cmpr_1_2.xbe

Attributes

```
xbe name=cmpr_1_2 evaluate=yes limit_tstep=yes save_history=yes
# if x > x0, y1 = high, else low
# (reverse if flag_inverting=1)
# y2 = not(y1)
Jacobian: variable
input_vars: x
output_vars: y1 y2
aux_vars:
iparms:
+  flag_invert=0
+  flag_quad=0
sparms:
rparms:
+  x0=0
+  y_low=0
+  y_high=1
+  x_1=0
+  x_2=0
+  t_1=0
+  t_2=0
+  epsl=1.0e-6
+  delt_min=1.0e-6
+  delt_nrml=0.001
stparms:
igparms:
outparms: x y1 y2
```

Description

`cmpr_1_2.xbe` is a comparator with the following behaviour.

(a) `flag_invert` = 0:
`y1` = `y_high` if `x` > `x0`,
= `y_low` if `x` < `x0`.

(b) `flag_invert` = 1:
`y1` = `y_high` if `x` < `x0`,
= `y_low` if `x` > `x0`.

The other output `y2` is the complement of `y1`.
The parameters `delt_min`, `delt_nrml`, and `epsl` are used for controlling the simulator time steps. Additional time points are forced, depending on the values of `delt_min` and `delt_nrml`, when $|x - x_0| < \epsilon$. This feature allows accurate simulation without having to make the average time step very small. Generally, `delt_nrml` should be made equal to the typical simulator time step while `delt_min` should be made much smaller (say, by a factor of 100).

`flag_quad` decides the type of interpolation used to estimate the cross-over time (when x changes sign). If `flag_quad` is 0, linear interpolation is used; if it is 1, quadratic intepolation is used. For more details, see Ref. [1].

References

1. M.B. Patil, R.D. Korgaonkar, K. Appaiah, "GSEIM: A General-purpose Simulator with Explicit and Implicit Methods," submitted to Sādhanā, also available at `https://arxiv.org/abs/2104.06621`