Coupling of FEM Programs for Simulation of Complex Systems

B. Krevet, W. Kaboth

Forschungszentrum Karlsruhe
Institut für Mikrostrukturtechnik (IMT)
Postfach 3640, D-76021 Karlsruhe, Germany, berthold.krevet@imt.fzk.de

ABSTRACT

A method is presented which couples single task FEM programs, mesh generators and other utility programs to solve complex systems. By this method we are connecting the FEM programs with special Script Interpreter programs and use an overlaying Regie program. The controlling information is given to the Interpreter programs by script files. The power of this concept is demonstrated with several examples.

INTRODUCTION

In the calculation and optimization of the performance of Microsystems and their production processes, different kinds of physical quantities are involved. In most cases they are dependent on each other. Calculating all these quantities with one single finite element program leads to complicated programs and huge FEM meshes with a high degree of freedom at each node demanding a large amount of computer storage for the solution. To avoid these problems we present a concept which couples single task FEM programs for the quantities to be calculated, mesh generators and other utility programs. The FEM programs themselves remain unchanged in most cases. The computer storage is reduced, for only a few variables are involved at the same time. For each problem individual mesh generators and optimal solvers can be used.

METHOD

The coupling shown in figure 1 is done by associating to each FEM program and mesh generator an additional program called Script Interpreter. The main tasks of these Interpreter programs are:

- to change volume, surface and boundary data, to read input and result files from all FEM programs and to generate a new input file for the associated FEM program as indicated in figure 1.
- to set additional boundary conditions and forces and to change the execution control code (route code) for the associated FEM program.
  The corresponding values are written into files. When a FEM program is called and these files are present the data are added to the FEM database or replace the original FEM data.
- to perform an additional data transfer between the FEM programs. The node numbers, properties and the node coordinates of selected nodes or elements are written into special coupling files which are common to all Script Interpreters. Reading these files the programs connect the node or element numbers of their associated FEM - input files to the entries by geometric matching. The Script Interpreter gets all information from scripts which contain a lot of simple commands, e.g. 'multiply diel lower 5 with femres'. This command leads to multiply the dielectric constant 'diel' of those elements in which 'diel' is less than 5 with the solution for the electric potential. When an element property operates on a node property, the average value of all elements which are common to the node is used. When a node property acts on an element property the average value of all nodes belonging to the element is used.

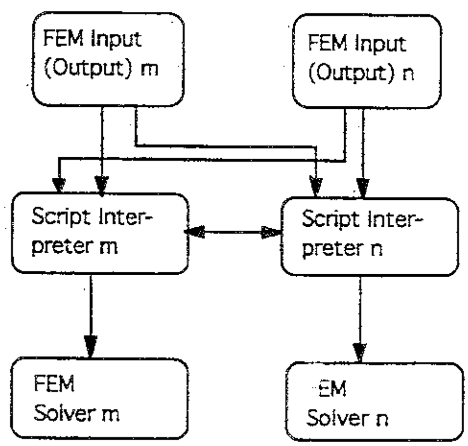


Figure 1. Coupling scheme of two FEM programs by Script Interpreter programs.

Complex calculations are done by successive calls to the FEM programs and their associated Interpreter programs by an overlaying Regie program within one execution sequence (step). For nonlinear problems this step has to be repeated. At each step the Regie program gets information of the calling sequences of the programs from its input file. To include time-dependent problems a defined time belongs to each step. The input file of the Regie program also contains a lot of scripts. With each call to an interpreter program the Regie program passes the corresponding script to it. Thus within each step the input file of any FEM program may be changed by the interpreter program depending on the results of previous steps. If a call to a FEM program is not needed any more, but it is in the execution list, the Script Interpreter can pass for it a 'pause' statement to the Regie program. Using this scheme, we connected several two dimensional programs for mechanical and electrical problems based on programs given by Schwarz [1], a program for diffusion analysis based on a program given by Pepper [2] and programs developed by the authors. These programs include two dimensional flow analysis, three dimensional shell and solid structural analysis and three dimensional magnetic field calculation [3]. The inclusion of the electromagnetic field solver Tosca [4] is under progress. For model generation the preprocessor of Tosca is used.

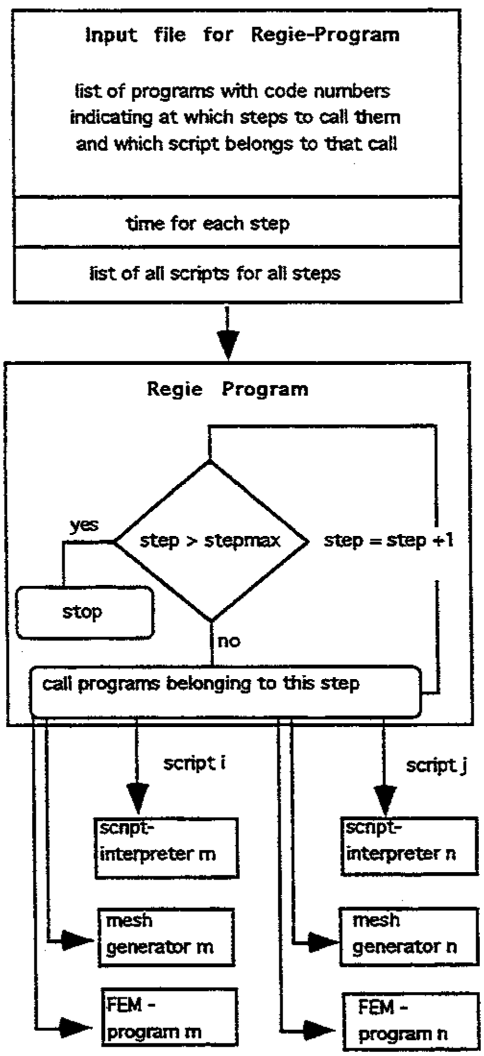


Figure 2. Coupling scheme of the FEM programs with the overlaying Regie program.

At the step shown in figure 2, two FEM programs are called with the according Script Interpreter and mesh generator. The calling sequence is from top to bottom.

APPLICATIONS

Controlling optimization procedures.

Figure 3 shows a clamp with an applied force in x direction. The clamp is fixed at the left wall. The stress may be reduced by changing its geometry. This has been done using a method described in [5] in which the stress is transformed to a fictive temperature increase. Due to the thermal expansion the clamp swells proportional to the stress. The input file for the Regie program to control this optimization procedure is listed in table 1. The program 'Scheibe' is called with alternating tasks. In one execution step it calculates the mechanical stresses and in the next step it calculates the thermal expansion, getting the temperature increase passed to it by the Script Interpreter. The two integers following the program name in the input file indicates in which steps it must be called. The next pair (n1, n2) of integer shows that script n2 must be passed n1 times to the program. During this optimization process different nodes may be fixed by means of the script files. In the example shown in figure 3 all other nodes are displaced during the first 20 execution steps (script 1). From step 21 to 31 only nodes are displaced which are within the rectangle shown in figure 3a (script 2). Figure 4 shows the stress reduction of all execution steps. After 31 steps the large deformation of the clamp results in a reduction factor of about 8.

Table 1. Script for mechanical stress reduction.

```
program
scheibe          1,31
script_scheibe  1,31  ;20,1;11,2
end program

time     ;10,0.1;2,0.2
end time

script 1
optimize mises
set strength 0.01
select all elements
pin nodes
457 ux
312 ux uy
0
stoprun 0.001 0.001
end script 1

script 2
optimize mises
set strength 0.01
select elements
239
246
:
0
```

```
pin nodes
457 ux
398 uy
:
0
stoprun 0.001 0.001
end script 2
```

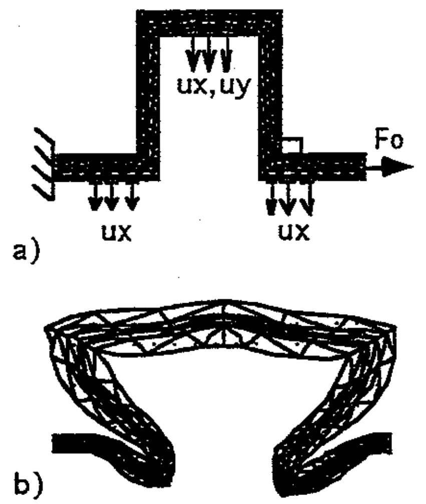


Figure 3. Deformation of a clamp to reduce mechanical stress. The clamp is fixed at the left wall and a force in x - direction is applied. For the nodes indicated with arrows in figure 3a the displacements in x (y) -direction are set to zero.

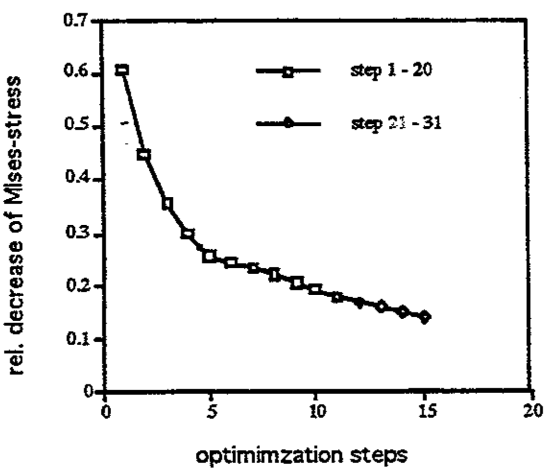


Figure 4. Stress reduction for the optimization procedure shown in figure 3.

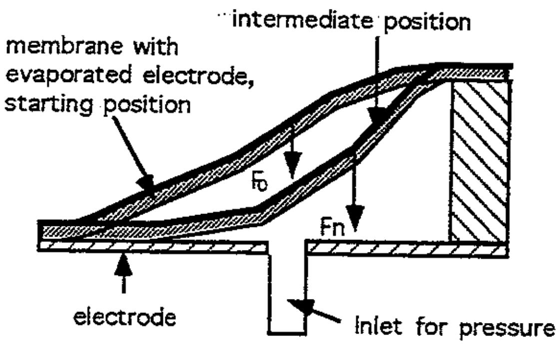
Coupling of different FEM Programs and mesh generators.

Figure 5 shows a membrane electrode which is deflected by electrical forces. This membrane is part of an electrostatic micro valve manufactured in IMT [6]. The ground plate with the pressure inlet forms the other electrode. For this problem two FEM programs are needed, one to solve the electrical field problem, the other to calculate the deflection of the membrane. The input file for the Regie program is listed in table 2.

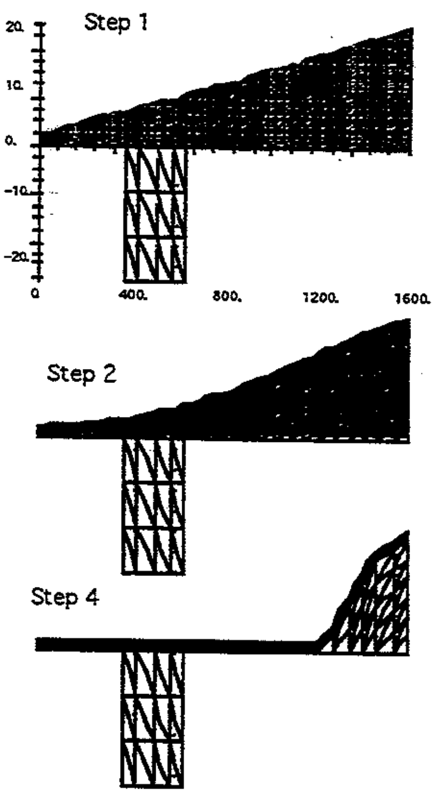
Table 2. Input file for the deflection of a plate in an electric field.

```
program
script_feld_2d  1,6  ; 5,1 ; 1,2
feldmesh2       2,6
feld_2d         1,6
script_plate3b  6,6  ; 1,3
genmem          1,6
plate3b         1,6
end program

script 1
couple pot 10.
end script 1

script 2
reset parameter
couple pot 10
enable check force
disable calc dforce
set parameter 5 1
set pressure 1
enable create constrain 3
end script 2

script 3
ramp epslinear
set parameter  6 1
end script 3
```

First the electrical forces are calculated for the starting position, then a mesh for the membrane is generated and the deflection is determined. The resulting new position of the membrane requires a mesh generator to produce a new mesh for the electrostatic solver to give the new forces. This alternating calling of the two FEM programs and the mesh generators have to be repeated. Figure 5b shows the membrane at three different steps. When convergence is achieved script 2 sets the pressure to the inlet and initiates constrain conditions to a contact element between the lower electrode and the membrane. Finally script 3 causes the program plate3b to run a nonlinear deflection analysis to calculate the threshold pressure for opening the valve. It is determined by this value of the pressure when the contact just opens. This state is unstable and leads to a complete deflection of the membrane because the electrical forces are getting smaller when the membrane leaves the lower electrode. Figure 6 shows the threshold pressure as function of the applied voltage for different inlet shapes.



a)



b)

Figure 5. a): Scheme of the valve
b): FEM mesh and membrane positions for successive calling steps

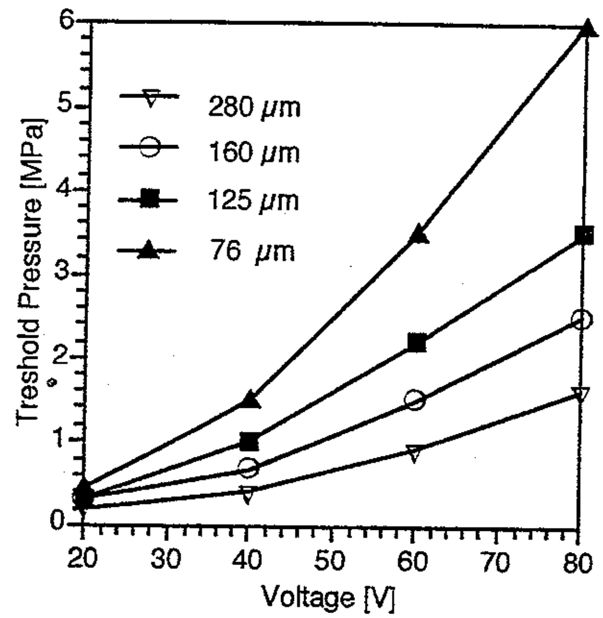


Figure 6. Pressure for opening the valve versus the voltage for different shapes of the slot .

Improvement of the performance of individual FEM Programs.

The performance of a single FEM program can be considerably increased by the use of the Script Interpreters. Figure 7 shows the starting conditions of a simple model for the development of PMMA with concentration C2 = 1 by a developer with concentration C1 after exposure with synchrotron radiation. We solve this problem by using a two-dimensional FEM program for the diffusion analysis of one component. The FEM program solves:

$$\frac{\partial C1}{\partial t} = \nabla K \nabla C1 + const*C1$$

In addition the Script Interpreter solves:

$$\frac{\partial C2}{\partial t} = const * C1$$

and calculates the source constant 'const' and the diffusion constant 'K':

$$const = f1(x,y) * C2; \quad K = f2(C1,C2)$$

The function f1(x,y) describes the radiation damage in the PMMA, the function f2(C1,C2) provides a continuos connection of the diffusion coefficient in the undeveloped PMMA (C2=1) and in those regions where C2 = 0. The input file for the Regie program is listed in table 3. The first script is used to set up the initial conditions, the second script performs the calculation shown above. The time dependence of concentration C2 is shown in figure 8 for two functions f1(x,y) .

Table 3. Input file for a PMMA solution model.

```
program
fem_2d          2,10
script_fem_2d  1,10  ;1,1;9,2
end program

time  ;10,0.25
end time

script 1
function 1
A=-50.5
B=1
func=A*(X-2.5)**2/6.25-1.5
end function 1
replace hgen lower 0.0 with function 1
select nodes with hgen lower 0
save hgen
create n_var 1 1.0
select nodes with hgen=0.0
set femres 1
end script 1

script 2
function 1
A=-50.5
func=A*(X-2.5)**2/6.25-1.5
end function 1
function 2
X=n_var 1
A=10
B=2
func=A-(A-B)*X
end function 2
restore n_var
set realc 1 0.25
set realc 2 0.25
operate ifunc 1 on cnew
multi res_n=all with function 1
operate ifunc 6 on n_var 1
save n_var
restore hgen
multiply hgen lower 0 with n_var 1
replace diffx lower 10 with function 2
diffx=diffy
end script 2
```

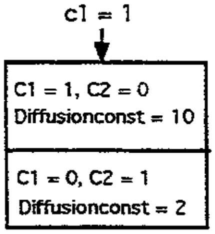


Figure 7. Starting condition of the development model. The developer concentration C1 on the top is fixed to 1.

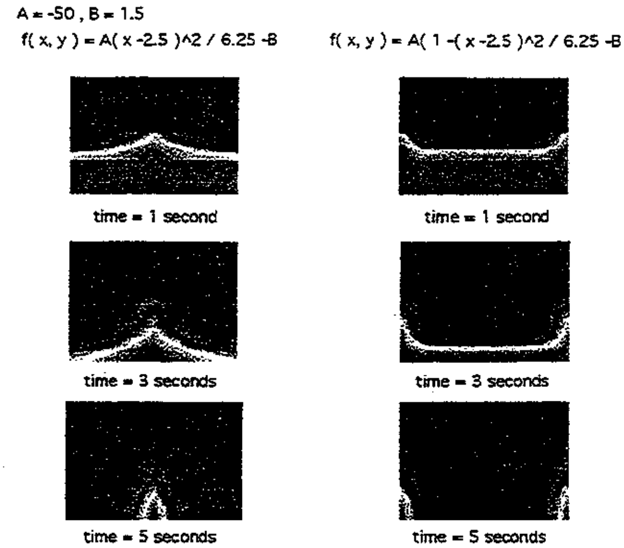


Figure 8. Time dependence of the resist concentration C2 for two different functions of the radiaten damage.

CONCLUSION

The procedure described above is applicable to a wide range of problems involving many physical variables. It is not restricted to the use of FEM programs, also boundary element programs or direct analysis programs may be included. Only small changes for the programs are needed. Proved script files could be used as macros for more complicated calculations. We plan to extend the concept by including the possibilitiy to exchange complete parts of the mesh between FEM programs and by adding logical commands to the Script Interpreter programs.

REFERENCES

1 H. R. Schwarz, "Fortran-Programme zur Methode der finiten Elemente", B. G. Teubner Stuttgart, 1991.

2 D. W. Pepper, J. C. Heinrich, "The Finite Element Method", Hemisphere Publishing Corporation, 1992.

3 B. Krevet, IEEE Transactions on Magnetics, Vol. 28, No.2, 1060-1063, 1992.

4 Vector Fields ltd., 24 Bankside, Kidlington, Oxford OX51JE, England.

5 C. Matthek, "Design in Nature", Interdisciplinary Science Reviews, Vol. 19, No.4, 298-314.

6 C.Goll, W. Bacher, B. Büstgen, D. Maas, R. Ruprecht, Proc. MME96, Barcelona, 231-234, 1996.