Holger Vogt

17.02.2025

**Setting up the matrix**

Example: resistor, using Sparse 1.3 solver.

Set up a new matrix, to become ckt→CKTmatrix

ngspice.exe!NIinit(CKTcircuit \* ckt) Zeile 32

 unter D:\Spice\_general\ngspice-master\src\maths\ni\niinit.c (32)

ngspice.exe!**CKTsetup**(CKTcircuit \* ckt) Zeile 80

 unter D:\Spice\_general\ngspice-master\src\spicelib\analysis\cktsetup.c (80)

ngspice.exe!CKTdoJob(CKTcircuit \* ckt, int reset, TSKtask \* task) Zeile 167

 unter D:\Spice\_general\ngspice-master\src\spicelib\analysis\cktdojob.c (167)

ngspice.exe!if\_run(CKTcircuit \* ckt, char \* what, wordlist \* args, INPtables \* tab) Zeile 378

 unter D:\Spice\_general\ngspice-master\src\frontend\spiceif.c (378)

ngspice.exe!dosim(char \* what, wordlist \* wl) Zeile 341

 unter D:\Spice\_general\ngspice-master\src\frontend\runcoms.c (341)

ngspice.exe!com\_run(wordlist \* wl) Zeile 409

 unter D:\Spice\_general\ngspice-master\src\frontend\runcoms.c (409)

Create the matrix structure for Sparse 1.3, to become ckt→CKTmatrix→SPmatrix

ngspice.exe!spCreate(int Size, int Complex, int \* pError) Zeile 127

 unter D:\Spice\_general\ngspice-master\src\maths\sparse\spalloc.c (127)

ngspice.exe!SMPnewMatrix(sSMPmatrix \* Matrix, int size) Zeile 1118

 unter D:\Spice\_general\ngspice-master\src\maths\KLU\klusmp.c (1118)

ngspice.exe!NIinit(CKTcircuit \* ckt) Zeile 41

 unter D:\Spice\_general\ngspice-master\src\maths\ni\niinit.c (41)

ngspice.exe!**CKTsetup**(CKTcircuit \* ckt) Zeile 80

 unter D:\Spice\_general\ngspice-master\src\spicelib\analysis\cktsetup.c (80)

ngspice.exe!CKTdoJob(CKTcircuit \* ckt, int reset, TSKtask \* task) Zeile 167

 unter D:\Spice\_general\ngspice-master\src\spicelib\analysis\cktdojob.c (167)

ngspice.exe!if\_run(CKTcircuit \* ckt, char \* what, wordlist \* args, INPtables \* tab) Zeile 378

 unter D:\Spice\_general\ngspice-master\src\frontend\spiceif.c (378)

ngspice.exe!dosim(char \* what, wordlist \* wl) Zeile 341

 unter D:\Spice\_general\ngspice-master\src\frontend\runcoms.c (341)

ngspice.exe!com\_run(wordlist \* wl) Zeile 409

 unter D:\Spice\_general\ngspice-master\src\frontend\runcoms.c (409)

Fill the matrix structure with the resistive elements.

ngspice.exe!RESsetup(sSMPmatrix \* matrix, GENmodel \* inModel, CKTcircuit \* ckt, int \* state)

 unter D:\Spice\_general\ngspice-master\src\spicelib\devices\res\ressetup.c (72)

ngspice.exe!**CKTsetup**(CKTcircuit \* ckt)

 unter D:\Spice\_general\ngspice-master\src\spicelib\analysis\cktsetup.c (112)

ngspice.exe!CKTdoJob(CKTcircuit \* ckt, int reset, TSKtask \* task)

 unter D:\Spice\_general\ngspice-master\src\spicelib\analysis\cktdojob.c (167)

ngspice.exe!if\_run(CKTcircuit \* ckt, char \* what, wordlist \* args, INPtables \* tab)

 unter D:\Spice\_general\ngspice-master\src\frontend\spiceif.c (378)

ngspice.exe!dosim(char \* what, wordlist \* wl)

 unter D:\Spice\_general\ngspice-master\src\frontend\runcoms.c (341)

ngspice.exe!com\_run(wordlist \* wl)

 unter D:\Spice\_general\ngspice-master\src\frontend\runcoms.c (409)

Depending on the availability of a ckt→CKThead[i], a setup routine in the specific device with type i is run via DEVices[i]→DEVsetup().

TSTALLOC(RESposPosPtr, RESposNode, RESposNode);

from ressetup.c, line 72, is equivalent to

if((here-> RESposPosPtr = SMPmakeElt(matrix, here->RESposNode, here->RESposNode)) == NULL)

 return(E\_NOMEM);

SMPmakeElt () calls spGetElement() which translates external row and column numbers to internal ones by Translate() and then calls spcFindElementInCol(), which either returns the already existing matrix element or creates a new one. Due to searching through linked lists this function is a bottleneck when creating large matrixes.