Graphpartitioning and Loadbalancing

Practical

Procedure for Load Balancer

- Creating the *Element Graph* which is described by the Jostle parameters *degree,edges,nnodes,nedges*

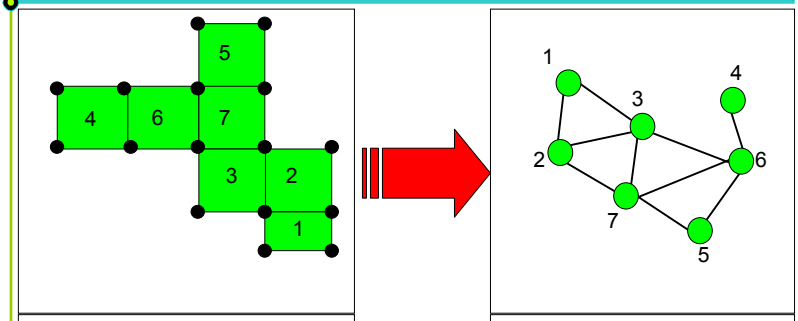
  Element Graph : A Graph whose nodes are the elements of the unstrucured grid and whose edges are between nodes of the element graph only if the elements have one or more nodes in common.

- Description of the parallel network with the array *network*

  network = (0,P,1,1) P = Number of Processors

- In case of variable subdomain weights , the array *processor_wt* is used to give each subdomain a weight.
- If every node and every edge should be weighted differently, these weights are stored in the arrays *node_wt* und *edge_wt* respectively.

From Finite Element Grid to Element Graph



Finite Element Grid | Element Graph

Parameter for Jostle

- nnodes = 7
- offset = 1
- nedges = 20
- dimension = 2 *not used*
- degree = [2 ,3 , 4 ,1,2,4,4]
- edges = [2,3 ,1,3,7, 1,2,6,7 ,6,6,7,3,4,5,7,2,3,5,6]
- node_wt = (int*) NULL *OR* node_wt = [1,1,4,1,1,1,1]
- edge_wt = (int*) NULL *OR* edge_wt = [1,1,1,3,1,1,3,1,1,1,1,1,1,1,1,1,1,1,1,1]
- coords = (double *) NULL *not used*
- partition = [-,-,-,-,-,-,-] *OR* partition = [0,0,0,1,0,1,1]
- network = [0,P,1,1] P = Anzahl Prozessoren
- processor_wt = (int *) NULL
- output_level = 0

Exercise - Preparations

- Make sure, exceed is running on your PC (X-server for NT)
- Before logging in using ssh, please activate the X11 forwarding
- Please log in using the account rzvmpi13 on servint1
  Password is available at the board
- bash
- cd ~/LOADBALANCING_PRACTICAL/#nr
- All files necessary for the exercise are there

Exercise

- Run the program for the domain decomposition in the following way: ./domdec ctif 1
- The result is only the whole domain
- Some output files are generated. The domain is stored in tcplt.dlt
- Start tecplot using: tecplot &
- Load the domain file into tecplot:
  File --> Load DataFile(s)
- Start the domain decomposition program with other domain numbers, (./domdec ctif <number of domains>)
  visulize the resulting domains using tecplot, discuss the result within your group

Exercise (more advanced I)

- Start domdec using an additional parameters for Jostle:
  - usually Jostle allows a load inbalance of 3 %
  - This can be changed by giving domdec an additional parameter:

    ./domdec ctif <number of domains> <allowed load imbalance>

  - try this with several domains and load imbalance values
  - discuss the results

Exercise (more advanced II)

- Start domdec using once more an additional parameters for Jostle:
  - Jostle offers the opportunity to predefine a load vector, i.e. we can specify machines with different performance
  - we predefined a machine array where the first machine has double the performance of the second one and the second one has double performance of the other machines.
  - To use this predefined machine array start domdec in the following way:
    ./domdec ctif <number of domains> <allowed load imbalance> 1

  - try once more several domains and load imbalance values with the machines array activated
  - discuss the results and compare them to the previous ones