

SmartSpice: Built-in Interconnect RC Network Reduction

Since version 4.0.1.R, SmartSpice has included a built-in interconnect RC network reduction capability. To enable this capability, option `int_rc_method` should be set to 1 or 2. This value selects the implementation algorithm. Depending on the value of `int_rc_method`, the RC networks around the grounded capacitors will be analyzed and the circuit topology will be changed.

Two additional options can be specified to control reduction.

The option `int_rc_cmin` specifies the capacitance threshold. All grounded capacitors with capacitance less than this threshold will be removed. The default value of `int_rc_cmin` is $1e-22$.

The option `int_rc_rmin` specifies the resistance threshold. Resistors with resistance less than the threshold will be reduced into a simplified equivalent network. The default value of `int_rc_rmin` is $1e-3$.

When the reduction feature is activated, SmartSpice will print out the following information while processing the input netlist with reduced equivalent interconnect RC networks:

```
Internal RC-reduction starting...
-----
Internal RC-reduction: detecting nodes with
grounded capacitors and number of connections
< 4 ...
Internal RC-reduction: grounded capacitors with
capacitance less than 1e-15 are : 5841
Internal RC-reduction: analyzing RC networks...
Internal RC-reduction: RC network chains: 5758
Internal RC-reduction: maximum length of RC
networks: 3
Internal RC-reduction: TOTAL STATISTICS :
-----
Internal RC-reduction: capacitors before : 62706
Internal RC-reduction: capacitors removed : 5697
Internal RC-reduction: capacitors reduction ratio
(%): 9.08525 %
Internal RC-reduction: resistors before : 100046
Internal RC-reduction: resistors removed : 815
Internal RC-reduction: resistors reduction ratio
(%): 0.814625
Internal RC-reduction: nodes before : 83101
Internal RC-reduction: nodes removed : 815
-----
```

Detailed Description of Two Algorithms

Common Part of Two Algorithms

The common part for two algorithms is detecting RC networks satisfying the specific predefined condition. First, SmartSpice starts by detecting nodes with grounded capacitors and a number of connections less than 4. Second, SmartSpice leaves only capacitors with capacitance greater than `int_rc_cmin`. Finally, SmartSpice detects all RC networks connected to such capacitors and links such networks in chain. In the previous output, the total count of such chains is reported as "Internal RC-reduction: RC network chains: 5758". The maximum length of chain is also reported. SmartSpice then traverses through all RC networks in a single chain and transforms the topology of all RC devices in the chain using the corresponding reduction technique.

Reduction Method 1

The first reduction method is defined by using the option `int_rc_method=1`. If it is specified, all capacitors in the chain will be removed. All internal nodes in the chain will then be analyzed and all internal resistors will be disconnected if possible. The final resistor with a connection to the external node of processing RC chain will be left, and its resistance will be adjusted to $R_{total} = \sum(R_i)$, where R_i – resistance of all internal serial resistors with resistance greater than `int_rc_rmin`. All unnecessary internal nodes with a number of connections equal to zero will then be removed. The final resistor will be connected to the external nodes of the processing chain. See Figures 1 and 2 for an illustrated description of this process in Example 1.

Reduction Method 2

The second reduction method is defined by selecting the option `int_rc_method=2`. First, after analyzing their resistance, all internal serial resistors will be removed and the new final resistor will be created with nodes connected to the external nodes of the processing RC chain. Second, the resistance R_{total} of the final resistor will be calculated based on all internal serial resistors.

After reduction, SmartSpice will report the final reduction statistics with the number of removed capacitors, resistors and nodes. If further basic reduction is not possible, SmartSpice will not change the topology.

Examples

Example 1

```
.option int_rc_method=1 int_rc_cmin=1e-18 int_rc_rmin=0.01
```

```
...
R1 Extnode1 Node1 0.001
C1 Node1 0 1e-19
R2 Node1 Node2 10K
C2 Node2 0 1e-19
R3 Node2 Extnode2 10K
C3 Extnode2 0 1e-19
...
```

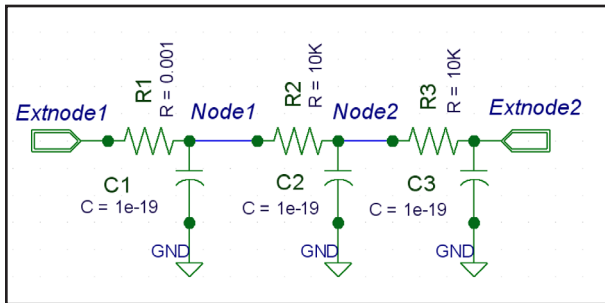


Figure 1. Example 1 – original block.

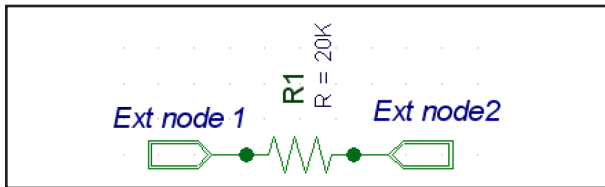


Figure 2. Example 1 – reduced block.

SmartSpice output:

```
Internal RC-reduction starting...
-----
Internal RC-reduction: detecting nodes with grounded capacitors and t_numconn < 4 ...
Internal RC-reduction: grounded capacitors with capacitance less than 1e-018 are: 3
Internal RC-reduction: analyzing RC networks...
Internal RC-reduction: RC network chains: 1
Internal RC-reduction: maximum length of RC networks: 3
Internal RC-reduction: TOTAL STATISTICS :
-----
Internal RC-reduction: capacitors before : 3
Internal RC-reduction: capacitors removed : 3
Internal RC-reduction: capacitors reduction ratio (%): 100
Internal RC-reduction: resistors before : 3
Internal RC-reduction: resistors removed : 2
Internal RC-reduction: resistors reduction ratio (%) : 67
Internal RC-reduction: nodes before : 5
Internal RC-reduction: nodes removed : 3
-----
```

Example 2

```
.option int_rc_method=2 int_rc_cmin=1e-18 int_rc_rmin=0.5
```

```
...
R1 Extnode1 Node1 1
C1 Node1 0 1e-19
R2 Node1 Node2 10K
C2 Node2 0 1e-19
R3 Node2 Extnode2 10K
C3 Extnode2 0 1e-19
...
```

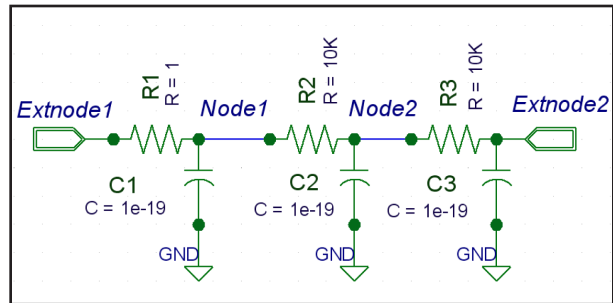


Figure 3. Example 2 (method 2) – original block.

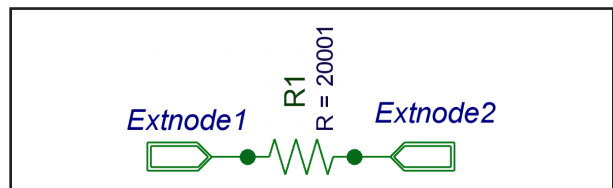


Figure 4. Example 2 (method 2) – reduced block.

Conclusion

The user will see a performance increase in model load and LU decomposition. The statement “.option acct=2” will output the number of final circuit equations. RC threshold enables the user to control the ratio of reduction. Larger thresholds will generate more compressed interconnect equivalent circuits. Breakdown may occur if excessively large thresholds are specified. SmartSpice does not give any recommendations for RC thresholds. The user must check simulation results after applying the RC reducer to see if the resulting simulation errors are acceptable.