Quick Documentation *Tiny*FEM

Version: 1.0.0

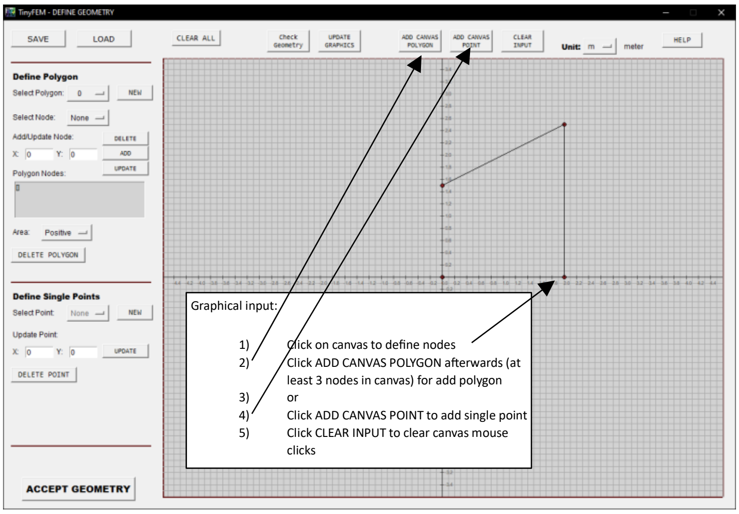
Authors: Elias Perras, Marius Mellmann

Steps

- Create geometry
- Set equation
- Set boundary conditions and material parameters
- Set calculation parameters
- Create mesh
- Solve

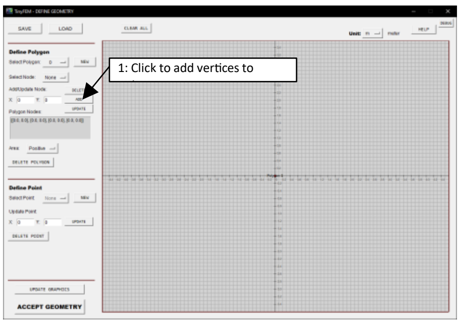
Update Notes:

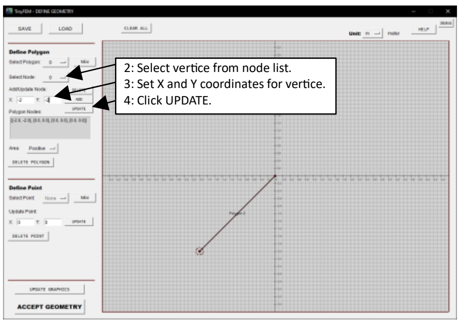
1) Graphical definition for geometry added
2) Geometry compatibility check added

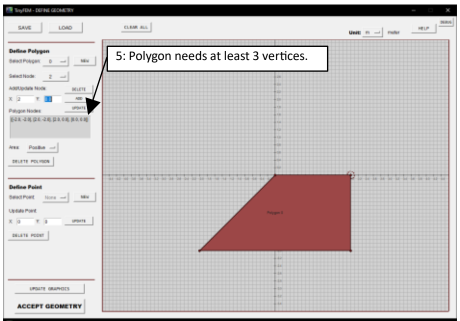


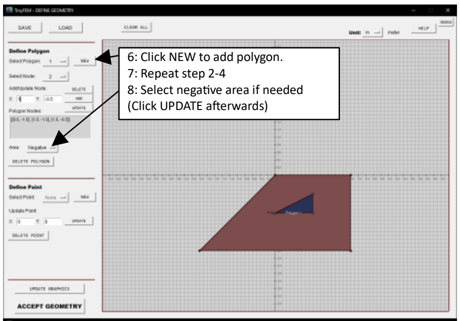
3) Geometry will now automatically be checked for compatibility if click on ACCEPT GEOMETRY or CHECK GEOMETRY

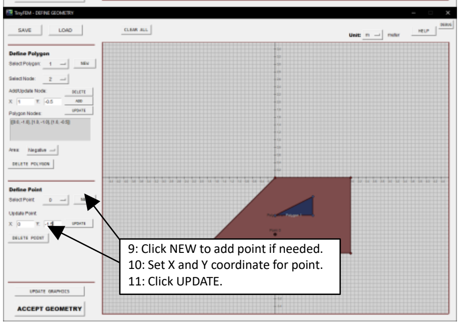
Geometry Creation Guide

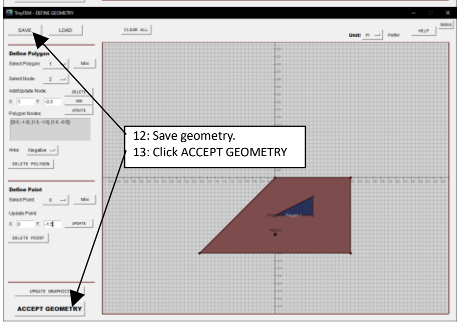








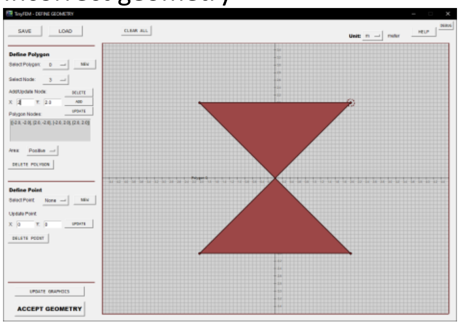




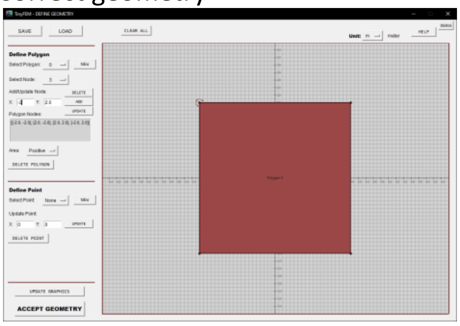
Guideline for compatible Geometry:

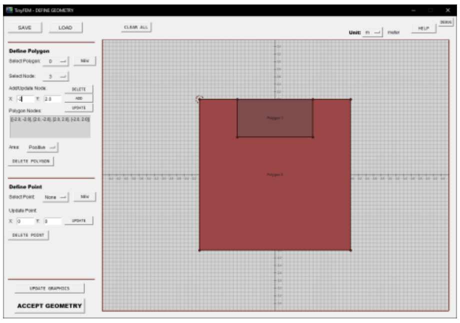
- Nodes must create non-overlapping polygon.
- Positive polygons can not be embedded in positive polygons.
- All nodes of negative polygons must be embedded in a positive polygon.
- Nodes of polygons must be identical.
- Single points must be inside a positive polygon and not on a boundary.
- Polygons must not overlap.
- Polygons and points should be defined inside the canvas dimensions (Valid geometry will still be solved, but you cannot see the geometry in DEFINE GEOMETRY window).

Incorrect geometry

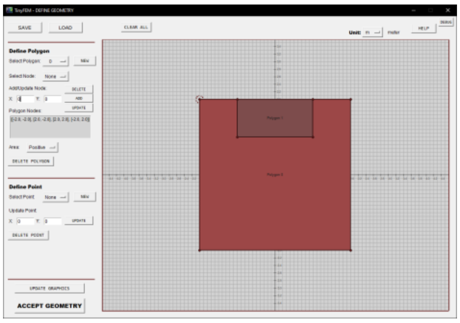


Correct geometry



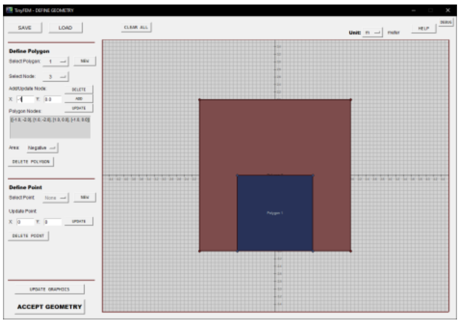


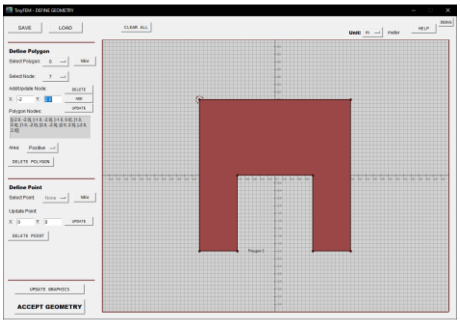
Wrong Polygon nodes for Polygon 0:
`(-2, -2)(2,-2)(2,2)(-2,2)`

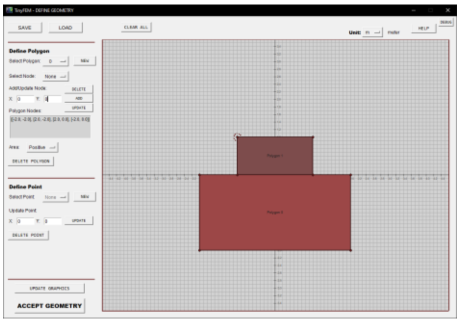


Correct Polygon nodes for Polygon 0:
`(-2, -2)(2,-2)(2,2)(1,2)(1,1)(-1,1)(-1,2)(-2,2)`

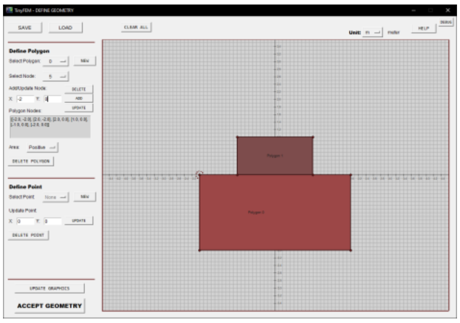
Warning: This does not cause an error but produces an invalid solution!



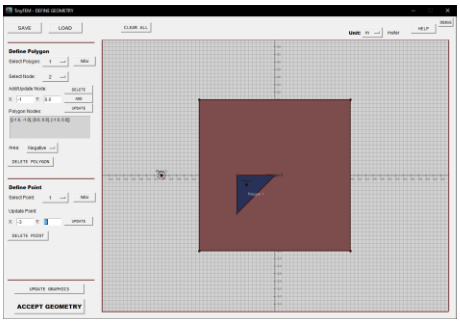


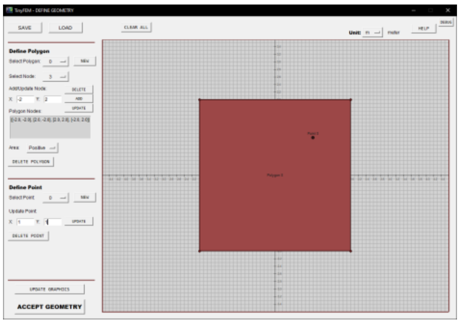


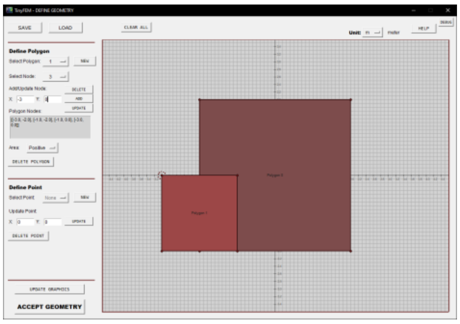
Wrong polygon nodes for polygon 0:
`(-2, -2)(2,-2)(2,0)(-2,0)`
Warning: This does not cause an error but can produce an invalid solution!



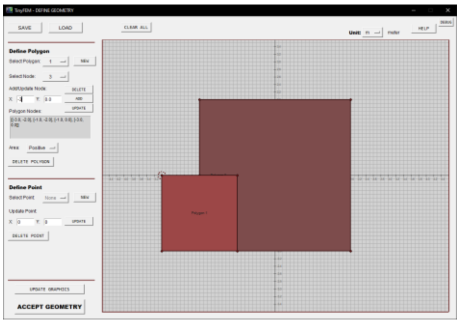
Correct polygon nodes for polygon 0:
`(-2, -2)(2,-2)(2,0)(1,0) (-1,0) (-2,0)`







Wrong polygon nodes for polygon 0:
`(-2, -2)(2,-2)(2,2)(-2,2)`



Correct polygon nodes for polygon 0:
`(-1.0, -2.0), (2.0, -2.0), (2.0, 2.0), (-2.0, 2.0), (-2.0, 0.0), (-1.0, 0.0)`

Boundary Conditions, Material and Calculation Parameters

Set Equation:

- Heat Equation solves the stationary heat equation:
  $\nabla \cdot \boldsymbol{q} = Q \quad \boldsymbol{q} = -k\nabla T$

- Helmholtz Equation solves the Helmholtz equation (Wave equation in the frequency domain for solving acoustic problems):
  $\nabla^2 p + k^2 p = q_F$

Set Boundary Conditions:

Available options depend on selected equation:

- Heat Equation:
  1) Dirichlet boundary conditions: Temperature on boundary. Set temperature T [K].
  2) Neumann boundary conditions: Flux on boundary. Set flux q [W/m²].
  3) Robin boundary conditions: Convective flux on boundary. Set fluid temperature T_{ext} [K] and heat transfer coefficient (Heat TC) h [W/(m²K)].

- Helmholtz Equation:
  1) Dirichlet boundary conditions: Pressure on boundary. Set pressure p [Pa].
  2) Neumann boundary conditions: Impedance on boundary. Set impedance Z [Pa s/m].
  3) Monopol sound source P [W/m].

Set Material Parameters:

- Heat equation: Set conductivity k [W/mk].
- Helmholtz equation: Set speed of sound c [ms/s] and density ρ [kg/m³].

Set Calculation Parameters:

- Set mesh density:
  1) Very Coarse
  2) Coarse
  3) Medium
  4) Fine
  5) Very Fine
- Helmholtz equation: Set frequency [Hz].

Solve the System

- Helmholtz equation: Select either pressure or sound pressure level.

Mesh Density

- Heat equation: Coarse and Medium usually sufficient.
- Helmholtz equation: Mesh Density dependent on wavelength (at least 10 nodes per wavelength). Setting should be at Fine or Very Fine, but for low wavelength (high frequency) the result may be inaccurate or false.

- Calculation time for setting Fine and Very Fine very high due to inefficient code in Python (*Work in progress*)!
- Calculation works with linear elements: Very inefficient for calculation of Helmholtz equation (*Work in progress*)!