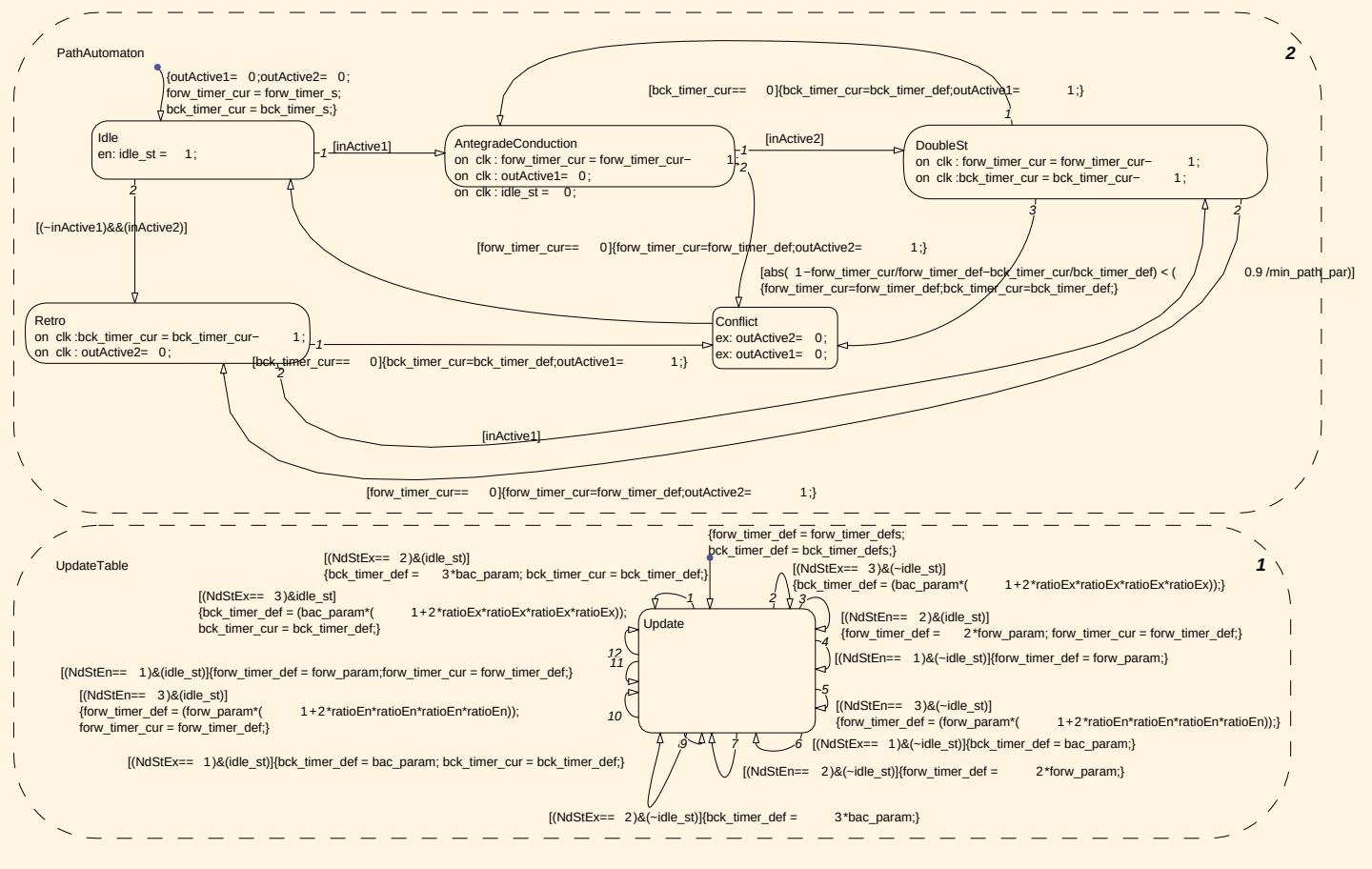
PathAutomaton
2
{outActive1= 0;outActive2= 0;
forw_timer_cur = forw_timer_s;
bck_timer_cur = bck_timer_s;}
Idle
en: idle_st = 1;
[inActive1]
AntegradeConduction
on clk : forw_timer_cur = forw_timer_cur−
on clk : outActive1= 0;
on clk : idle_st = 0;
[inActive2]
DoubleSt
on clk : forw_timer_cur = forw_timer_cur− 1;
on clk :bck_timer_cur = bck_timer_cur− 1;
[bck_timer_cur== 0]{bck_timer_cur=bck_timer_def;outActive1= 1;}
[(~inActive1)&&(inActive2)]
[forw_timer_cur== 0]{forw_timer_cur=forw_timer_def;outActive2= 1;}
[abs(1−forw_timer_cur/forw_timer_def−bck_timer_cur/bck_timer_def) < (0.9 /min_path_par)]
{forw_timer_cur=forw_timer_def;bck_timer_cur=bck_timer_def;}
Retro
on clk :bck_timer_cur = bck_timer_cur− 1;
on clk : outActive2= 0;
Conflict
ex: outActive2= 0;
ex: outActive1= 0;
[bck_timer_cur== 0]{bck_timer_cur=bck_timer_def;outActive1= 1;}
[inActive1]
[forw_timer_cur== 0]{forw_timer_cur=forw_timer_def;outActive2= 1;}
1
{forw_timer_def = forw_timer_defs;
bck_timer_def = bck_timer_defs;}
UpdateTable
[(NdStEx== 2)&(idle_st)]
{bck_timer_def = 3*bac_param; bck_timer_cur = bck_timer_def;}
[(NdStEx== 3)&(~idle_st)]
{bck_timer_def = (bac_param*(1+2*ratioEx*ratioEx*ratioEx*ratioEx));}
[(NdStEx== 3)&idle_st]
{bck_timer_def = (bac_param*(1+2*ratioEx*ratioEx*ratioEx*ratioEx));
bck_timer_cur = bck_timer_def;}
Update
[(NdStEn== 2)&(idle_st)]
{forw_timer_def = 2*forw_param; forw_timer_cur = forw_timer_def;}
[(NdStEn== 1)&(idle_st)]{forw_timer_def = forw_param;forw_timer_cur = forw_timer_def;}
[(NdStEn== 1)&(~idle_st)]{forw_timer_def = forw_param;}
[(NdStEn== 3)&(idle_st)]
{forw_timer_def = (forw_param*(1+2*ratioEn*ratioEn*ratioEn*ratioEn));
forw_timer_cur = forw_timer_def;}
[(NdStEn== 3)&(~idle_st)]
{forw_timer_def = (forw_param*(1+2*ratioEn*ratioEn*ratioEn*ratioEn));}
[(NdStEx== 1)&(idle_st)]{bck_timer_def = bac_param; bck_timer_cur = bck_timer_def;}
[(NdStEx== 1)&(~idle_st)]{bck_timer_def = bac_param;}
[(NdStEn== 2)&(~idle_st)]{forw_timer_def = 2*forw_param;}
[(NdStEx== 2)&(~idle_st)]{bck_timer_def = 3*bac_param;}