

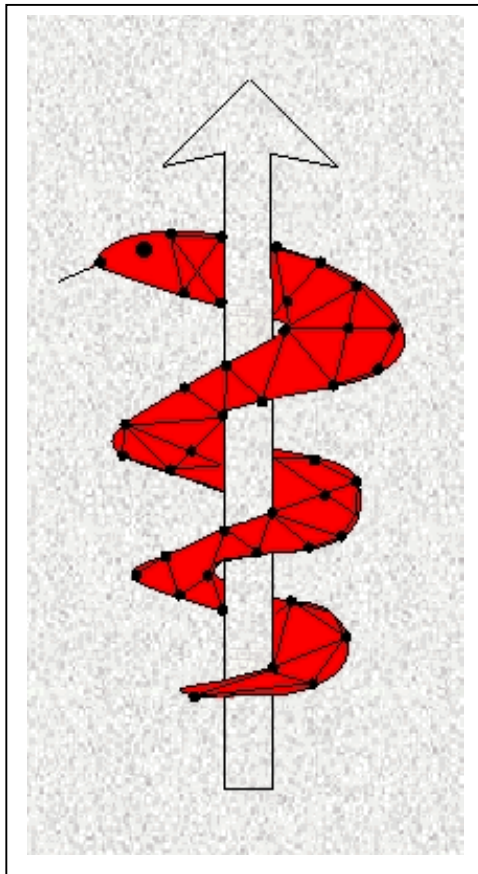


The IST Programme Project No. 10378

SimBio

SimBio - A Generic Environment for Bio-numerical Simulation

<http://www.simbio.de>



Deliverable D3a Numerical Solution System Design Report

Status: Final
Version: 1.0
Security: Public

Responsible: NEC
Authoring Partners: NEC, MPI, ESI

Release History

Version	Date
0.1	31.08.00
1.0	29.09.00

The SimBio Consortium :

NEC Europe Ltd. – UK
A.N.T. Software – The Netherlands
K.U. Leuven R&D – Belgium
ESI Group – France
Smith & Nephew - UK

MPI of Cognitive Neuroscience – Germany
Biomagnetisches Zentrum Jena – Germany
CNRS-DR18 – France
Sheffield University – UK

1. INTRODUCTION	2
2. DRAMA.....	3
2.1. INTRODUCTION	3
2.2. IMPLEMENTATION	3
2.2.1 <i>Current Status</i>	3
2.2.2 <i>Interaction with other SimBio Components</i>	4
2.2.3 <i>Preliminary Results</i>	5
2.3 TECHNICAL DEVELOPMENT PLAN.....	8
3. AZTEC AND PEBBLES.....	8
3.1 THE AZTEC LIBRARY	8
3.1.1 <i>Introduction</i>	8
3.1.2 <i>Implementation</i>	9
3.1.3 <i>Technical Development Plan</i>	9
3.2 AMG-PRECONDITIONED CG AND PEBBLES-SOLVER LIBRARY	9
3.2.1 <i>Introduction</i>	9
3.2.2 <i>Implementation</i>	9
3.2.3 <i>Technical Development Plan</i>	10
4. NEC SOLVER LIBRARY PILUTS.....	10
4.1. INTRODUCTION	10
4.2. IMPLEMENTATION	10
4.2.1 <i>Current Status</i>	10
4.2.2 <i>Interaction with other SimBio Components</i>	10
4.2.3 <i>Preliminary Results</i>	11
4.3 TECHNICAL DEVELOPMENT PLAN.....	12
5. S-CAUCHY	12
5.1 INTRODUCTION	12
5.2 IMPLEMENTATION	13
5.2.1 <i>Current Status</i>	13
5.2.2 <i>Interaction with other SimBio Components</i>	13
5.2.3 <i>Preliminary Results</i>	15
5.3 TECHNICAL DEVELOPMENT PLAN.....	16
6. PAMSAFE.....	17
6.1. INTRODUCTION	17
6.2. IMPLEMENTATION	17
6.2.1 <i>Current Status</i>	17
6.2.2 <i>Interaction with other SimBio components</i>	17
6.2.3 <i>Preliminary Results</i>	18
6.3 TECHNICAL DEVELOPMENT PLAN.....	18
6.3.1 <i>GENERIS™</i>	18
6.3.2 <i>PAM-SAFE™ Solver</i>	18
6.3.3 <i>Converters</i>	19
7. HEAD-FEM	19
7.1 INTRODUCTION	19
7.2 IMPLEMENTATION	20
7.2.1 <i>Current Status</i>	20
7.2.2 <i>Interaction with other SimBio Components</i>	20
7.2.3 <i>Preliminary Results</i>	20
7.3 TECHNICAL DEVELOPMENT PLAN.....	21
REFERENCES	22

1. Introduction

This workpackage plays an important role within the SimBio project as it represents a core module of the generic environment. Figure 1 shows the central position of the numerical solution system in the project. Input for the solution system are distributed or sequential FE meshes generated by the software developed in Subtask 1.2 (see D1.2a). The application desired by the SimBio user drives the selection of the mesh type that has to be provided by Subtask 1.2.

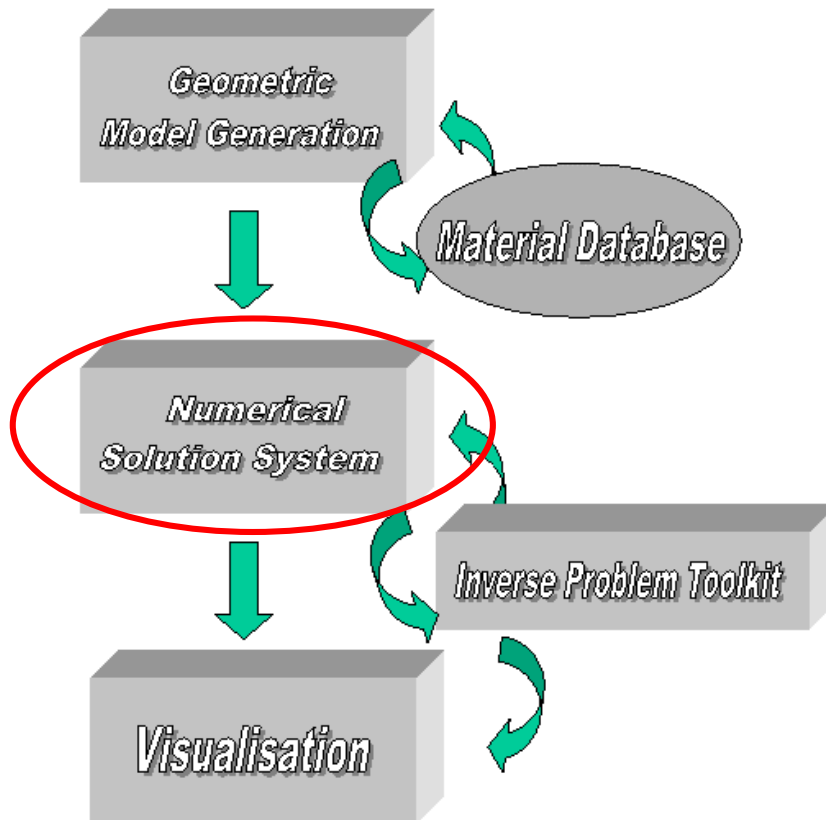


Figure 1: This diagram shows the position of the numerical solution system in the SimBio project.

The numerical solution system contains tools enabling efficient equation solving on parallel HPC systems (1.-3. in the list below) as well as full finite element codes usable for simulation studies on human organs (4.-6. in the list below). Sequential execution is naturally included within parallel solvers and FE methods. The numerical solution system (NSS) component additionally includes an interface to enable the integration of external commercial simulation software.

As the SimBio environment is intended to fulfill the requirements imposed by a wide range of medical problems, its solution system shows a large assortment of numerical recipes. The following is a list of numerical tools that comprise the SimBio solver system. A detailed explanation of each module is given in the remainder of this deliverable. The NSS comprises:

I. Linear Solver Libraries/Partitioning Tool

1. **DRAMA**, a tool for partitioning finite element meshes for efficient parallel execution.
2. **AZTEC/PEBBLES**, public domain libraries for parallel equation solving.
3. **PILUTS**, a highly efficient standalone NEC solver library (**P**arallel **I**ncomplete **L**U with **T**hreshold preconditioned **S**olvers). The routines can be executed as parallel standalone tools or can be called as parallel subprograms in an application code parallelised by domain partitioning.

II. Full Problem Solvers

4. **CAUCHY**, a full FE code for sequential execution with coupling for parallel solvers provided.
5. An interface for **PAMSAFE**, a fully non-linear FE code for biomechanical applications.
6. **HEAD-FEM**, a linear fully parallel FE code for special biomechanical applications.

It should be noted that the future SimBio user does not need to have detailed knowledge about the functionality and the handling of all the different codes available in this workpackage. He will be guided by a graphical user interface that chooses the software tools needed for a specific task and couples them correctly (see D6a).

The remainder of this deliverable is organised as follows. For each of the six components listed above

- a short introduction to the medical/mathematical problem addressed by the software is given,
- the status of implementation of the software is illustrated by,
 - describing the functionality of its current version,
 - explaining the interaction with other SimBio software modules,
 - presenting, whenever possible, preliminary results obtained with the current version
- and the plans for the further technical developments are presented.

2. DRAMA

2.1. Introduction

The DRAMA library supports dynamic load balancing for parallel message-passing mesh-based simulation codes. Although the library was originally developed for applications with dynamic, solution-adaptive mechanisms, it can equally well be applied to static problems. Based on a very general cost model, DRAMA provides access to a range of parallel partitioning algorithms through a simple and effective mesh-based interface.

2.2. Implementation

2.2.1 Current Status

A repartitioner tool that provides load-balancing, matrix partitioning and basic data-migration capabilities has been implemented (F90) based on the DRAMA library. The tool consists of the following subroutines:

```
drama_simbio_part
├── read_drama_mesh
├── repartition
│   ├── read DRAMA options
│   ├── DRAMA_INIT
│   ├── set DRAMA options
│   ├── cost function evaluation
│   ├── DRAMA geometric/graph/mesh migration
│   ├── cost function evaluation
│   ├── migrate data
│   ├── write_pmvis_mesh
│   └── DRAMA_FINALIZE
└── write_cauchy_mesh
```

The DRAMA library has been extended to give special support to the SimBio applications for the partitioning of sparse symmetric matrices. The use of two additional methods, a DRAMA-internal optimised node/element ordering and MeTis nested dissection, have been added to enable efficient parallel iterative solvers and preconditioners. We will give an example that shows that the SimBio solvers take advantage of the new functionality.

2.2.2 Interaction with other SimBio Components

The present version of the repartitioner tool uses file interfaces based on a common ASCII file format definition. It can be linked with VGrid, S-CAUCHY and the PILUTS solver in a linear chain (see Figure 7).

The ASCII (input) mesh file format starts with a two line header followed by the node list and the element list (one per line):

```
! Header
myid size
dim1 dim2 dim3 dim4 dim5 dim6
! Node list
(influence_id x y z)_i          i=1,dim4
! Element list
(type nnode ((l,id)_j, j=1,nnode)_i  i=1,dim1
```

The partitioner tool outputs 3 files:

ca_infl.kno	influence node (see Section 5.1 for explanation) file read by CAUCHY
ca_main.dst	partitioning information read by PILUTS solver
ca_main.geo	geometry file for CAUCHY

The following example shows the format of the influence node file for a simple case of a single influence node:

BOI - KNOTENFILE

```
=====
=====
```

BOI - KNOTENNUMMERNKARTE

3586

The second file containing the partitioning information for 8 processors of the NEC LAMP system (machine names: berlin, bonn, dortmund and dresden) has the format:

```
8
0 berlin
1 berlin
2 bonn
3 bonn
4 dortmund
```

```

5 dortmund
6 dresden
7 dresden
0 8920 17852 26780 35706 44635 53557 62480 71403
0 8156 16396 24819 33228 41347 49556 57343 65360

```

The last two lines describe the distribution of nodes and elements to the sub-domains (processors) in the form of a node and an element pointer.

The geometry file consists of a header, a node (co-ordinate) and an element (connectivity) section:

BOI - GEOMETRIEFILE

BOI - STEUERKARTE

```

ID : 23642067
ANZAHL DER KNOTEN : 71403
ANZAHL DER ELEMENTE : 65360
GEOMETR. STRUKTUR - DIMENSION : 3

```

EOI - STEUERKARTE

BOI - KOORDINATENKARTE

```
(x y z)_i i=1, 71403
```

EOI - KOORDINATENKARTE

BOI - ELEMENTKNOTENKARTE

```
(type: 11 12 13 14 15 16 17 18)_i i=1, 65360
```

EOI - ELEMENTKNOTENKARTE

EOI - GEOMETRIEFILE

Note that the geometry file requires the partitioner tool to convert from DRAMA double numbering (local number, processor number) to global node numbering.

2.2.3 Preliminary Results

To demonstrate the efficiency of the approach we show preliminary results coupling the mesh generator VGrid (see D1.2a) with the DRAMA partitioner tool and the S-CAUCHY (see Chapter 5) application that calls the solver with two different matrix partitionings. The first partition is suitable for a parallel solver while the second one is designed for a sequential direct solver.

As discussed in D1.2a the original node numbering of VGrid meshes is not optimal but it is satisfactory as Figure 2 and Table 1 reveal. Thus, the user has to decide whether or not repartitioning and/or renumbering is appropriate for their application.

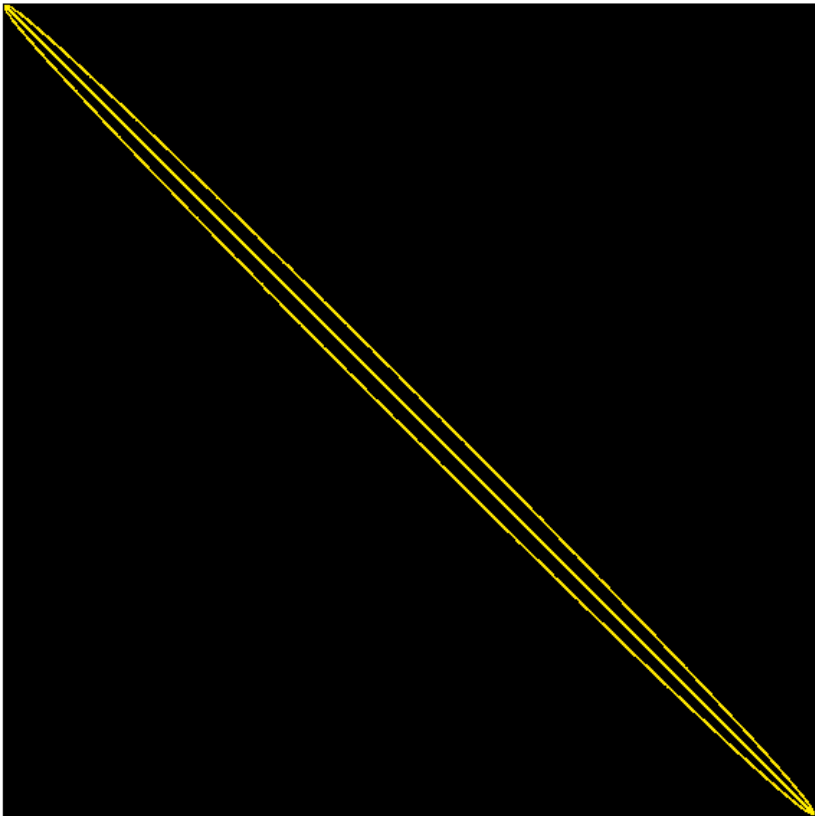


Figure 2: Position of the non-zero entries in the original matrix (based on a VGrid mesh).

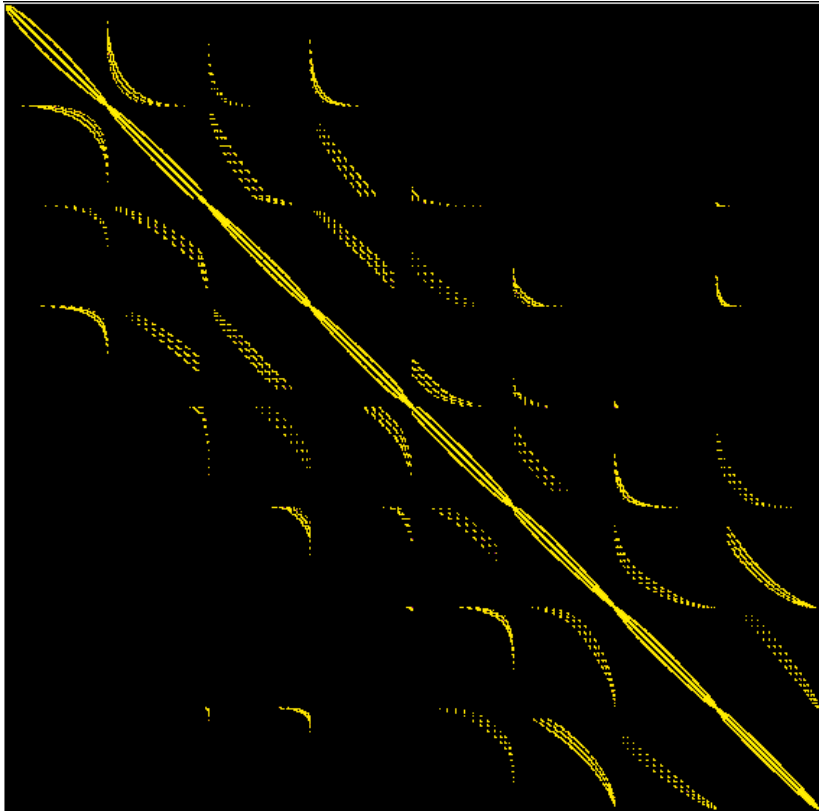


Figure 3: Position of the non-zero entries in the DRAMA (PJOSTLE) partitioned matrix.

Matrix partition	iterations	Execution time [s]	Total matrix fill-in
Original	166	16.18	2287475
DRAMA	108	10.56	2685171

Table 1: Comparison of the CG/ILDLT solver performance on 8 processors of a NEC Cenju-4 system with the original (Figure 2) and the DRAMA (PJOSTLE) matrix pattern (for eight processors) shown in Figure 3. The threshold value was 0.001. The order of the matrix was 71403 with 1402157 non-zero entries.

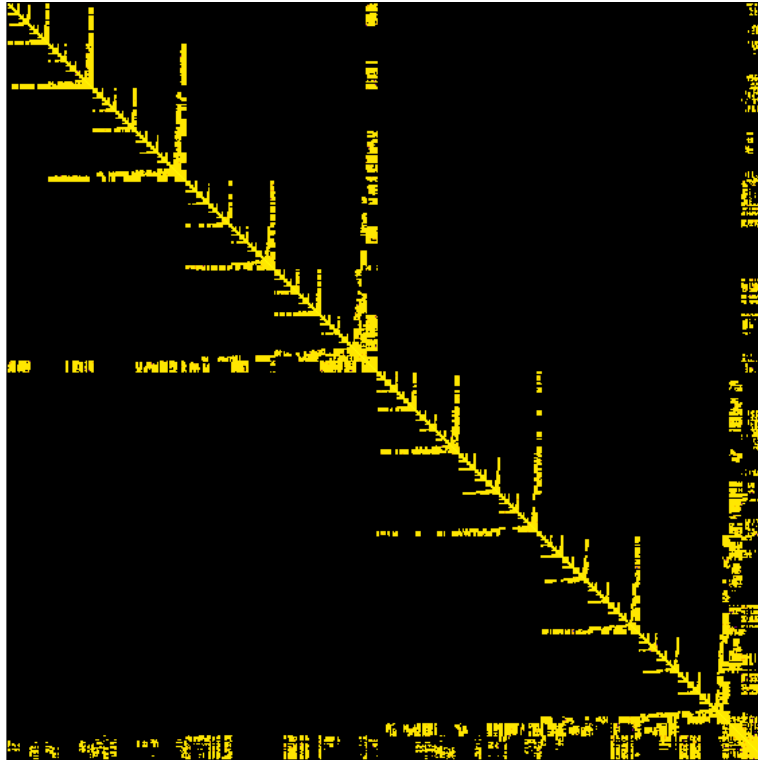


Figure 4: Position of the non-zero entries in the DRAMA (nested dissection) partitioned matrix.

Matrix Partition	Iterations	Execution time [s]	Total matrix fill-in
original	0	36.6	121710676
DRAMA	0	8.6	35527748

Table 2: Comparison of the QMR/ILDLT solver performance on 1 processor of a NEC Cenju-4 system with the original and the DRAMA (MeTis nested dissection) partition shown in Figure 4. The times given are for forward/back substitution only. The threshold value was 0.0 (equivalent to a direct method). The order of the matrix was 71403 with 1402157 non-zero entries.

Tables 1 and 2 show that in both cases the performance of the solver is improved when using the DRAMA partition.

Figure 5 depicts the subdivision of a typical SimBio mesh (stemming from VGrid) representing the human head. The mesh has been partitioned with the JOSTLE option of the DRAMA tool and enables the efficient execution of parallel applications on 16 processors.

2.3 Technical Development Plan

To enhance the coupling between the applications it is planned to translate the present ASCII file format to a universal platform independent binary form (VISTA file format). The binary format is more compact and will reduce the communication overhead.

If required by the applications a fine tuning of the node ordering can be implemented in the DRAMA library to enhance the partitioning quality.

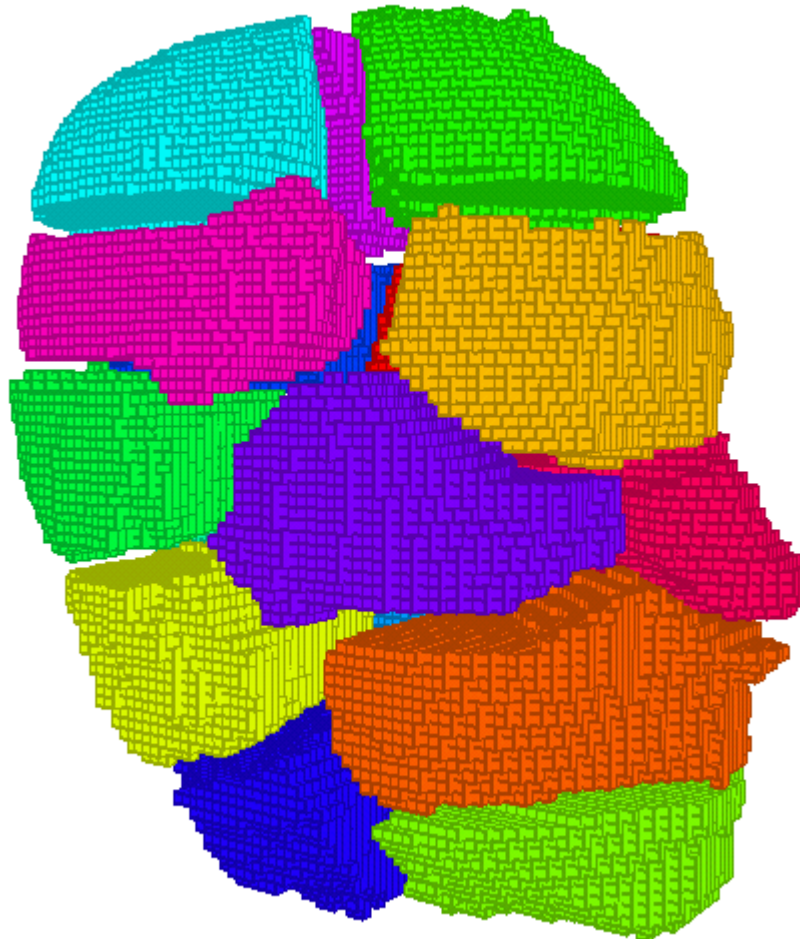


Figure 5: Visualisation of 16 partitions of a hexahedral head mesh obtained with the DRAMA tool.

3. AZTEC and PEBBLES

3.1 The AZTEC Library

3.1.1 Introduction

AZTEC is a massively parallel iterative solver library for solving sparse linear systems [aztec]. The library has been developed by the Sandia National Laboratories, USA. AZTEC is parallelised on the basis of the Message Passing Interface (MPI) library.

Many important scientific and engineering applications require the use of linear solvers. The AZTEC iterative solver package grew out of a specific application: modelling reacting flows. Its primary goal has been to provide state-of-the-art iterative methods that perform well on parallel computers (applications of over 200 Gflops have been achieved on the Sandia-Intel Tflop Computer) and at the

same time are easy to use for application engineers. In addition to providing standard iterative methods to engineers, the AZTEC library is also used for research on preconditioners.

3.1.2. Implementation

3.1.2.1 Current Status

AZTEC is a library for solving linear systems, which is both easy-to-use and efficient. Simplicity is attained using the notion of a global distributed matrix. The global distributed matrix allows a user to specify pieces (different rows for different processors) of his application matrix exactly as he would in the serial setting (i.e. using a global numbering scheme). Issues such as local numbering, ghost variables, and messages are ignored by the user and are instead computed by an automated transformation function. Efficiency is achieved using standard distributed memory techniques; locally numbered submatrices, ghost variables, and message information computed by the transformation function are maintained by each processor so that local calculations and communication of data dependencies is fast. Additionally, AZTEC can take advantage of advanced partitioning techniques (e.g. from DRAMA) and utilizes efficient dense matrix algorithms when solving block sparse matrices.

Methods implemented in AZTEC are: CG, CGS, BiCGSTAB, GMRES, TFQMR

Preconditioners: Point & block Jacobi, Gauss-Seidel, least-squares polynomials, and overlapping domain decomposition using sparse LU, ILU, BILU within domains.

3.1.2.2 Interaction with other SimBio Components

AZTEC has been used for the parallelisation of the HEAD-FEM code. Chapter 6 presents more details about the interaction between HEAD-FEM and AZTEC.

3.1.3 Technical Development Plan

No further developments of AZTEC are planned within the SimBio project.

3. 2 AMG-preconditioned CG and PEBBLES-Solver Library

3.2.1. Introduction

Large sparse systems of linear equations arising from the FE method have to be solved efficiently, in particular whenever repeated solutions with different right-hand-sides are required (e.g. in WP4.1. and ST7.1.). A robust method that provides fast convergence for a wide class of “real life” applications is to use multigrid techniques for preconditioning the conjugate gradient method [jun91]. A pure algebraic multigrid (AMG) preconditioner will be applied within SimBio because this method is the optimal choice for the evaluation application “source localisation” (see D7.1a for further information).

3.2.2. Implementation

3.2.2.1 Current status

PEBBLES (**P**arallel and **E**lement **B**ased gray **B**ox **L**inear **E**quation **S**olver) is a public domain solver library developed at the J.Kepler University Linz, Austria [rei00]. PEBBLES focuses on algebraic multigrid methods for symmetric positive definite systems of linear equations arising from elliptic partial differential equations. Thus, PEBBLES provides optimal methods for SimBio’s source localisation application. The AMG is especially used as a preconditioner for the CG method.

3.2.2.2 Interaction with other SimBio Components

The AMG preconditioned CG solver from the PEBBLES package has been integrated in SimBio’s numerical solution system and coupled to S-CAUCHY (see Chapter 5 of this design report). For the one-processor performance tests, the coupling between this two modules is a “loose file coupling”, i.e. the FE equation system is generated by S-CAUCHY and written out to a file. This binary file is finally read by the PEBBLES software for AMG CG performance tests.

3.2.2.3 Preliminary Results

First AMG CG performance tests for one processor and a comparison between the convergence behaviour of the different solver strategies described in Chapter 4 have been carried out. The results of this comparative study will be presented in Chapter 5 of this design report.

3.2.3 Technical Development Plan

A test version of a parallel AMG preconditioned CG solver based on a data decomposition of the mesh designed for distributed memory computers [Haa00] is available in PEBBLES. This version will be integrated into the NSS for further performance tests and comparisons with parallelised incomplete factorisation preconditioned CG solvers (see Chapter 4). In this case, S-CAUCHY calls the PEBBLES solver during the construction of the geometry matrix element by element.

4. NEC Solver Library PILUTS

4.1. Introduction

The parallel sparse solver library PILUTS comprises the iterative methods CG, symmetric QMR, and BiCGstab [saad96]. For preconditioning, incomplete factorisations with threshold combined with diagonal or row and column scaling are applied [saad99]. For symmetric, positive definite matrices, CG preconditioned by an incomplete Cholesky decomposition (LLT) with threshold or by an incomplete LDLT factorisation with threshold is suited. A symmetric QMR version with incomplete LDLT factorisation with threshold preconditioning solves general symmetric problems, BiCGstab preconditioned by an incomplete LU decomposition with threshold equation systems with non-symmetric matrix.

4.2. Implementation

4.2.1 Current Status

The solvers and preconditioners are parallelised by matrix and vector partitioning. Each processor obtains a block of matrix rows and vector segments which correspond to the nodes of a subdomain computed by a domain partitioning tool. The scheme for necessary communication is automatically derived from the sparsity pattern of the matrix [bas96], [bas99]. Local computations and data transfers are overlapped to reduce waiting times. A special feature of the parallel solver developed is a block reorganization of the matrix data [bas96], [bas99]. The number of blocks corresponds to the number of processors used. For a row-wise matrix-vector operation, each block only requires access to the vector components of one processor. This block structure makes an efficient overlap of computations and communication possible and additionally supports cache exploitation. As opposed to many other parallel iterative solvers the symmetric QMR and the CG variant developed require only one global communication per iteration since all vector-vector reductions are grouped [bas00].

In the parallel case, the incomplete LLT, LDLT or LU decompositions are performed per subdomain, i.e., per local diagonal block of the matrix. For this purpose, the block reorganisation of the matrix data is exploited. Thus, both preconditioner construction and application are totally parallel. The disadvantage of these block preconditioners is that with increasing number of subdomains the preconditioning effect usually decreases since couplings between subdomains are neglected. On the other hand, preconditioner construction and application usually gets cheaper with increasing number of subdomains if the same threshold value is used [bas00].

4.2.2 Interaction with other SimBio Components

The PILUTS solvers have been coupled with the DRAMA tool (see Chapter 2) and the application code S-CAUCHY (see Chapter 5). In this environment, parallel stand-alone versions of the solvers are used. The equation system data and the partitioning information is given to the solvers via files. First results of the successful coupling can be found in Chapters 2 and 5.

Moreover, the integration of the solvers into the biomechanics code HEAD-FEM is planned (see Chapter 7). Since HEAD-FEM is parallelised by domain partitioning the call of the solvers as parallel subroutines is best suited.

4.2.3 Preliminary Results

Figure 6 displays the time and scaling behaviour of the parallel PILUTS solvers for one forward solution within S-CAUCHY (isotropic problem, symmetric matrix of order 71403 with 1402157 non-zeros) on up to 64 processors of an NEC Cenju-4 parallel system. A threshold of 10^{-2} was used, the iteration was stopped if the residual norm divided by the initial norm was less than 10^{-10} . Diagonal scaling, incomplete LU or LDLT with threshold (ILUT or ILDLT) are compared for preconditioning. Furthermore, the effect of Reverse Cuthill-McKee (RCM) ordering of the matrix was tested. Note the log10 scale on the y-axis.

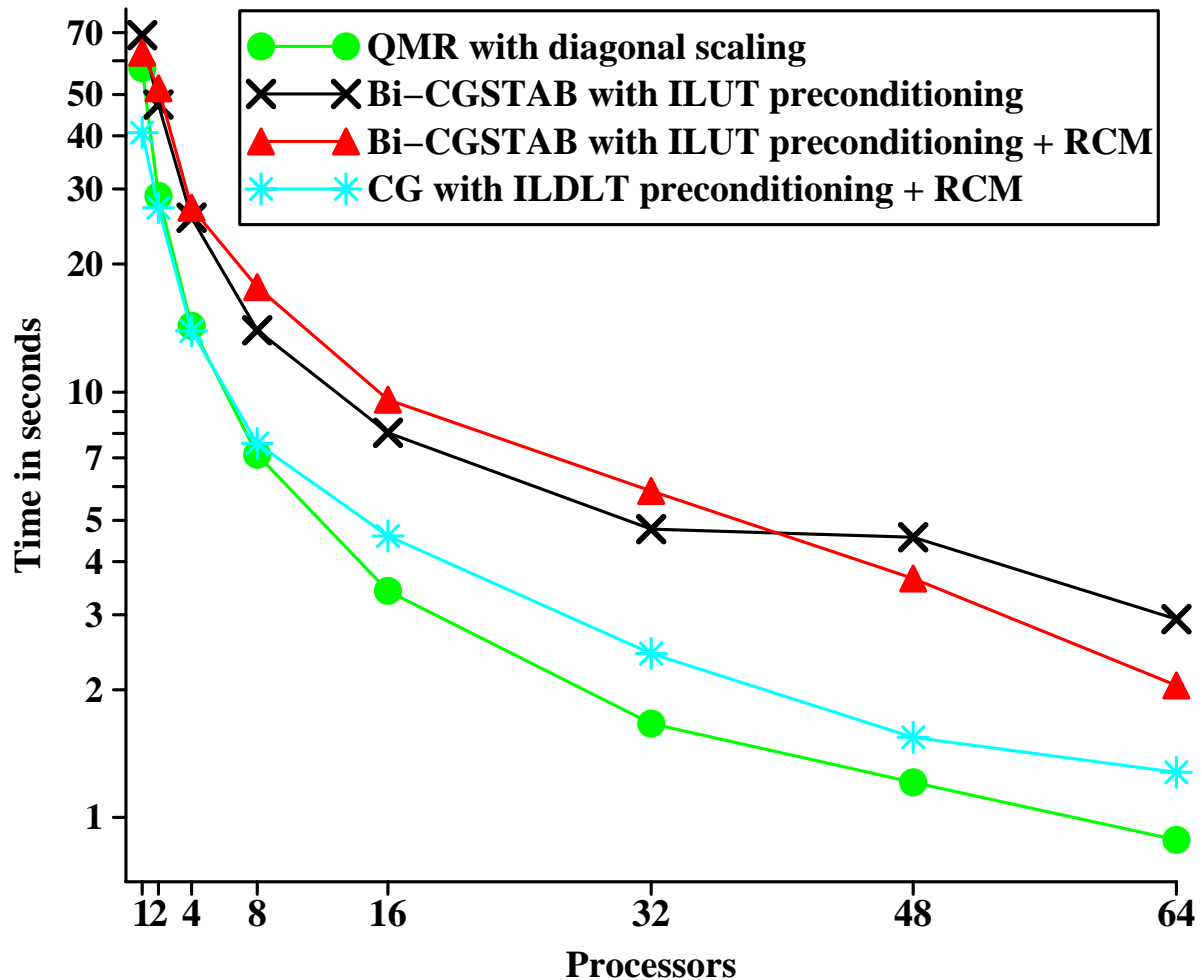


Figure 6: Time and scaling behaviour of the parallel PILUTS solvers.

In Figure 6, diagonally preconditioned QMR shows a significantly better time and scaling behavior than BiCGstab/ILUT although it needs 307 iterations for convergence. BiCGstab/ILUT or BiCGstab/ILUT/RCM require 57 or 52 iterations on 1 processor but 303 and 221 on 64 processors. Additionally, the costs for constructing and applying the preconditioner are so high that QMR is even faster on 1 processor. The fill-in value of ILUT for 1 processor is 3207146. This shows that preconditioner construction and application is distinctly more costly than a matrix-vector multiplication since the matrix only has 1402157 nonzeros. The total fill-in for 64 processors is 476356. The speedups on 64 processors for QMR, BiCGstab/ILUT and BiCGstab/ILUT/RCM are 65.0 (slightly superlinear due to cache effects), 23.6 and 30.7, respectively.

CG/ILDLT/RCM is the fastest method for up to 4 processors. The difference is distinct for 1 processor: The execution time for CG/ILDLT/RCM is 30% faster than the time for diagonally scaled QMR. The fill-in value of ILDLT is 860716 for one processor; CG/ILDLT/RCM requires 106 iterations. For higher processor numbers, the preconditioning effect of block ILDLT decreases so much that diagonally scaled QMR becomes the faster method. For 64 processors, the fill-in of the ILDLT factors is 219821, CG/ILDLT/RCM requires 334 iterations, and a speedup of 31.8 is achieved.

Note that CG/ILDLT/RCM is in all cases markedly faster than BiCGstab/ILUT due to smaller preconditioning costs and fewer matrix-vector multiplications for more than 1 processor.

4.3 Technical Development Plan

1. Integration of the **PILUTS** solvers into **HEAD-FEM**.
2. Further investigation of the effect of matrix re-ordering via **DRAMA** on incomplete block factorisation preconditioning.
3. Making the incomplete block factorisation preconditioners more global by considering couplings between the blocks.

5. S-CAUCHY

5.1 Introduction

A main purpose of brain mapping is to chart the internal structure of the brain and to localise areas of certain function or dysfunction using non-invasive measurement techniques. A widespread technique to measure the functional state of the brain is to record the electromagnetic activity of the brain at the surface of the head and to perform source reconstruction for localising the actually active brain area(s). During the last years source localisation of cerebral activity with respect to the individual anatomy became a prominent goal of electroencephalography (EEG) and magnetoencephalography (MEG) (see Figure 7).

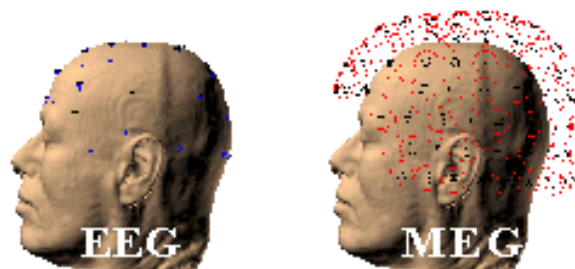


Figure 7: EEG and MEG electrodes on the head surface.

Though the finite element method (FEM) is known to be able to treat geometries of arbitrary complexity and to be capable of modelling variable material properties and anisotropy in computer simulations, it was rarely applied to the problem of source localisation in the brain. This might partly be explained by the relatively high computational costs of FEM when compared with other techniques such as the Boundary Element Method (BEM). However, FEM is starting to attract an increasing interest in this scientific community because anisotropy modelling can only be done on the basis of finite elements. Due to the fact that computer power is still increasing the major disadvantage of the finite element method might become less and less important.

An interdisciplinary research team headed by Prof. Buchner of RWTH Aachen/Germany started in 1993 to develop the finite element based program CAUCHY. The major goal was to prepare the tools for source reconstruction within the individual anatomy for experimental use. This tool is the basis for the further improvement of source localisation software planned within SimBio. The improved tool will be referred to as S-CAUCHY in the following where the capital S stands for SimBio.

For a better understanding of the remainder of this deliverables a brief CAUCHY glossary is given:

- **Forward problem:** Calculation of the electromagnetic fields for a known dipolar current source in the brain.

- **Inverse problem:** see Design report D4.1a
- **Influence node:** Location where a dipole is assumed (e.g. a gray matter node or a node at the cortical surface).
- **Influence or lead field matrix:** A matrix containing the forward field solution for all dipoles at influence nodes. The field solution is restricted to the EEG/MEG-measurement points. In general three columns in the lead field matrix per influence node describe the field distribution of three unit dipoles in the Cartesian coordinate directions. To give an example: If 148 MEG-magnetometers and 128 electrodes have to be simulated and if the gray matter was discretised with 5000 influence nodes, the lead field matrix will have 276 rows and 15.000 columns.

5.2 Implementation

5.2.1 Current Status

As explained in Section 2.1 CAUCHY is a calculation tool for source reconstruction based on the finite element method. It was originally written in F77. One of the first modifications made within the SimBio project was to enable dynamic memory allocation by rewriting some of the CAUCHY core routines. Thus, memory allocation is much more efficient in the new S-CAUCHY.

CAUCHY equation systems are solved by preconditioned conjugate gradient solvers. In the original CAUCHY version, Jakobi- and incomplete Cholesky with “no fill in” preconditioners have been used [buc97]. Within S-CAUCHY incomplete factorisation preconditioners with threshold techniques (Chapter 4) and algebraic multigrid preconditioners (Chapter 5) have been added and compared with the other preconditioners. First tests using the parallelised versions of the threshold factorization preconditioner have been carried out (see Section 5.2.3).

Some of the inverse methods delivered with the original CAUCHY software will become part of the inverse toolbox developed in ST4.1 (for further details see D4.1a).

5.2.2 Interaction with other SimBio Components

Figure 8 shows a linear test chain of SimBio software tools that are coupled to calculate one forward solution for source localisation. The depicted chain is designed for the solvers described in Chapter 4. The component interaction for the AMG solvers of the PEBBLES package is explained in Chapter 3.

VGrid is SimBio’s mesh generation tool (see D1.2a) whose sequential output is converted into a distributed mesh format readable by the DRAMA tool (see Section 2). The result of the DRAMA call is a reordered sequential mesh file and the partitioning information that is handed over to the solver described in Chapter 4. In this interaction scenario S-CAUCHY is used for the assembly of the FE geometry matrix (being positive definit, symmetric, sparse) and for the setup of the right hand side of the linear equation system (the source). S-CAUCHY’s task is to take account of boundary conditions of the governing partial differential equation and to model dipolar sources in the brain. In the current implementation S-CAUCHY stores the equation system in a compressed row binary format. This binary file together with the partitioning information is input for the parallel solver that calculates potential values evoked by the dipolar source. The results are visualised in an image processing environment (IPE) that is described in D5a.

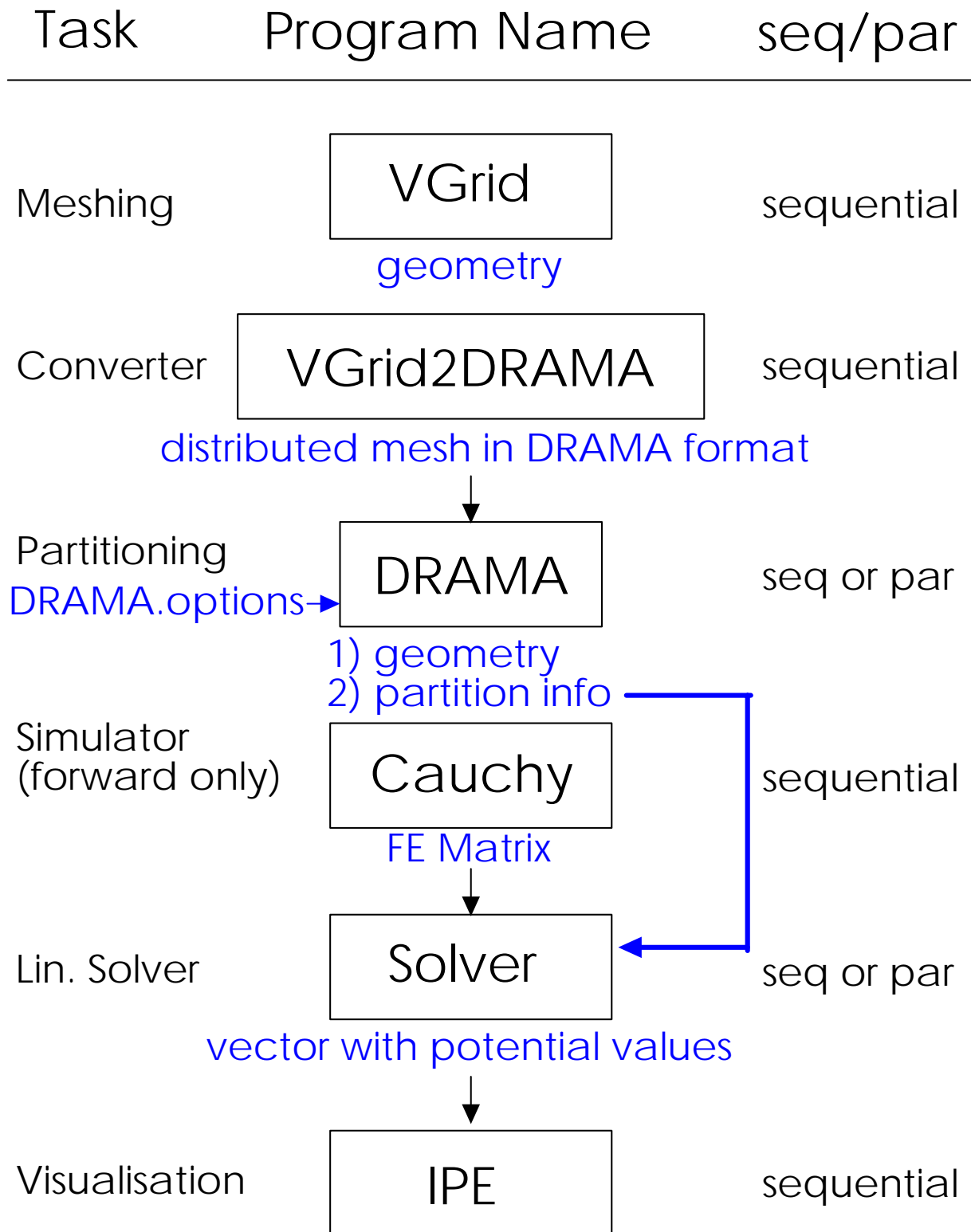


Figure 8: A linear software chain that generates a forward solution starting from FE meshes

Once the FE matrix for a specific geometry with well defined material properties is set up, it will not be subject to any changes. The only data being modified in the course of the program execution is the right hand side of the equation system, i.e., the position of the dipolar source.

One possible way of doing source localisation is to calculate a field distribution for each influence node of a discretized influence space. The calculated potentials at the EEG/MEG measurement points are stored in the lead field matrix (see Section 5.1 for explanations). This lead field matrix serves as

input for *some* of the inverse methods to be developed in ST4.1. In a typical setting the equation system has to be solved for about 10.000 right hand sides (i.e. dipole locations). Thus, the main calculation costs of a forward computation are to be expected in the equation solution procedure. The setup of an FE matrix represents only a small percentage of the whole calculation time. Therefore S-CAUCHY will remain a sequential tool, because the effort that would have to be spent for parallelisation would not pay off in terms of reduction of execution time.

5.2.3 Preliminary Results

A detailed description of the following recently presented results and further simulation studies on anisotropic multi-layer sphere models can be found in [wol00]. For a realistic simulation of the electromagnetic field propagation in the human head for a given dipolar current source in the brain, a 5-tissue (white and gray matter, liquid, skull and scalp) volume conduction model was created.

Isotropic brick meshes with and without shifted nodes for surface smoothing (see Section 3.4.1 of the design report for ST1.2.) were used to study the influence of the “nodeshifting” procedure on solver performance. No significant differences in the solver times were observed. Thus, only the results obtained with the “nodeshifted” mesh are presented in the following.

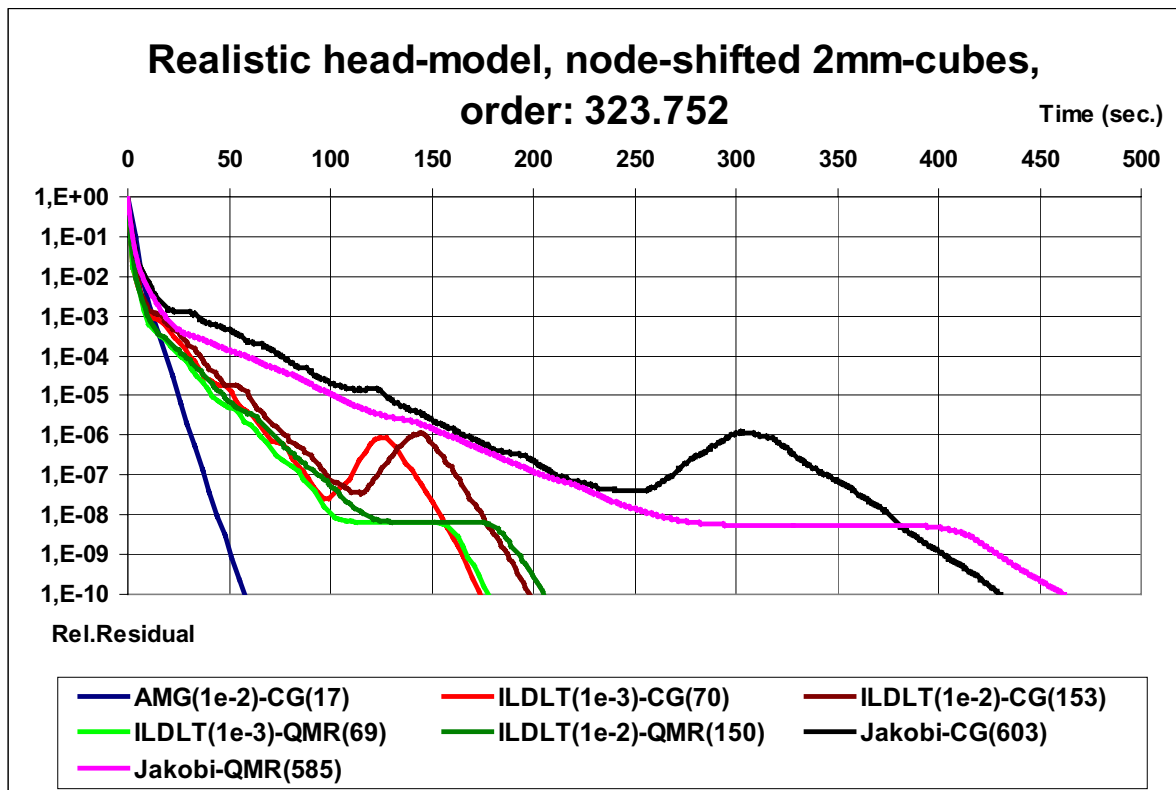


Figure 9: 1-proc. solver comparison (L_2 -relative residual).

A comparison between the solver performance of the incomplete factorization (Chapter 4) and AMG (Chapter 5) preconditioning for the CG method was carried out on a 1-processor-machine in order to test the “mathematical” properties of the different preconditioning techniques with regard to the SimBio application. The results are shown in Figure 9. Up to a relative residual of 10^{-3} in the solution of the FE equation system, the differences between the incomplete factorisation (ILDLT) and the AMG preconditioned Krylov methods are minimal. For residuals ranging from 10^{-5} up to 10^{-10} , the AMG preconditioned CG method is the fastest in all tested cases and a factor 2 to 3 times faster than the best tuned ILDLT preconditioned Krylov method.

5.3 Technical Development Plan

S-CAUCHY input and output routines will be changed to read and write Vista format files.

The FE mesh is stored in a Vista graph file (see D1.2a) that also contains the information about influence nodes. The S-CAUCHY tool creates the lead field matrix in binary Vista format. The lead field matrix is stored as a one-dimensional image file in order to avoid compatibility problems between Fortran77 and C++ matrix conventions.

The effect of the “nodeshifting” will be examined. This can only be done for a simple geometry such as the multi-layer anisotropic sphere model, where analytical solutions for the calculation of the dipole induced potential distribution exist [mun88, mun93]. In a first step, the potential distribution is derived for a monopole load and in a second step the distance between a positive and a negative monopole is decreased up to zero in order to describe the potential of a dipole source. In CAUCHY, a dipolar source is numerically represented as a monopole load around the dipole node following the law of Saint Venant [buch97]. We call this source model the “blurred dipole”. The “blurred dipole” is dependent on the FE-mesh. As for studying the consequences of “nodeshifting”, the source models have to be identical, a “real dipole” model has to be implemented in S-CAUCHY. This is realised by an analytical treatment of a “real-dipole” induced potential singularity under the assumption of an infinitely extended volume conductor with constant conductivity and a numerical treatment of the rest potential in the considered volume conductor [Ran97, Mar98].

S-CAUCHY will be coupled with the parallel version of PEBBLES as it is described in Chapter 3. During the setup phase of the geometry matrix, the element matrices will be transferred to PEBBLES. Then PEBBLES generated the local stiffness matrices and administers the boundary nodes. Finally, PEBBLES has to be called for each different right-hand-side.

The S-CAUCHY tool will be tightly coupled with the inverse toolbox (ST4.1). The toolbox contains a variety of state of the art inverse source reconstruction algorithms. They will be implemented in C++ using a modular class structure.

The Fortran77 functions from the S-CAUCHY tool will be called from the C++ classes. The parameter transformation will be assured in the concrete C++ class and the global variables which are used in S-CAUCHY will be set in the calling C++ class. Besides, the parameter file used by the solver will be passed to S-CAUCHY.

A hierarchic class structure will be used to reduce the number of interfaces and to keep them clean of implementation details. The class structure of the inverse toolbox is described in Chapter 3.2 of the final design report of ST 4.1. The coupling of S-CAUCHY tools concerns search space grids, grids for forward computations and weighting algorithms, forward simulators and L1-norm inverse algorithms. Thus the toolbox of ST 4.1 provides abstract class interfaces for grid generators, forward simulators and inverse analyzers using a discrete search space. S-CAUCHY tools will be interfaced to classes that are derived from these abstract classes. The definitions of the class interface are available via the access restricted document part of the SimBio WWW-pages (SimBio Documents: <http://www.ccr-l-nece.technopark.gmd.de/simbio/reports.html>). In addition a class describing a grid is defined in appendix C of the final design report of ST 4.1.

To provide access to the methods of the inverse toolbox including S-CAUCHY tools, three interface levels are defined in chapters 3.4 –3.6 of the final design report of ST 4.1. These interfaces provide reasonable scenarios containing a complete source modelling procedure. A description and examples are given to add new user scenarios. One level of the interfaces enables the calling of inverse methods on a command line level. This level can be integrated in a complete SimBio application environment.

For the tight coupling between ST4.1 and WP3 in the case of a continuous parameter space for the inverse reconstruction described here, the S-CAUCHY tool will be extended to provide a source simulation also for dipoles which are not on the discrete grid, but in the continuous space. The simulator will be able to simulate several dipoles in the continuous space in the same time. In the case of a discrete parameter space, the coupling between WP3 and ST4.1 will be on a simple file level.

6. PAMSAFE

6.1. Introduction

Since 90's years, the FEM code, explicit non-linear solver developed by ESI, has been completed with airbag, seatbelt and dummy models to make PAM-SAFE™ a comprehensive occupant safety package. From this safety solver, the first studies in biomechanical field have been performed [pam95, bea95]. That answered to the safety user request to model the behaviour of the human and not only the one of the dummy [pam98, pam99]. The development of the biomechanical models for the health began with the European project, KNEES-UP [knee], where the first 3D FE knee model has been developed for improving menisci prostheses. This one year project, in collaboration with Sheffield University, permitted us to bring to light the specific problems met with this kind of biomechanical models (very confine environment and very small components, ...). Another specific problem of these simulations in the health care field is the large simulation time required for e.g. a gait cycle simulation (in the order of 1 second).

6.2. Implementation

The biomechanical models and simulations require some improvements of the PAM-SAFE™ package, the solver and its pre-processor, GENERIS™. It is planned to

- to facilitate the modelling of the mesh,
- to improve the modelling of the biological materials, in particular the menisci,
- and to decrease the CPU time of the simulation.

6.2.1 Current Status

6.2.1.1 Mesh Element Type

Until now, the PAM-SAFE™ solver works with 8-nodes solid elements in order to obtain an accurate response of the mesh. This restriction will be overcome within this project. Four-node elements, (i.e. tetrahedra) are wide-spread finite elements used for modelling complicated geometries. Thus, for the human geometry that is especially complex this element type is very suitable. The tetrahedral formulation must therefore be improved in the PAM-SAFE™ solver. Moreover the WP1 of SimBio develops an automatic mesh generator providing tetrahedral meshes (see D1.2a).

6.2.1.2 Biological Material Characterisation

Due to the complexity of the biological material and due to the difficulty of performing experimental tests for their characterisation, it is very important to take into account the microstructure of the biological material. In particular, for the knee model the microstructure of the menisci and of the ligaments is of high relevance. The fibre direction is the main characteristics of the microstructure of these biological components. Currently, the user defines the fibre direction in the global frame or in the local frame of each element constituting the material. The first solution (global frame) cannot be used for a very complex fibre direction as it is to be found in the menisci. The second solution (local frame) is very demanding for the user, because no automatic tool exists. Thus, to accurately control this fibre direction the mesh must follow this direction. This is an extremely challenging requirement for the mesh generation process. Furthermore, no visualisation tool of fibre is currently available. Some divergence problems have been observed with materials types like hyper-elastic or composite developed in PAM-SAFE™ code for the specific parameters of the meniscus. Particular attention will be paid to reinforce or adapt these material formulations to the meniscus case.

6.2.1.3 Parallel version of PAM-SAFE™ solver

Currently, the parallel version of PAM-SAFE™ does not integrate all new developments of the standard version. A setting of new (or future) developments necessary for the bio-mechanical studies like the knee behaviour study will be performed and an effort to decrease the CPU time of this kind of simulation will be undertaken.

6.2.2 Interaction with other SimBio components

Figure 10 shows the interaction between the PAM-SAFE™ package and other SimBio components:

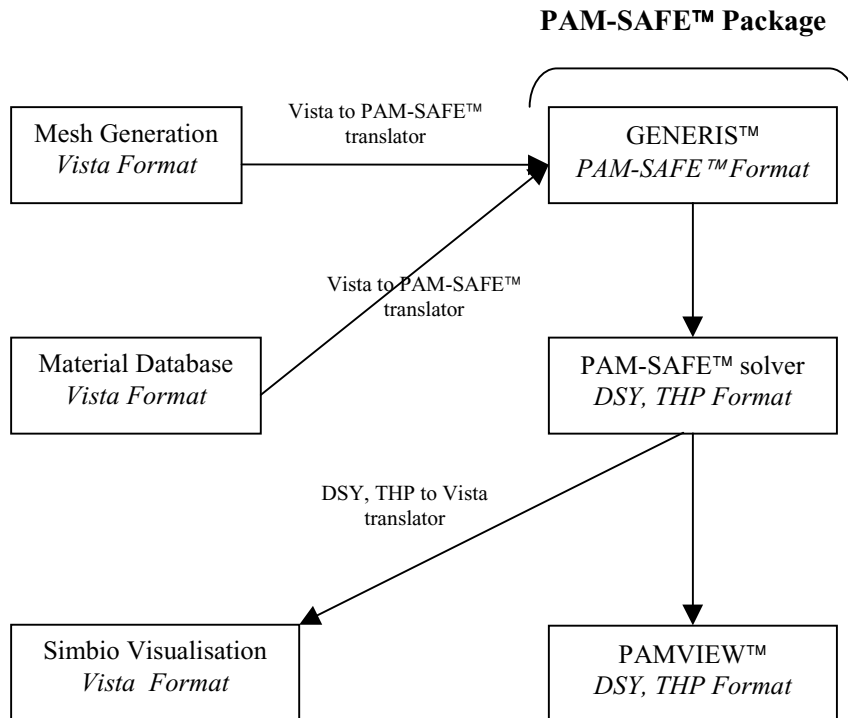


Figure 10: The above diagram shows that the main interaction of the PAM software is to be expected with the mesh generation tool.

For an optimal integration of the ESI tools (GENERIS™, PAM-SAFE™, PAMVIEW™) into the SimBio environment some scenarios will be defined and transferred to WP6. We will clarify whether or not the translators are integrated to the tools, following the global SimBio strategy, and how an efficient corbatisation of the ESI package will be created for a common user (see D6a).

6.2.3 Preliminary Results

First tests have been performed to evaluate the current tetrahedral formulation response in relation to the one of the hexahedral formulation.

6.3 Technical Development Plan

As explained in the current status of the PAM-SAFE™ package (see 6.2.1), the technical development concerns the pre processor, GENERIS™ and the PAM-SAFE™ solver.

6.3.1 GENERIS™

1. Visualisation of the fibre direction in elements, in particular solids.
2. Automatic orientation of the fibre in the element local frame.
3. Reading and writing of the tetrahedron element definition.

6.3.2 PAM-SAFE™ Solver

1. Evaluation of current tetrahedral formulation and its improvement.
2. Improvement of some materials in the case of specific parameters of menisci.
3. Performance evaluation of the current parallel version.
4. Settings of new (or future) developments necessary for the knee behaviour study in the parallel version.
5. Setting new features in the splitter tool.
6. Work on the simulation time reduction for this kind of simulation.

6. 3. 3 Converters

Moreover to permit the interaction with the Simbio environment, translators for SimBio data will be developed:

1. Mesh Vista to Pam-SAFE™ format
2. PAM-SAFE™ output DSY, THP to Vista format
3. Integration of the tool in the SimBio environment

7. HEAD-FEM

7.1 Introduction

About twenty-five years ago FEM was first used to study the biomechanics of the head [ward75]. A detailed review of the progress made ever since is given in [khalil82, sauren93]. Most head models have been designed to find mechanisms describing brain injury after impact as occurring in accidents. A deeper understanding of these mechanisms is necessary for the improvement of head security devices. The other models deal with investigating the consequences of structural alterations within the head. Intracranial mass changes due to tumor growth or fluid accumulation [subra95] have been examined within this context.

Though the head models have proven their general applicability to the field of head biomechanics they are still far from being a tool for the neurologist in clinical practice. This situation is mainly explained by the following shortcomings of existing models:

- Most models are based on a crude representation of the brain anatomy. Many head models are two-dimensional and of poor spatial resolution.
- Their geometry is based on images taken either from anatomical textbooks or adapted from medical image data.
- Current models are not flexible. Finite element meshes of the head are often generated by hand and designed to reproduce the average geometry of the human head. Thus, they do not allow neurological pathologies or deviations from the average head geometry to be taken into account.
- The material models reflecting the biomechanical behaviour of head tissue under load are very simple because of the lack of accurately measured parameters.

To the best of our knowledge only two mechanical head models have been presented to the high performance computing (HPC) community:

Subramaniam and coworkers have created a model dedicated to the phenomenon of hydrocephalus. An obstruction of an aqueductal site within the brain (e.g. caused by a tumor) leads to a water (cerebrospinal fluid, CSF) accumulation that causes an increase of intracranial hydrostatic pressure. Though their simulations are carried out on a supercomputer using a commercial FE package, the underlying two-dimensional FE mesh comprises only several hundreds of finite elements. Nevertheless the authors ensure that their concept leads to physically reasonable results.

The Wayne State University (WSU) head model can be considered to be the most detailed concerning brain anatomy. The three-dimensional (3D) WSU model [zhou95] has been created to establish a relationship between mechanical stress caused by impact and neurological deficit. Recently, it has been further developed in collaboration with PAM System International (PSI) by modelling a more realistic skull structure [pam98]. The WSU model consists of 22000 mesh nodes and the slightly more complicated PSI model comprises 26000 nodal points. Though the spatial resolution seems relatively high, the WSU/PSI model still suffers from a rather poor FE representation of the complex-shaped anatomical structures. In particular the ventricles, large cavities within the brain are barely recognisable. Another drawback of the WSU model can be seen in the costly mesh generation process. According to the authors the production of the FE mesh was "painstaking and time-consuming". The geometry of the mesh is defined once only to represent an average human head.

In this design report, we describe a biomechanical FE code named HEAD-FEM designed to model the mechanical response of the head for some special medical applications. To overcome some of the abovementioned restrictions this FE tool efficiently exploits HPC technology enabling simulations based on FE meshes with a spatial resolution about five times higher than that of previous models. The high spatial resolution guarantees

- a precise FE representation of neuroanatomical structures and
- a high numerical accuracy of the results obtained in reasonable calculation time.

7.2 Implementation

7.2.1 Current Status

HEAD-FEM is a parallel FE code for specialised biomechanical applications. The code is written in C and parallelised on the basis of the Message Passing Interface (MPI) library. Currently, HEAD-FEM exploits the linear solvers provided in the AZTEC library [aztec]. HEAD-FEM enables linear static and dynamic FE analyses. Currently, only eight-node brick FE elements are implemented. The dynamic version has successfully been exploited to model the response of the whole head to externally applied forces.

The HEAD-FEM version for dynamic FE analyses has been installed on a number of parallel architectures, namely on

- a 128 processor NEC Cenju-3 supercomputer (MIPS R4400 in a multi stage interconnection network)
- a 64 processor NEC Cenju-4 supercomputer (MIPS R10000 in a multi stage interconnection network)
- a 32 processor PC cluster (PentiumPro 200MHz interconnected by a high speed myrinet switch).

7.2.2 Interaction with other SimBio Components

As HEAD-FEM is running completely in parallel it requires a distributed mesh as input. Currently a recursive coordinate bisection (RCB) tool is used to partition the mesh generated by VGrid. The n submeshes (where n denotes the number of processors) are read by HEAD-FEM to subsequently assemble the FE stiffness matrix (being positive definit, symmetric and sparse) and to finally solve the system of linear equations. The results of a linear static FE analysis with HEAD-FEM is a vector of displacements for the nodes of the mesh. A postprocessing step offers to the SimBio user the choice between a variety of visualisation types (wireframes, stress plots etc.) of the simulation results. All the output of the postprocessing tool is in Vista format that is accepted by the SimBio visualisation module VM (see D5a).

7.2.3 Preliminary Results

The code for dynamic analyses has proven to be scalable on the above listed machines. The good performance of the FE code enables large-scale simulation results to be obtained in less than one hour for the dynamic case and in a couple of minutes for a static analysis. These timings refer to simulations with one million unknowns executed on 32 processors of the NEC Cenju-4 machine. Figure 11 shows a result obtained with the dynamic HEAD-FEM version. A snapshot of a pressure distribution in the brain evoked by an externally applied force is depicted. The force is applied in the region of the forehead leading to a pressure increase (magenta) in the frontal brain area and to a pressure decrease in the occipital brain region (yellow-red).

The HEAD-FEM code for static FE analyses is installed on the NEC Cenju-4 system . For the SimBio target applications (see Design Report for Subtask 7.2) the static version of HEAD-FEM will be used.

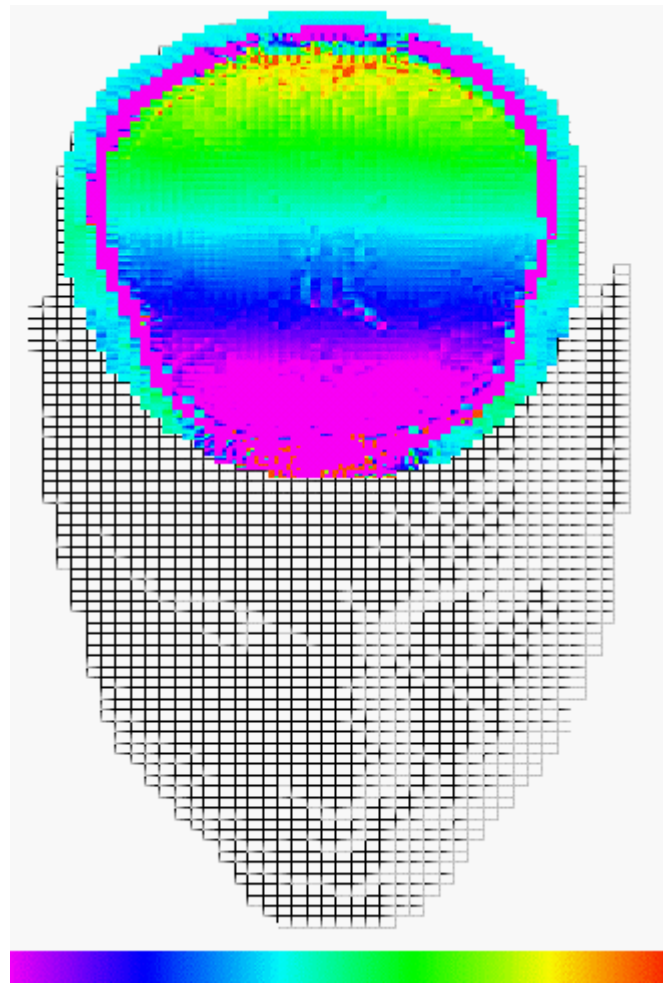


Figure 11: Snapshot of the pressure distribution in the head evoked by an externally applied force.

7.3 Technical Development Plan

HEAD-FEM input and output routines will be changed to read and write Vista format files.

As the mesh generator creates brick, tetrahedron and hybrid meshes, the HEAD-FEM code has to be extended to be capable of handling tetrahedra elements.

HEAD-FEM will also be extended by some nonlinear FE features: It is planned

- to add methods for geometrically nonlinear simulations (such as Newton-Raphson),
- to implement a non-linear material model (e.g. a viscoelastic tissue model) and
- to develop a contact/friction algorithm to take account of sliding interfaces such as brain and skull.

Furthermore the DRAMA partitioner tool will be exploited to find the optimal partitioning for the AZTEC solvers used in HEAD-FEM. Besides the NEC solver routines will be coupled to the HEAD-FEM code enabling studies on the optimal solver strategy for equation systems stemming from biomechanical applications.

References

- [aztec] Hutchinson, SA, Shadid JN, Tuminaro, RS. Aztec User's Guide: Version 1.1 (1995). Sandia National Laboratories Technical Report SAND95-1559.
- [bas00] Basermann, A., "Parallel Block ILUT/ILDLT Preconditioning for Sparse Eigenproblems and Sparse Linear Systems", to appear in Numerical Linear Algebra with Applications, special issue of the Sparse99 meeting, Minneapolis, 1999.
- [bas99] Basermann, A., "Parallel Preconditioned Solvers for Large Sparse Hermitian Eigenvalue Problems", J. Dongarra and V. Hernandez, eds: VECPAR'98 – Third international conference for vector and parallel processing, Lecture Notes in Computer Science, Springer, Berlin, vol. 1573, pp. 72-85, 1999.
- [bas96] Basermann, A., "QMR and TFQMR methods for sparse nonsymmetric problems on massively parallel systems", The Mathematics of Numerical Analysis, series: Lectures in Applied Mathematics, vol. 32, pp. 59-76, 1996.
- [bea95] Beaugonin, M.; Haug, E., Munck, G., Cesari, D., "A Preliminary Numerical Model of the Human Ankle under Impact Loading"; PLEI Conference, Washington, D.C., USA, 1995.
- [buc97] Buchner, H., Knoll, G., Fuchs, M., Rienacker, A., Beckmann, R., Wagner, M., Silny, J., and Pesch, J. (1997) Inverse localization of electric dipole current sources in finite element models of the human head. *Electroencephalography and Clinical Neurophysiology*, **102**:267-278
- [haa00] Haase, G., Kuhn, M. and Reitzinger, S. (2000) Parallel AMG on distributed memory computers, Preprint submitted to Elsevier Preprint.
- [jun91] Jung, M. and Langer, U. (1991) Applications of multigrid methods to practical problems. *Surv. Math. Ind.*, **1**, 217-257.
- [khalil82] Khalil, T.B. and Viano, D.C. (1982) Critical Issues in Finite Element Modeling of Head Impact. *Proceedings of the 26th Stapp Car Crash Conference*, 87-103.
- [knee] Knees-UP Project, ESPRIT HPCN TTN Action, Realistic 3D Model of the Human Knee. World-Wide Web document:
<http://kiefer.gmd.de:8002/popcorn/projects/KNEESUP/www-kneesup.htm>
- [kui95] Kuijpers, A.H.W.M., Claessens, M.H.A. and Sauren, A.A.H.J. (1995) The Influence of Different Boundary Conditions on the Response of the Head to Impact: A Two-dimensional Finite Element Study. *Journal of Neurotrauma*, **12**, 715-724.
- [mar98] Marin, G., Guerin, C., Baillet, S., Garnero, L. and Meunier, G. (1998) Influence of skull anisotropy for the forward and inverse problem in EEG: Simulation studies using FEM on realistic head models, *Human Brain Mapping*, **6**, 250-269.
- [mun88] de Munck, J.C. (1988) The potential distribution in a layered anisotropic spheroidal volume conductor, *J.Appl.Phys.*, **64** (2), 464-470.
- [mun93] de Munck, J.C. and Peters, M. (1993), A fast method to compute the potential in the multi sphere model, *IEEE Trans. Biomed. Eng.*, **40**(11), 1166-1174.

- [pam95] Haug, E., "*Biomechanical Models in Vehicle Accident Simulation*", PAM-User's Conference in Asia ; PUCA'95, pp. 233-256, Shinyokohama, Japan, November 16-17, 1995.
- [pam98] Haug, E., Beaugonin, M. and Tramecon, A. (1998) Current Status of Articulated and Deformable Human Models for Impact and Occupant Safety Simulation at ESI Group, *PAM'98 User's Conference*
- [pam99] Haug, E., Beaugonin, M., Godest, A. C., Tramecon, A., Allain, J. C., Hyncik, L., Choi, L. H., Lee, I. H. (1999) Recent Trends in Biomechanical Simulation and Applications at ESI Group, *PAM'99 User's Conference*.
- [rei00] Reitzinger, S. (2000) PEBBLES: Parallel and element based gray box linear equation solver <http://www.sfb013.uni-linz.ac.at/~reitz/pebbles.html>.
- [saad96] Saad, Y., "Iterative methods for sparse linear systems", PWS Publishing Company, Boston, 1996.
- [sauren93] Sauren, A.A.H.J. and Claessens, M.H.A. (1993) Finite Element Modeling of Head Impact: The Second Decade. *Proceedings of the International IRCOBI Conference on the Biomechanics of Impact*, 241-254.
- [subra95] Subramaniam, R.P., Neff, S.R., and Rahulkumar, P. (1995) A numerical study of the biomechanics of structural neurologic diseases. *High Performance Computing-Grand Challenges in the Computer Simulation Society in Computer Simulations, San Diego* 552-560
- [ward75] Ward, C.C. and Thompson, R.B. (1975) The Development of a Detailed Finite Element Brain Model. *Proceedings of the 19th Stapp Car Crash Conference*, 641-674.
- [wol00] Wolters, C., Reitzinger, S., Basermann, A., Burkhardt, S., Hartmann, U., Kruggel, F. and Anwander, A. (2000), Improved tissue modelling and fast solver methods for high resolution FE-modelling in EEG/MEG-source localisation, *Proceedings of the 12th International Conference on Biomagnetism, Helsinki, Finland, August 12-17*, <http://www.simbio.de>.
- [zhou95] Zhou, C., Khalil, T.B and King, A.I. (1995) A New Model Comparing Impact Responses of the Homogenous and Inhomogenous Human Brain. *Proceedings of the 39th Stapp Car Crash Conference*, 121-138.