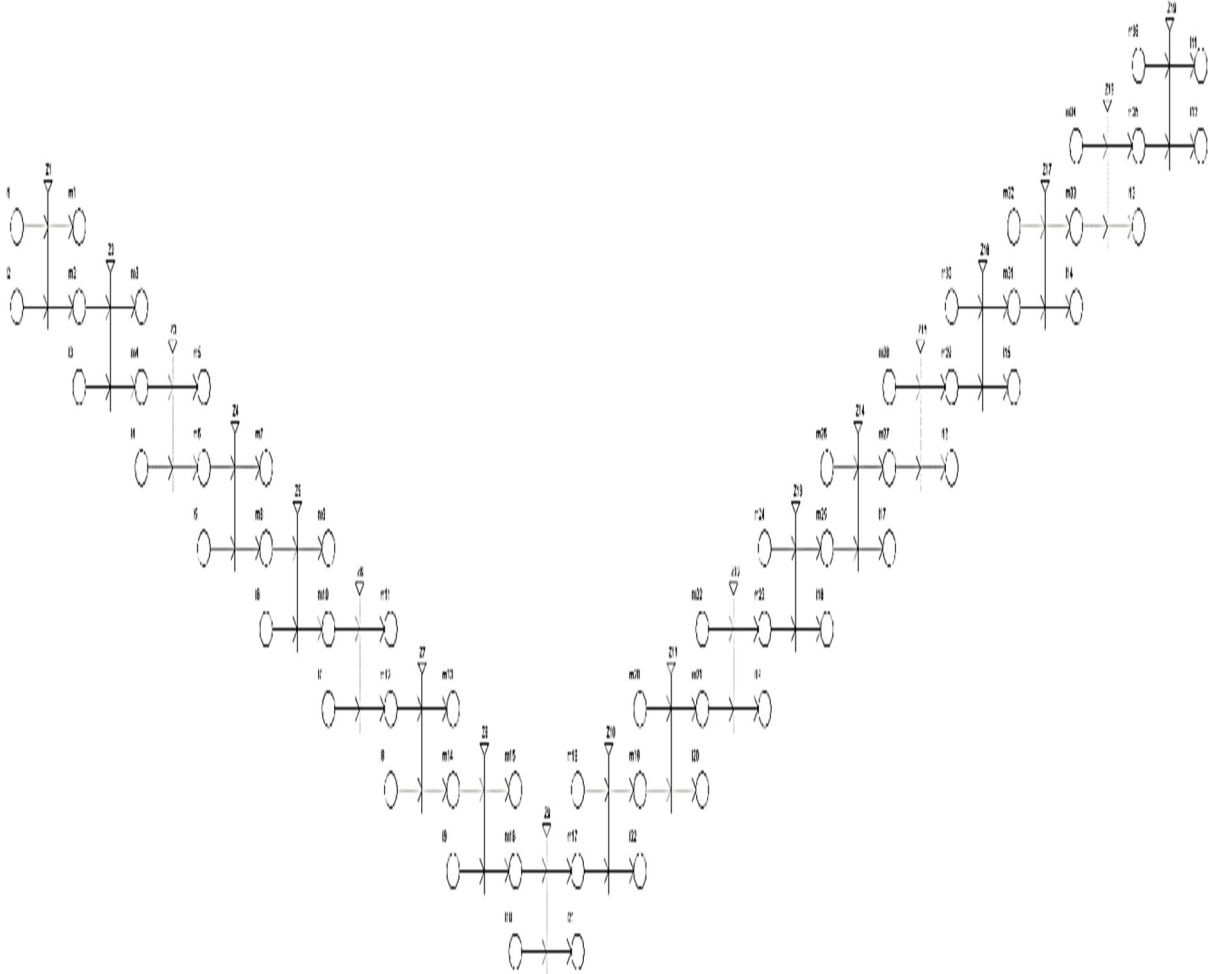


Figure 6. Example 2

Additional notes for the application [5, 6]:

- Application has a restriction of maximum 33 transitions at a time and no restriction of number of inputs/outputs.
- The letter ‘m’ should not be used as a name for an input or output, it is reserved for fictitious places’ names. The application will not give an error but will produce a wrong image.
- The application has system for error detection when inputting the transitions.
- You will need Java Runtime Environment (JRE) to use it.

#### 4 Generalized Net Model of the Algorithm for normalization of a transition using operator G1

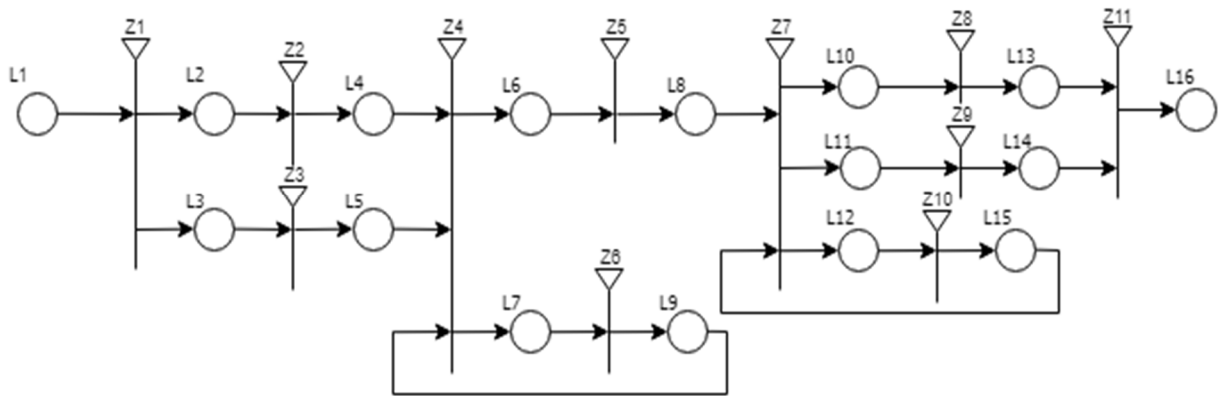


Figure 7. The GN model

Used abbreviations:

- OT – Object of type “Transition”;
- OGN – Object of type “Generalized Net”;

The starting characteristics of the token in  $L_1$  is “An Object of type “Transition”” (OT), which contains the number of input and output places and their names.

$M$  – the capacities of the places is 1 for all.

Transition  $Z_1$  – Create an empty Object of type “Generalized Net”.

$$Z_1 = \langle \{L_1\}, \{L_2, L_3\}, R_1, \vee\{L_1\} \rangle$$

$$R_1 = \frac{\begin{array}{c|cc} & L_2 & L_3 \\ \hline L_1 & W_{1,2} & W_{1,3} \end{array}}$$

Token receives a characteristic – “Empty OGN”;

$W_{1,2}$  – OT has at least 2 input places;

$W_{1,3}$  – OT has only 1 input place;

Transition  $Z_2$  – From OT take two input places and add them to OGN. Add a transition to OGN.

$$Z_2 = \langle \{L_2\}, \{L_4\}, R_2, \vee\{L_2\} \rangle$$

$$R_2 = \frac{\quad}{L_2} \left| \begin{array}{l} L_4 \\ true \end{array} \right.$$

Transition  $Z_3$  - From OT take one input place and add it to OGN. Add a fictitious place and a transition to OGN.

$$Z_3 = \langle \{L_3\}, \{L_5\}, R_3, \vee\{L_3\} \rangle$$

$$R_3 = \frac{\quad}{L_3} \left| \begin{array}{l} L_5 \\ true \end{array} \right.$$

$Z_4$  – Check number of remaining input places of OT.

$$Z_4 = \langle \{L_4, L_5, L_9\}, \{L_6, L_7\}, R_4, \vee\{L_4, L_5, L_9\} \rangle$$

$$R_4 = \frac{\quad}{L_4} \left| \begin{array}{cc} L_6 & L_7 \\ W_{4,6} & W_{4,7} \\ L_5 & W_{5,6} & W_{5,7} \\ L_9 & W_{9,6} & W_{9,7} \end{array} \right.$$

$W_{4,6} = W_{5,6} = W_{9,6}$  – OT has no more input places;

$W_{4,7} = W_{5,7} = W_{9,7}$  – OT has more input places;

$Z_5$  – Reverse building direction.

$$Z_5 = \langle \{L_6\}, \{L_8\}, R_5, \vee\{L_6\} \rangle$$

$$R_5 = \frac{\quad}{L_6} \left| \begin{array}{l} L_8 \\ true \end{array} \right.$$

Transition  $Z_6$  - From OT take one input place and add it to OGN. Add a transition and two fictitious places to OGN.

$$Z_6 = \langle \{L_7\}, \{L_9\}, R_6, \vee\{L_7\} \rangle$$

$$R_6 = \frac{\quad}{L_7} \left| \begin{array}{l} L_9 \\ true \end{array} \right.$$

Transition  $Z_7$  – Check number of remaining output places of OT.

$$Z_7 = \langle \{L_8, L_{15}\}, \{L_{10}, L_{11}, L_{12}\}, R_7, \vee\{L_8, L_{15}\} \rangle$$

$$R_7 = \frac{\quad}{L_8} \left| \begin{array}{ccc} L_{10} & L_{11} & L_{12} \\ W_{8,10} & W_{8,11} & W_{8,12} \\ L_{15} & W_{15,10} & W_{15,11} & W_{15,12} \end{array} \right.$$

$W_{8,10} = W_{15,10}$  – OT has only two output places;

$W_{8,11} = W_{15,11}$  – OT has only one output place;

$W_{8,12} = W_{15,12}$  – OT has more than two output places.

Transition  $Z_8$  - Add a transition to OGM. From OT take the last two output places and add them to OGM.

$$Z_8 = \langle \{L_{10}\}, \{L_{13}\}, R_8, \vee\{L_{10}\} \rangle$$

$$R_8 = \frac{\quad}{L_{10}} \left| \begin{array}{l} L_{13} \\ true \end{array} \right.$$

Transition  $Z_9$  - Add a transition to OGM. From OT take the last input place and add it to OGM. Add a fictitious position for output to OGM.

$$Z_9 = \langle \{L_{11}\}, \{L_{14}\}, R_9, \vee\{L_{11}\} \rangle$$

$$R_9 = \frac{\quad}{L_{11}} \left| \begin{array}{l} L_{14} \\ true \end{array} \right.$$

Transition  $Z_{10}$  - Add a transition to OGM. From OT take an output place and add it to OGM. Add a fictitious place for output to OGM.

$$Z_{10} = \langle \{L_{12}\}, \{L_{15}\}, R_{10}, \vee\{L_{12}\} \rangle$$

$$R_{10} = \frac{\quad}{L_{12}} \left| \begin{array}{l} L_{15} \\ true \end{array} \right.$$

Transition  $Z_{11}$  - Remove OT from the token and leave it only with OGM.

$$Z_{11} = \langle \{L_{13}, L_{14}\}, \{L_{15}\}, R_{11}, \vee\{L_{13}, L_{14}\} \rangle$$

$$R_{11} = \frac{\quad}{\begin{array}{l} L_{13} \\ L_{14} \end{array}} \left| \begin{array}{l} L_{15} \\ true \\ true \end{array} \right.$$

## 5 Conclusion

The article illustrates the algorithm and application used for automation of the normalization process using G1 operator. The model can be used to demonstrate the process of the G1 operator. It would be fruitful to pursue further research about how various aspects of generalized nets can be automated to make working with them easier.

## References

- [1] Atanassov, K., E. Sotirova, *GeneralizedNets*, Prof. M. Drinov Academic Publ. House, Sofia, 2017.
- [2] Orozova, D., *Generalized Net Models of intelligent tutoring environments*, "Prof. Marin Drinov" Academic Publishing House, Sofia, 2011.



- [3] Orozova, D., K. Atanassov, Generalized Net Model of the Process of Selection and Usage of an Intelligent e-Learning System, *Comptes rendus de l'Académie bulgare des Sciences*, book No 5, vol. 65, 2012, 591-598.
- [4] *Generalized Nets Wiki*, Web resource: [http://www.ifeenia.org/wiki/Generalized\\_nets](http://www.ifeenia.org/wiki/Generalized_nets)
- [5] *Java Standard Edition 7 API Reference*, Web resource: <https://docs.oracle.com/javase/7/docs/api/>
- [6] *IntelliJ Java IDE*, Web resource: <https://www.jetbrains.com/idea/>