Matlab's Functions:

FILOSOFI: this function returns Pre, Post matrix and the initial marking
for an example of philosopher's system.

****

##SYNTAX##

[Pre,Post,M0]=Filosofi(n,m):

Input arguments:

n: Philosopher's number (it must be at least 2)

m: Number of philosopher's state (it must be at least 3)

Output arguments:

Pre: Pre-incidence matrix of Petri net

Post: Post-incidence matrix of Petri net

M0: Initial marking of Petri net

See also: ORDER1, ORDER2, UNFOLDING_N, GRAPH1, GRAPH2, UNFOLDING_GRAPH, GRAPHMCM

****
****

GRAPH1: this function returns the instruction to build the graph of order 1 unfolding of a Petri net

****

##SYNTAX##

graph1(Pre,Post,M0) :

Input arguments:

Pre: Pre-incidence matrix of Petri net

Post: Post-incidence matrix of Petri net

M0: Initial marking

See also: ORDER1, ORDER2, UNFOLDING_N, UNFOLDING_GRAPH, GRAPH2.

****
****

GRAPH2: this function returns the instruction to build the graph of order 2 unfolding of a Petri net

****

##SYNTAX##

unfolding_graph(Pre,Post,M0) :

Input arguments:

Pre: Pre-incidence matrix of Petri net

Post: Post-incidence matrix of Petri net

M0: Initial marking

See also: ORDER1, ORDER2, UNFOLDING_N, GRAPH1, UNFOLDING_GRAPH.

****
****

GRAPHMCM: this function returns the instruction to build the graph of finite McMillan's prefix of a Petri net

****

##SYNTAX##

unfolding_graph(Pre,Post,M0) :

Input arguments:

Pre: Pre-incidence matrix of Petri net

Post: Post-incidence matrix of Petri net

M0: Initial marking

See also: ORDER1, ORDER2, UNFOLDING_N, GRAPH1, GRAPH2.

****
****

ORDERMCM: this function returns the finite prefix of McMillan of a Petri net.

****

##SYNTAX##

[Pre_u,Post_u,p_tier,t_tier,Pexp,Texp,cut_off,cut_off_ord]=orderMcM(Pre,Post,M0) :

Input arguments:
Pre: Pre-incidence matrix of Petri net
Post: Post-incidence matrix of Petri net
M0: Initial marking

Output arguments:
Pre_u: Pre-incidence matrix of unfolding net
Post_u: Post-incidence matrix of unfolding net
p_tier: It is a cell array containing vectors of places' tiers
t_tier: It is a cell array containing vectors of transitions' tiers
Pexp: It is a vector containing all places of unfolding net
Texp: It is a vector containing all transitions of unfolding net
cut_off: It is a vector containing cut off transitions
cut_off_ord: It is a vector containing ordered cut off transitions

See also: ORDER1, ORDER2, UNFOLDING_N, GRAPH1, GRAPH2, UNFOLDING_GRAPH, GRAPHMCM

****
****

ORDER1: this function returns the order 1 unfolding of a Petri net

****
##SYNTAX##
[cut_off1,Pre_u,Post_u,p_tier,t_tier,Pexp,Texp]=order1(Pre,Post,M0) :

Input arguments:
Pre: Pre-incidence matrix of a Petri net
Post: Post-incidence matrix of a Petri net
M0: Initial marking of a Petri net

Output arguments:
Pre_u: Pre-incidence matrix of unfolding net
Post_u: Post-incidence matrix of unfolding net
p_tier: It is a cell array containing vectors of places' tiers
t_tier: It is a cell array containing vectors of transitions' tiers
Pexp: It is a vector containing all places of unfolding net
Texp: It is a vector containing all transitions of unfolding net
cut_off1: It is a vector containing cut off transitions
U: It is a cell array containing all output just described.

See also: ORDER2, UNFOLDING_N, MCMILLAN GRAPH1, GRAPH2, UNFOLDING_GRAPH, GRAPHMCM

****
****

ORDER2: this function returns the order 2 unfolding of a Petri net

****

##SYNTAX##

[Pre_u,Post_u,p_tier,t_tier,Pexp,Texp,cut_off1,cut_off2]=order2(Pre,Post,M0) :

Input arguments:

Pre: Pre-incidence matrix of Petri net
Post: Post-incidence matrix of Petri net
M0: Initial marking

Output arguments:

Pre_u: Pre-incidence matrix of the unfolding net
Post_u: Post-incidence matrix of the unfolding net
p_tier: It is a cell array containing vectors of places' tiers
t_tier: It is a cell array containing vectors of transitions' tiers
Pexp: It is a vector containing all places of the unfolding net
Texp: It is a vector containing all transitions of the unfolding net
cut_off1: It is a vector containing order 1's cut off transitions
cut_off2: It is a vector containing order 2's cut off transitions

See also: ORDER1, UNFOLDING_N, MCMILLAN, GRAPH1, GRAPH2, UNFOLDING_GRAPH, GRAPHMCM

****
****

REACH_U: this function can be used in four different forms that depend on the parameter "opt":

****

##SYNTAX##

[ret,p_corr,X]=reach_u(M0,M,U,opt) :

1) opt= it is not present.

[ret]=reach_u(M0,M,U)

This function says if the generic marking M of a Petri net is reachable.

Input arguments:

M0: Initial marking of a Petri net

M: Generic marking of a Petri net
U: It is a cell array containing the output of order1 function.
Output arguments:
ret: It returns 1 if M is reachable, 0 otherwise.

2) opt= 1

[ret,p_corr]=reach_u(M0,M,U,1)

This function says if the generic marking M of a Petri net is reachable and returns the unfolding's sets of places correspondents.

Input arguments:
M0: Initial marking of a Petri net
M: Generic marking of a Petri net
U: It is a cell array containing the output of order1 function.
opt: Modality of operation.
Output arguments:
ret: It returns 1 if M is reachable, 0 otherwise.
p_corr: It is a matrix. In its lines there are all sets of unfolding's places that corresponds to the marking M.

3) opt= 2

[ret,p_corr,X]=reach_u(M0,M,U,2)

This function says if the generic marking M of a Petri net is reachable and returns the unfolding's sets of places correspondents. It is returned the shortest configuration of the marking M too.

Input arguments:
M0: Initial marking of a Petri net
M: Generic marking of a Petri net
U: It is a cell array containing the output of order1 function.
opt: Modality of operation.
Output arguments:
ret: It returns 1 if M is reachable, 0 otherwise.
p_corr: It is a matrix. In its lines there are all sets of unfolding's places that corresponds to the marking M.
X: It is a vector containing the shortest configuration of the marking M.

3) opt= 2

[ret,p_corr,X]=reach_u(M0,M,U,2)

This function says if the generic marking M of a Petri net is reachable and returns the unfolding's sets of places correspondents. The function returns all configurations of the marking M too.

Input arguments:

M0: Initial marking of a Petri net
M: Generic marking of a Petri net
U: It is a cell array containing the output of order1 function.
opt: Modality of operation.
Output arguments:
ret: It returns 1 if M is reachable, 0 otherwise.
p_corr: It is a matrix. In its lines there are all sets of
unfolding's places that corresponds to the marking M.
X: It is a vector containing all configurations of the
marking M.
See also: ORDER2, UNFOLDING_N, MCMILLAN, GRAPH1, GRAPH2, UNFOLDING_GRAPH, GRAPHMCM

****
****

UNFOLDING_GRAPH: this function returns the instructions to build the unfolding graph of a Petri net

****
##SYNTAX##
unfolding_graph(Pre,Post,M0,f) :

Input arguments:
Pre: Pre-incidence matrix of Petri net
Post: Post-incidence matrix of Petri net
M0: Initial marking
f: Number of places' tiers

See also: ORDER1, ORDER2, MCMILLAN, UNFOLDING_N, GRAPH1, GRAPH2, GRAPHMCM

****
****

UNFOLDING_N: this function returns the unfolding of a Petri net

****
##SYNTAX##

[Pre_u,Post_u,p_tier,t_tier,Pexp,Texp,t2]=unfolding_n(Pre,Post,M0,f) :

Input arguments:

Pre: Pre-incidence matrix of Petri net

Post: Post-incidence matrix of Petri net

M0: Initial marking of Petri net

f: Number of places' tiers

Output arguments:

Pre_u: Pre-incidence matrix of the unfolding net

Post_u: Post-incidence matrix of the unfolding net

p_tier: It is a cell array containing vectors of places' tiers

t_tier: It is a cell array containing vectors of transitions' tiers

Pexp: It is a vector containing all places of the unfolding net

Texp: It is a vector containing all transitions of the unfolding net

See also: ORDER1, ORDER2, MCMILLAN, GRAPH1, GRAPH2, UNFOLDING_GRAPH, GRAPHMCM

****
****