# 3.2. The Generation of the FEM-GALERKIN-Scheme: Linear Triagular Elements as Typical Example

## 3.2.1. A Model Problem

- Let us consider a 2D heat conduction problem in Variational Formulation as model problem:

(2)
$$\boxed{\begin{aligned}
&\text{Find } u \in V_g := \{ v \in V = H^1(\Omega) : v = g_1 \text{ on } \Gamma_1 \} : \\
&a(u,v) = \langle F, v \rangle \quad \forall v \in V_0 := \{ v \in V : v = 0 \text{ on } \Gamma_1 \}, \\
&\text{where } a(u,v) := \int_\Omega (\lambda(x) \nabla u \cdot \nabla v + a(x) u v)\, dx + \int_{\Gamma_3} \varkappa u v \, ds, \\
&\langle F, v \rangle := \int_\Omega f v \, dx + \int_{\Gamma_2} g_2 v \, ds + \int_{\Gamma_3} g_3 v \, ds, \\
&\Omega \subset \mathbb{R}^2 \text{ bd Lip}, \ \Gamma = \partial\Omega \in C^{0,1}, \ \Gamma = \overline{\Gamma}_1 \cup \overline{\Gamma}_2 \cup \overline{\Gamma}_3,
\end{aligned}}$$

under the following assumptions:

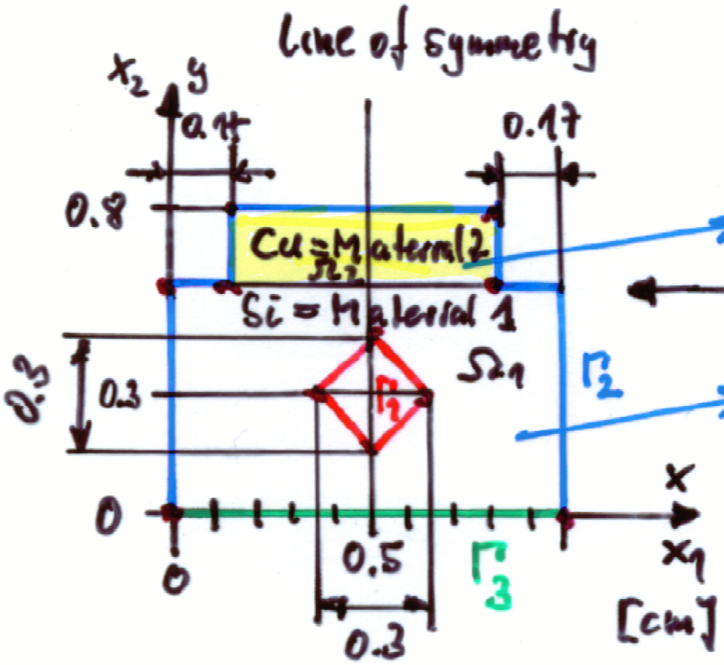
(3)
1) $\lambda \in L_\infty(\Omega) : \exists \underline{\lambda}, \overline{\lambda} = \text{const} > 0 : \underline{\lambda} \le \lambda(x) \le \overline{\lambda} \ \text{a.e. in } \Omega \ \forall x \in \Omega$
2) $a \in L_\infty(\Omega) : a(x) \ge 0 \ \overset{\cdot}{\forall} x \in \Omega$ (i.e., a.e. in $\Omega$)
3) $\varkappa \in L_\infty(\Gamma_3) : \varkappa(x) \ge 0 \ \overset{\cdot}{\forall} x \in \Gamma_3$ (i.e., a.e. in $\Gamma_3$)
4) $f \in L_2(\Omega)$,
5) $g_2 \in L_2(\Gamma_2)$, $g_3 \in L_2(\Gamma_3)$,
6) $g_1 \in H^{1/2}(\Gamma_1)$, i.e. $\exists \tilde{g}_1 \in H^1(\Omega) : \tilde{g}_1 = g_1$ on $\Gamma_1$ (trace)
7) $\Omega \subset \mathbb{R}^2$ bd, $\Gamma = \partial\Omega \in C^{0,1}$, $\text{meas}_1(\Gamma_1) = \int_{\Gamma_1} ds > 0$,

- **Ex. 3.1.:** Give the Classical Formulation (CF) of (2) under classical smoothness assumptions, and give the CF of the example CHIP (interface conditions!), see L11-02!
- **Ex. 3.2:** Show that the assumption of the LAX-MILGRAM-Theorem are fulfilled under the assumptions (3) imposed on the data!

  Hint: First homogenize the Dirichlet BC!

  Result: $\Rightarrow \exists! \, u \in V_g : (2)$

# Example CHIP:




- $f \equiv 0$ (no heat sources)
- $a \equiv 0$ (heat isolation in $z$-dir.)
- $\lambda(x) = \lambda_2 = \lambda_{Cu} = 3.95 \left[\frac{W}{cm\,K}\right]$ Cu (copper)

interface $\Gamma_I$

- $\lambda(x) = \lambda_1 = \lambda_{Si} = 0.01 \left[\frac{W}{cm\,K}\right]$ Si (Silicon)
- $\Gamma_1$ : $g_1 = \text{const} = 500 [K]$

$\Gamma_2$ : $g_2 = 0$ (isolation)

$\Gamma_3$ : $\frac{\partial u}{\partial N} := \lambda_{Si} \frac{\partial u}{\partial n} =$

$\equiv -\lambda_{Si} \frac{\partial u}{\partial y} = \varkappa (u_0 - u)$

with $\varkappa = 0.2 \left[\frac{W}{cm^2 K}\right]$

$u_0 = 300 [K]$

# Remarks:

1. $\exists ! \, u \in V_g$ : (2) are ensured for the given data ($\Rightarrow$ Ex. 2.2)!
2. The solution $u$ has a sharp bend ("Knick") at the interface, i.e. $u \notin H^2(\Omega)$, but only $u \in H^{1+s}(\Omega)$ with some $s < 1/2$ !
3. Loss of regularity of the solution $u$ due to
   - corners at $\Gamma$, in particular, with inner angles $> \pi$,
   - changes of the BC on $\Gamma$,
   - corners in the interfaces $\Gamma_I$ (this is not the case here!),
   - interface $\Gamma_I$ meets the boundary $\Gamma$ :

$\Rightarrow u \in H^{1+s_i}(\Omega_i)$ with some $s_i$ : $1/2 < s_i < 1$.

# 3.2.2. Mesh Generation (Triangulations)

- We decompose the computational domain $\Omega \subset \mathbb{R}^d$, where we look for the solution of the BVP (2), into finite elements $\delta^{(r)}$, e.g. into "small" triangles or rectangles ($d=2$) resp. teds or hexs ($d=3$):

→ **Triangulation** $\mathcal{T}_h := \{ \delta_r : r \in R_h \}$:

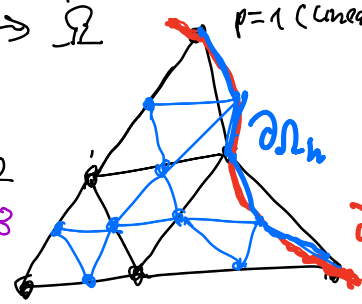
$|\partial\Omega - \partial\Omega_h| = O(h^{p+1})$

$p=1$ (linear)

1) $\bar{\Omega} = \bigcup_{r \in R_h} \bar{\delta}_r$ or $\bar{\Omega}_h = \bigcup_{r \in R_h} \bar{\delta}_r \xrightarrow{h \to 0} \bar{\Omega}$

2) $\bar{\delta}_r \cap \bar{\delta}_{r'} = \begin{cases} \emptyset \\ \text{edge} \\ \text{vertex} \\ \text{face} \end{cases}$ of both triangles $d=2$, $d=3$

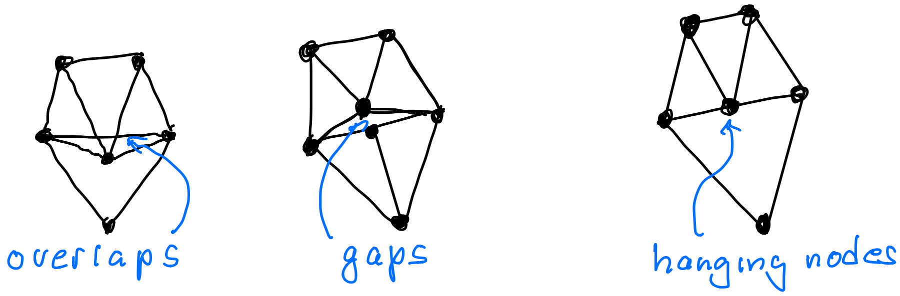
$\forall r, r' \in R_h : r \neq r' \;\; \forall h \in \Theta$




- The decomposition of $\Omega$ into finite elements (FE) depends on the
  - properties of the domain: $\partial\Omega$ = polygon, curvilinear, corners etc.
  - input data of the BVP: discontinous coeff. (interfaces), RHS, BC, ...
  - element typs : → element catalog: △, △, ...
  - accuracy : $\xrightarrow{p}$, $\xrightarrow{h}$ fineness of the mesh: $h$

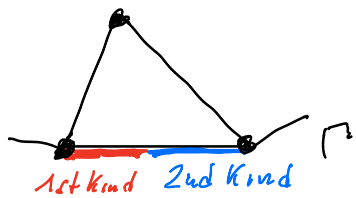
## General hints for the mesh generation:

- inadmissible meshes:

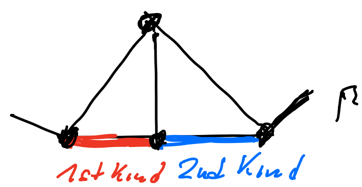



- meshing in the case of changing BC:

wrong

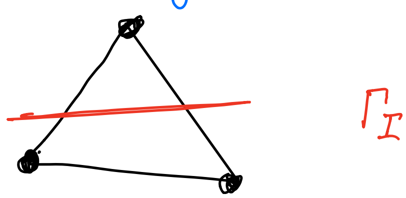



right

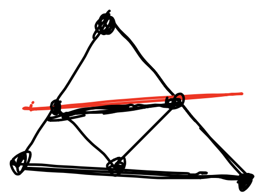



- meshing interfaces $\Gamma_I$:

wrong

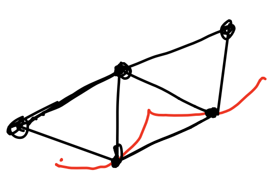



right

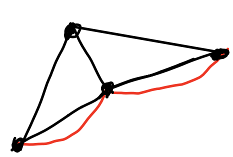


- meshing in the case of kinks in $\Gamma$ or $\Gamma_I$:

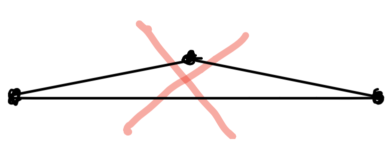
wrong

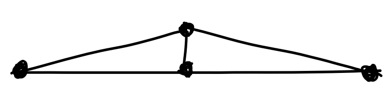


right



- avoid too obtuse or too acute angles





L11-05

■ **Definition** of regular (quasiuniform) triangulations:

(4)

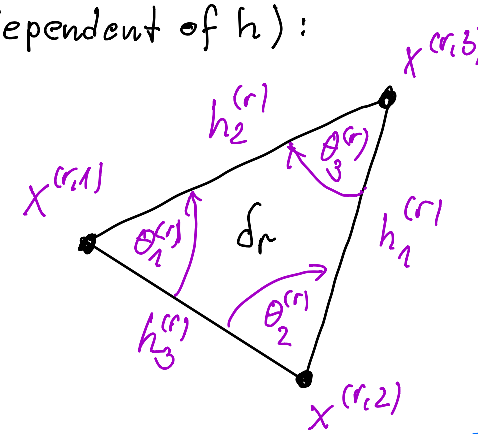
$\exists\, \alpha_0, \theta_0 = \text{const} > 0$ (independent of $h$):

$\alpha_0 h \le h_1^{(r)}, h_2^{(r)}, h_3^{(r)} \le h$

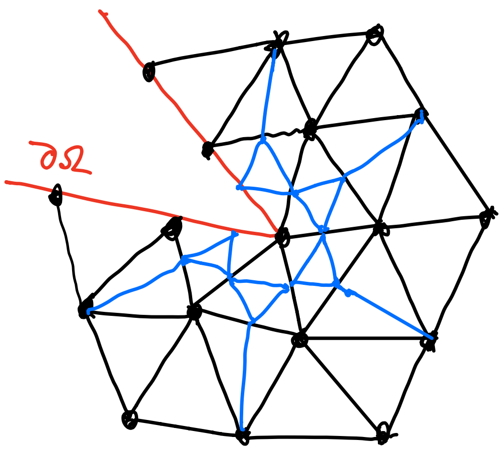
$0 < \theta_0 \le \theta_1^{(r)}, \theta_2^{(r)}, \theta_3^{(r)} \le \pi - \theta_0$

$\forall r \in R_h \ \forall h = \max_{r,\alpha} h_\alpha^{(r)} \le h_0$

$h \in \textcircled{H}$

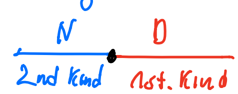



■ **BUT** local mesh refinement due to strongly changing $\nabla u$ because of



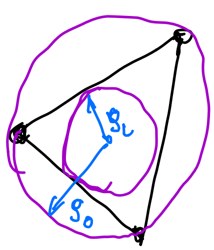

term "shape regular"

- re-entrant corners:
  $u(x,y) = c\, r^\alpha \sin\varphi + \text{reg.}$
  $\alpha = \frac{\pi}{\Theta}$ in case $\Gamma_1$
- Kinks in $\Gamma$ or $\Gamma_T$
- changes in the BC



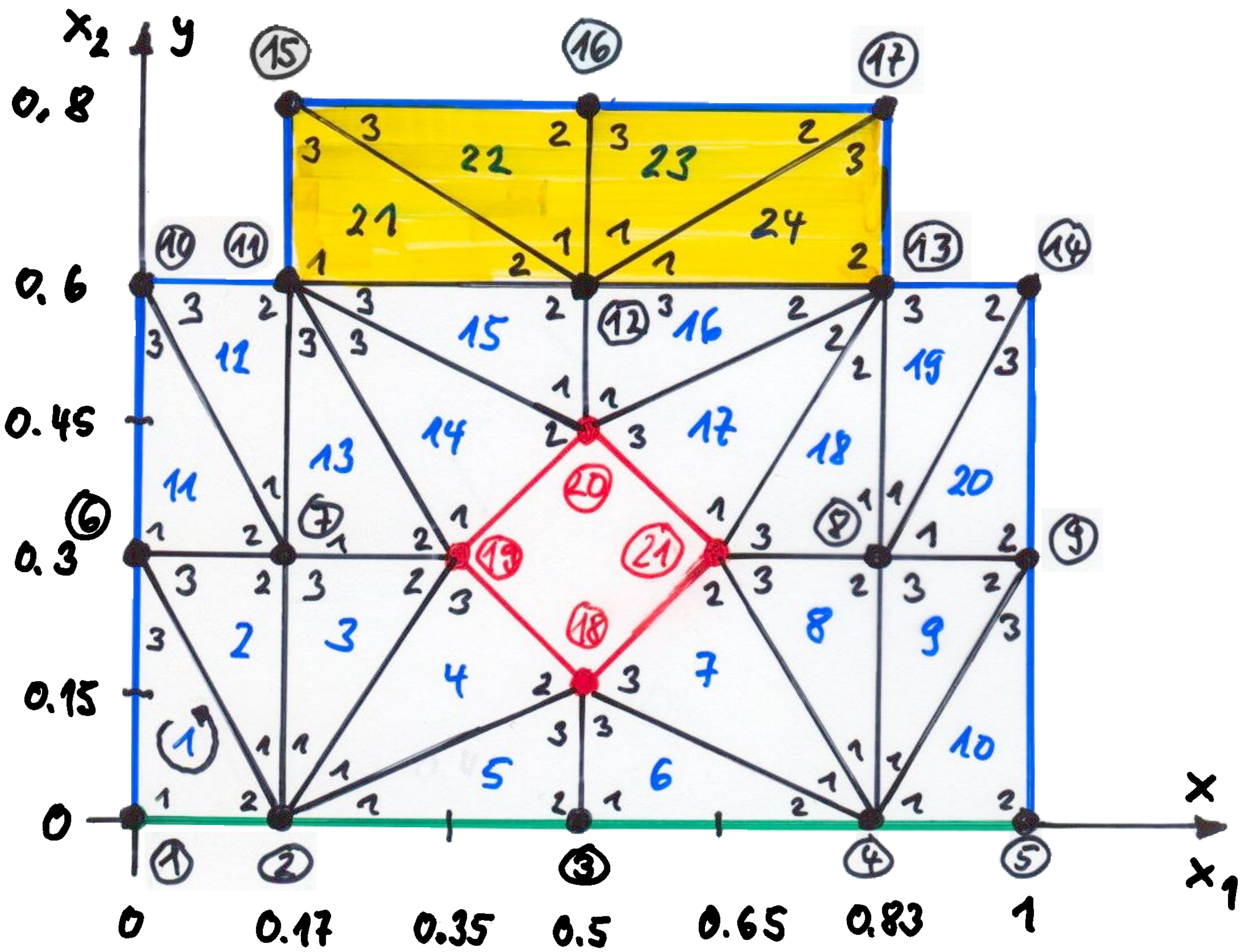

$\hat{=}\ \alpha = \frac{\pi}{2\pi} = \frac{1}{2}$

- boundary layers
- strong changes in the RHS $f$
- etc




$0 < c_\tau \le \frac{\rho_i}{\rho_o} \le 1$ with $c_\tau = \text{const} \ne c(h)$ !

- Discretization (Meshing) of the Computational Domain $\Omega$ of the heat conduction problem CHIP:




$\Longrightarrow$ see mesh file CHIP.NET for the FEM code FEM 2D

L11-07

T10

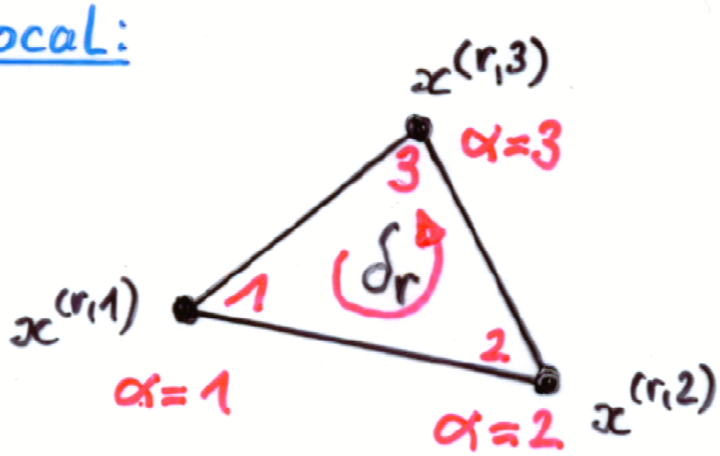
| Mesh File CHIP.NET | | | | Remarks |
|---|---|---|---|---|
| 21 | 24 | | | NX, NE |
| 0 | 0 | | | |
| .17 | 0 | | | |
| .5 | 0 | | | |
| .83 | 0 | | | |
| 1 | 0 | | | $x_i, y_i$ |
| 0 | .3 | | | |
| .17 | .3 | | | $i = \overline{1,NX}$ |
| .83 | .6 | | | |
| 1 | .6 | | | |
| .17 | .8 | | | |
| .5 | .8 | | | |
| .83 | .8 | | | |
| .5 | .15 | | | |
| .35 | .3 | | | |
| .5 | .45 | | | |
| .65 | .3 | | | |
| 1 | 2 | 6 | 1 | |
| 2 | 7 | 6 | 1 | |
| 2 | 19 | 7 | 1 | |
| 2 | 18 | 19 | 1 | Connectivity Material No. |
| 2 | 3 | 18 | 1 | |
| 3 | 4 | 18 | 1 | $\overbrace{IK(1,k), IK(2,k), IK(3,k),}$ MP(k) |
| 4 | 21 | 18 | 1 | |
| 4 | 8 | 21 | 1 | |
| 4 | 9 | 8 | 1 | $k= \overline{1,NE}$ |
| 4 | 5 | 9 | 1 | |
| 6 | 7 | 10 | 1 | |
| 7 | 11 | 10 | 1 | |
| 7 | 19 | 11 | 1 | |
| 19 | 20 | 11 | 1 | |
| 20 | 12 | 11 | 1 | |
| 20 | 13 | 12 | 1 | |
| 20 | 21 | 13 | 1 | |
| 21 | 8 | 13 | 1 | |
| 8 | 14 | 13 | 1 | |
| 8 | 9 | 14 | 1 | |
| 11 | 12 | 15 | 2 | |
| 12 | 16 | 15 | 2 | |
| 12 | 17 | 16 | 2 | |
| 12 | 13 | 17 | 2 | |
| 2 | number of closed boundaries | | | |
| 14 | number of nodes on boundary 1 | | | description of the boundary |
| 4 | number of nodes on boundary 2 | | | |
| 1 | boundary 1 | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 9 | | | | |
| 14 | | | | |
| 13 | | | | $\partial\Omega$ |
| 17 | | | | |
| 16 | | | | |
| 15 | | | | |
| 11 | | | | |
| 10 | | | | |
| 6 | | | | |
| 18 | boundary 2 | | | |
| 21 | | | | |
| 20 | | | | |
| 19 | | | | |

- **Meshing** the computational domain $\Omega$ means the generation of a global and a local mesh topology and their connection:

**global:**

- element numbering: $\mathbb{R}_h = \{1, 2, \dots, R_h\}$,
  $R_h$ = Number of Elements = NE = 24,
- node numbering: $\overline{\omega}_h = \{1, 2, \dots, \overline{N}_h\}$,
  $\overline{N}_h = N_h + \partial N_h$ = Number of Nodes = NX = 21,
  $17 + 4$
- node coordinates:
  $x^{(i)} = (x_i, y_i) = (x_1^{(i)}, x_2^{(i)}) = (x_{1,i}, x_{2,i})$
  $i \in \overline{\omega}_h$

**local:**




$x^{(r,\alpha)}$ – local nodes

$\alpha \in A_r = A = \{1, 2, 3\}$ = local node numbering

**Connection:**

| | |
|---|---|
| $r : \alpha \longleftrightarrow i = i(r,\alpha)$ <br> $\mathbb{R}_h \ni r$, $A_r \ni \alpha$, $\overline{\omega}_h \ni i$ | $i : x^{(i)} = (x_1^{(i)}, x_2^{(i)})$ nodal coordinates <br> $\overline{\omega}_h \ni i$ |

- The mesh file required by every FE-code contains, at least, the following 2 data blocks the structure of which are explained for our example CHIP (see also T 09 and T 10a)

$r : \alpha \leftrightarrow i$

element connectivity table

| $r \in R_h$ | $A_r \ni \alpha \longleftrightarrow$ | $i \in \overline{\omega}_h$ | | Further element information, e.g. Material Property |
|---|---|---|---|---|
| numbering of the ele. $r$ | global node numbers of the local nodes $x^{(r,1)}$ $\alpha=1$ | $x^{(r,2)}$ $\alpha=2$ | $x^{(r,3)}$ $\alpha=3$ | MP |
| 1 | 1 | 2 | 6 | 1 |
| 2 | 2 | 7 | 6 | 1 |
| 3 | 2 | 19 | 7 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| $R_h$=NE=24 | 12 | 13 | 17 | 2 |

or marking elem. for refinement

$i : x_i, y_i$

Coordinates of the nodes

| $i$ | 1 | 2 | 3 | ... | 20 | 21 |
|---|---|---|---|---|---|---|
| $x_i$ | 0.0 | 0.17 | 0.5 | ... | 0.5 | 0.65 |
| $y_i$ | 0.0 | 0.0 | 0.0 | | 0.45 | 0.3 |

- <u>Remark 2.2:</u>

Further data for characterizing or marking nodes are possible, e.g.:
- description of the boundary $\Gamma$ (see input file for FEM2D)
- coding of the boundary conditions,
- coding of refinement information at some node,

etc.

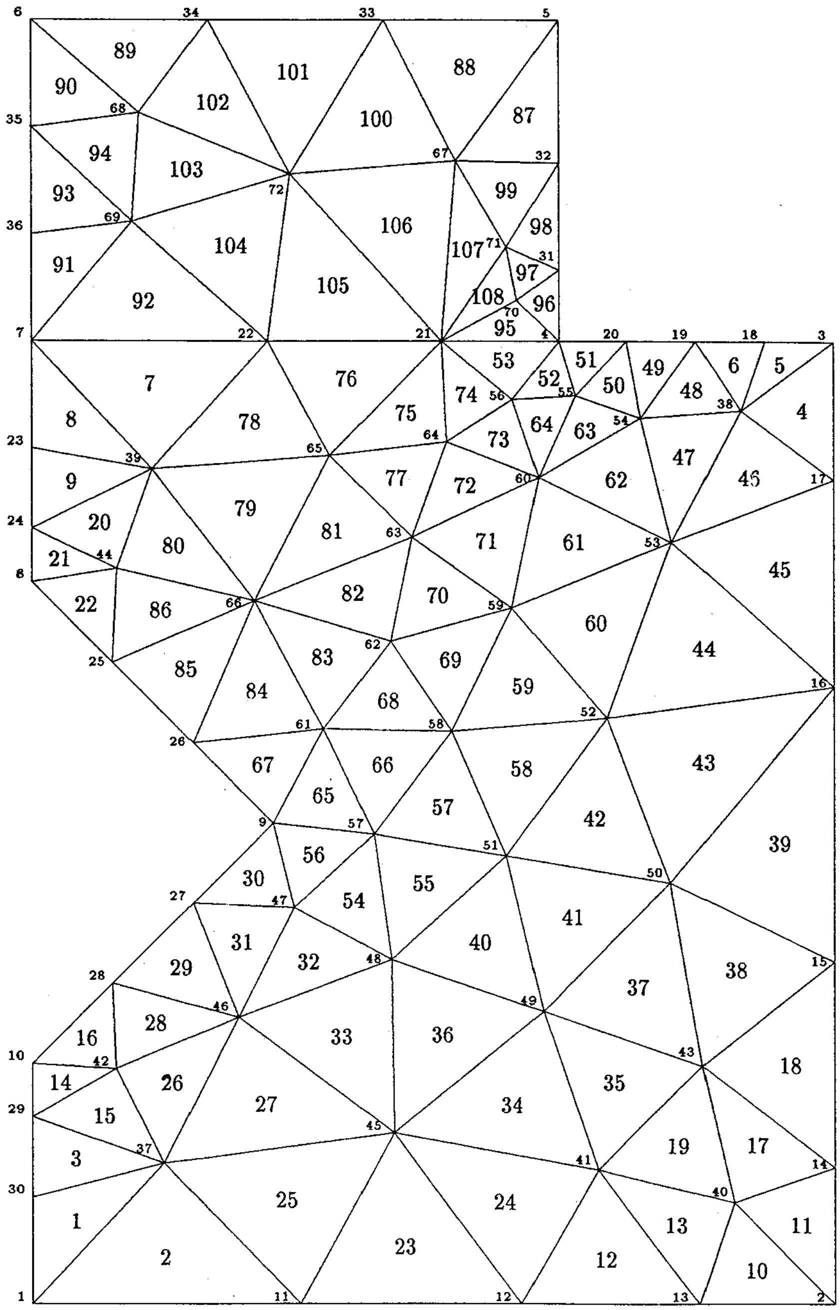
Finer Mesh L 11-10 T 10d

- $r$ : $\alpha \leftrightarrow i$, $r \in \mathbb{R}_h$, $\alpha \in A^{(r)} = A$, $i \in \bar{\omega}_h$

| Elementnummer | globale Knotennummern der lokalen Knoten $P_1^{(r)}$ | $P_2^{(r)}$ | $P_3^{(r)}$ | Elementkennzahl z.B. Materialbereichsnummer |
|---|---|---|---|---|
| 1 | 1 | 37 | 30 | 1 |
| 2 | 1 | 11 | 37 | 1 |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $R_h = 108$ | 21 | 70 | 71 | 2 |

- $i$ : $(x_{1,i}, x_{2,i})$

| $i$ | 1 | 2 | 3 | $\cdots$ | $\bar{N}_h = 72$ |
|---|---|---|---|---|---|
| $x_{i,1}$ | 0.5 | 1.0 | 1.0 | $\cdots$ | 0.66 |
| $x_{i,2}$ | 0.0 | 0.0 | 0.6 | $\cdots$ | 0.7 |



Vernetzung der rechten Hälfte des Gebietes aus der Modellaufgabe 1

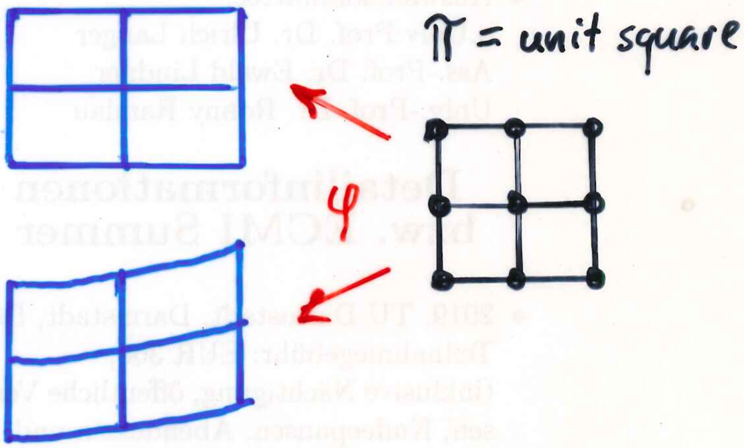
## Methods for Generating the Mesh Data:

1. Generating the data connecting the <u>local</u> and <u>global</u> numbering of the nodes $\boxed{r:\alpha \longleftrightarrow i}$ and the nodal coordinates $\boxed{i : x_i, y_i}$ by <u>"hand"</u>!
2. <u>Semi-automatic</u> generation of the mesh data, i.e., e.g., decompose the domain $\overline{\Omega}$ into subdomains $\overline{\Pi}_e$. To mesh the subdomains $\overline{\Pi}_e$, we use prepared meshes of standard domains from some libary. These meshes of the standard domains will be maped to the subdomains $\overline{\Pi}_e$. We have to ensure that the composed mesh of $\overline{\Omega}$ is admissible (conform)!

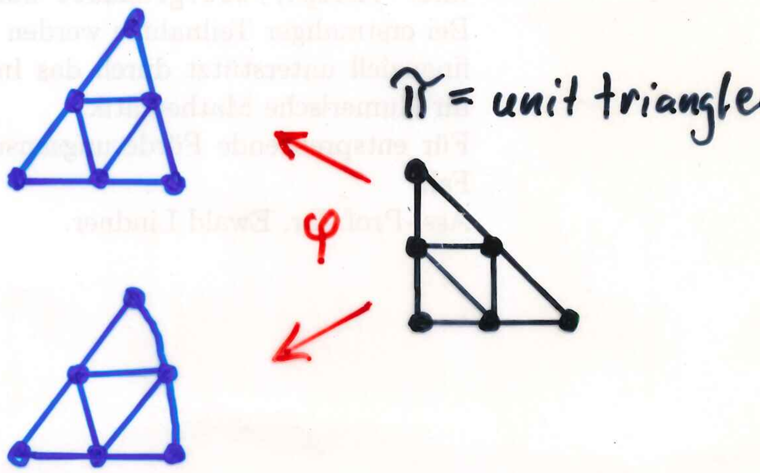
$$\overline{\Omega} = \bigcup_e \overline{\Pi}_e \qquad \overline{\Pi}_e = \varphi_e(\overline{\hat{\Pi}})$$

$\overline{\Pi}_e$ = rectangle

$\overline{\Pi}_e$ = curvilinear rectangle

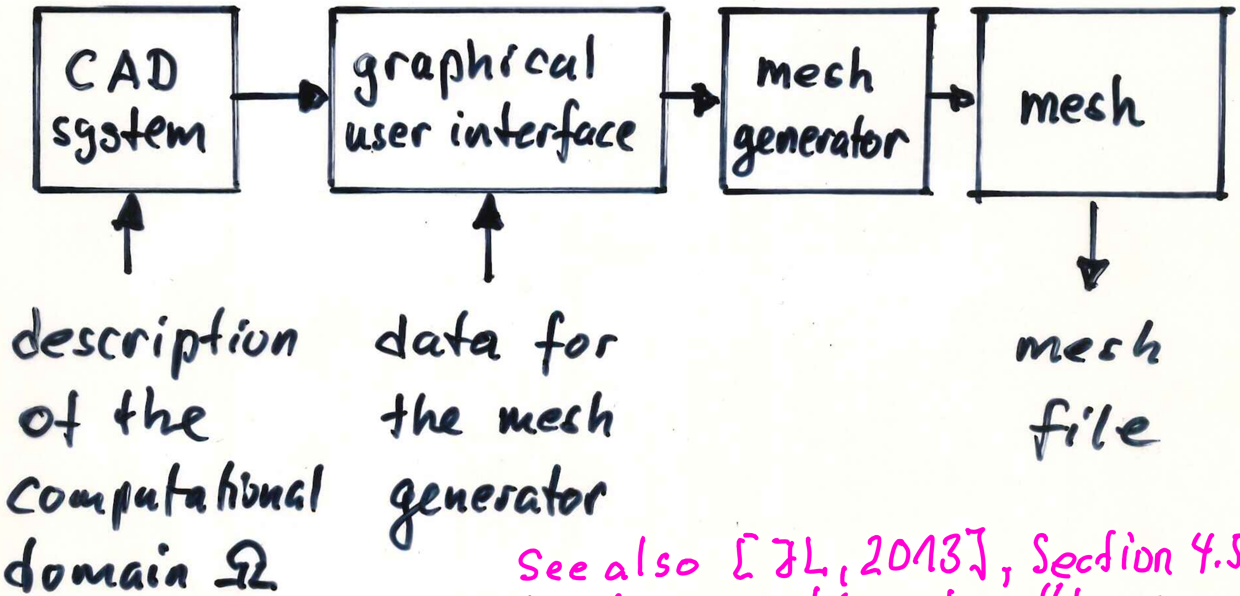



$\overline{\Pi}_e$ – straight or curvilinear triangle



3. The use of automatic mesh generators:

An automatic mesh generator usually requires a description of the boundary $\partial\Omega$ of $\Omega$ or a decomposition of $\Omega$ into subdomains $\overline{\Omega} = \bigcup_e \overline{\Omega}_e$ and a description of the boundaries $\partial\Omega_e$ of the subdomains $\Omega_e$. In addition to this, the mesh generator needs some information about the fineness ($h$) of the mesh, e.g. by providing the distribution of the nodes on the subdomain boundaries $\partial\Omega_e$




See also [JL, 2013], Section 4.5.1
Mesh generation algorithms:
1) Advancing Front
2) Delauny

## Examples:

NETGEN : http://www.hpfem.jku.at/netgen/index.html

SPIDER : http://www.meshing.at
— " — .org

## Mesh generators in the web:

http://www-users.informatik.rwth-aachen.de/~roberts/software.html
→ Robert Schneiders

http://www.andrew.cmu.edu/user/sowen/mesh.html
→ Steve Owen

# 4. Mesh generation using

<u>a priori</u> and / or <u>a posteriori</u>

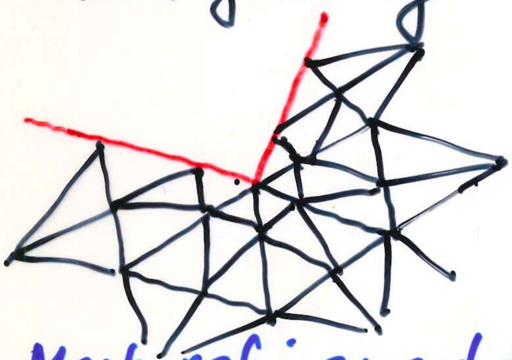
information

<u>before the FE calculation</u>

analysis of the input data
( $\exists$ of obtuse corners in $\Gamma = \partial\Omega$, $\Gamma_I$, coefficient jumps in the PDE etc.)

$\Downarrow$

mesh grading



<u>after the FE calculation</u>

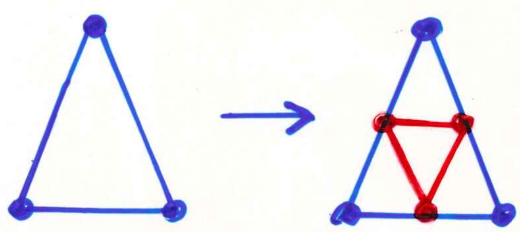
analysis of the FE solution $u_h$ :
a posteriori error estimates
(see Section 2.6)

$\Downarrow$

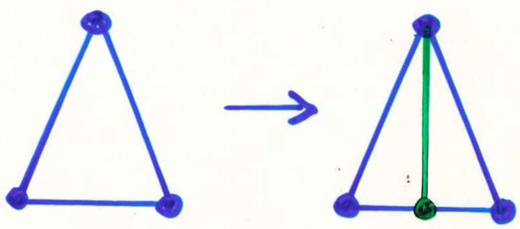
mesh refinement, i.e.
mark the elements where the error is large,
and refine these elements

Mesh refinement techniques for triangles :

red refinement



green refinement
(triangle bisection)



L11-14

11d

# Spezialforschungsbereich F013
# A Cooperation of the Subprojects F1306 and F1309 and F1311

WETGEN

## Geometrical input formats

### CSG

CSG (constructive solig geometry) represents geometry using **primitives combined by boolean operations**. The surfaces of the primitives are described implizitely by nonlinear equations.

**Supported primitives**: Halfspace, cylinder, sphere, cone

```
solid cube =
            plane (0, 0, 0; 0, 0, -1)
        and plane (0, 0, 0; 0, -1, 0)
        and plane (0, 0, 0; -1, 0, 0)
        and plane (100, 100, 100; 0, 0, 1)
        and plane (100, 100, 100; 0, 1, 0)
        and plane (100, 100, 100; 1, 0, 0);

solid all =
            cube
        and sphere (50, 50, 50; 75)
        and not sphere (50, 50, 50; 60);
```

### STL

STL (stereolithography) files are the de-facto standard CAD representation for rapid prototyping. They use **faceted surface representation**, i.e. a **list of triangular surface patches** with no adjacency information.

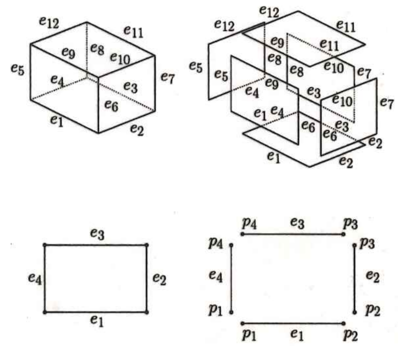
```
solid Solidname
  facet normal 9.838605e-01 3.226734e-02 1.760037e-01
    outer loop
      vertex   -1.070000e+02 0.000000e+00 1.816000e+02
      vertex   -1.060000e+02 0.000000e+00 1.760100e+02
      vertex   -1.070000e+02 1.200000e+00 1.813800e+02
    endloop
  endfacet
  facet normal 9.824255e-01 9.205564e-02 1.623759e-01
    outer loop
      vertex   -1.070000e+02 1.200000e+00 1.813800e+02
      vertex   -1.060000e+02 0.000000e+00 1.760100e+02
      [...]
    endloop
  endfacet
  [...]
endsolid
```

### STEP AP 203

STEP (Standard for the Exchange of Product model data) is an ISO standard. Is was designed as a successor of IGES and VDAFS. AP 203 (Application protocol) uses a **boundary representation**, i.e. a **list of surface patches** defining the boundary of the solid. These are bounded by edges with well-defined starting and ending points. **Additional topological information** (how the faces are joined together) is included.

**Supported surface types**: Plane, cylinder, sphere, cone, torus, sweep and rotatinal surfaces, b-spline and rational b-spline surfaces

**Supported curve types**: Line, circle, ellipse, parabola, hyperbola, b-spline and rational b-spline curve




## Mesh generation features

- **Different elements** supported: Triangles, quadrilaterals; tetrahedra, prisms, pyramids
- **Rule based advancing front mesh generator**: The rules can be specified in form of data structures
- **Surface mesh generation using advancing front methods**: In a trust region around the current segment whose radius is controlled by the geometry, the front is transformed into local 2D-coordinates and the 2D rules are applied.
- **Volume mesh generation** using a combination of **Delaunay's algorithm and advancing front methods**; We use Delauney's algorithm for large parts of the volume and advancing front methods for generating a conforming closure to the boundary mesh.
- **Local mesh size control**: The mesh size is controlled by the local curvature of the geometry.
- **Anisotropic mesh generation** for thin layers
- **Mesh optimization** of surface and volume mesh using
  1. free point relaxation,
  2. point relaxation on edges and surfaces,
  3. edge swapping,
  4. point collapsing,
  5. edge splitting.

## Examples