POST-PROCESSING CALCULATIONS

HOW TO NUMERICALLY COMPUTE
FREE-SURFACE ELEVATION, VELOCITY, PRESSURE, FORCES

April 2020

DualSPHysics team

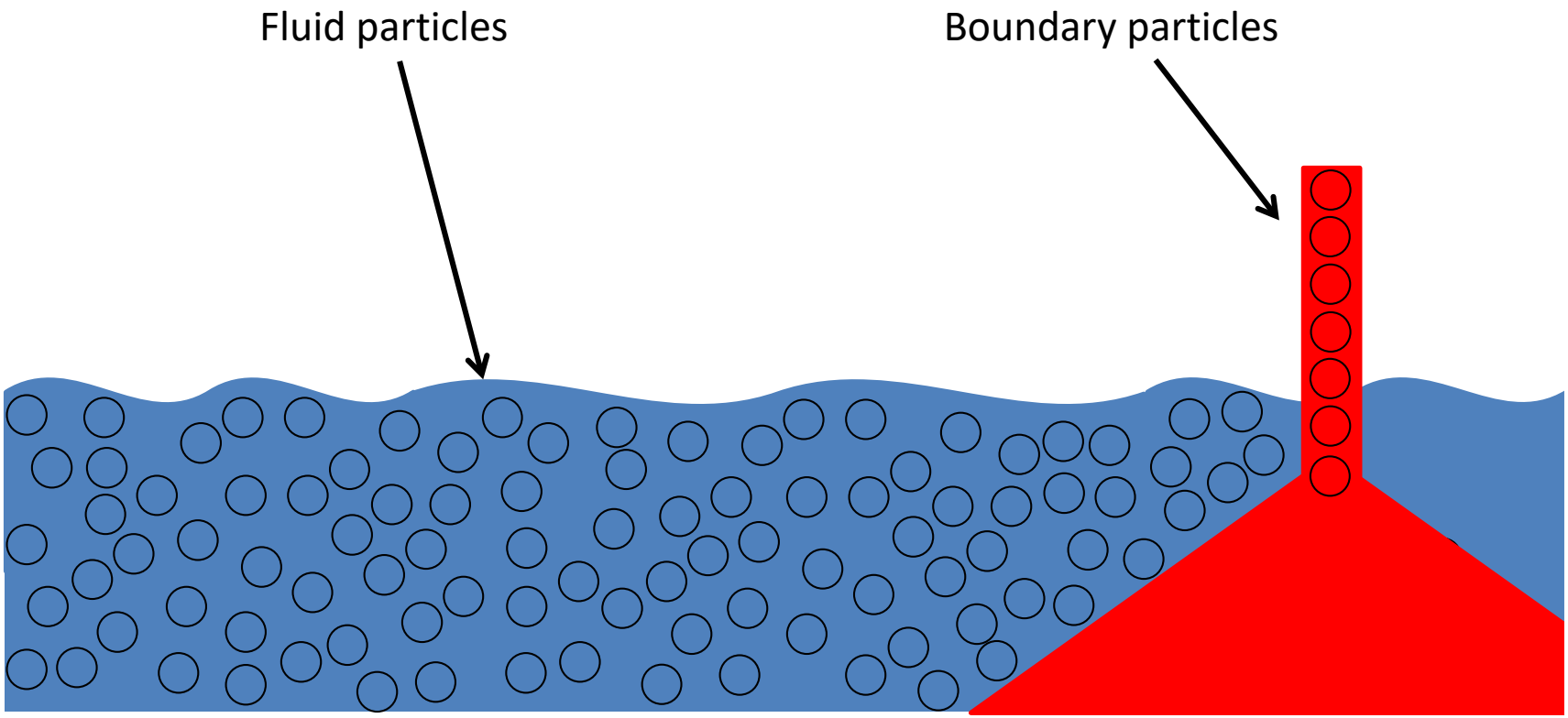
HOW TO NUMERICALLY COMPUTE

VELOCITY

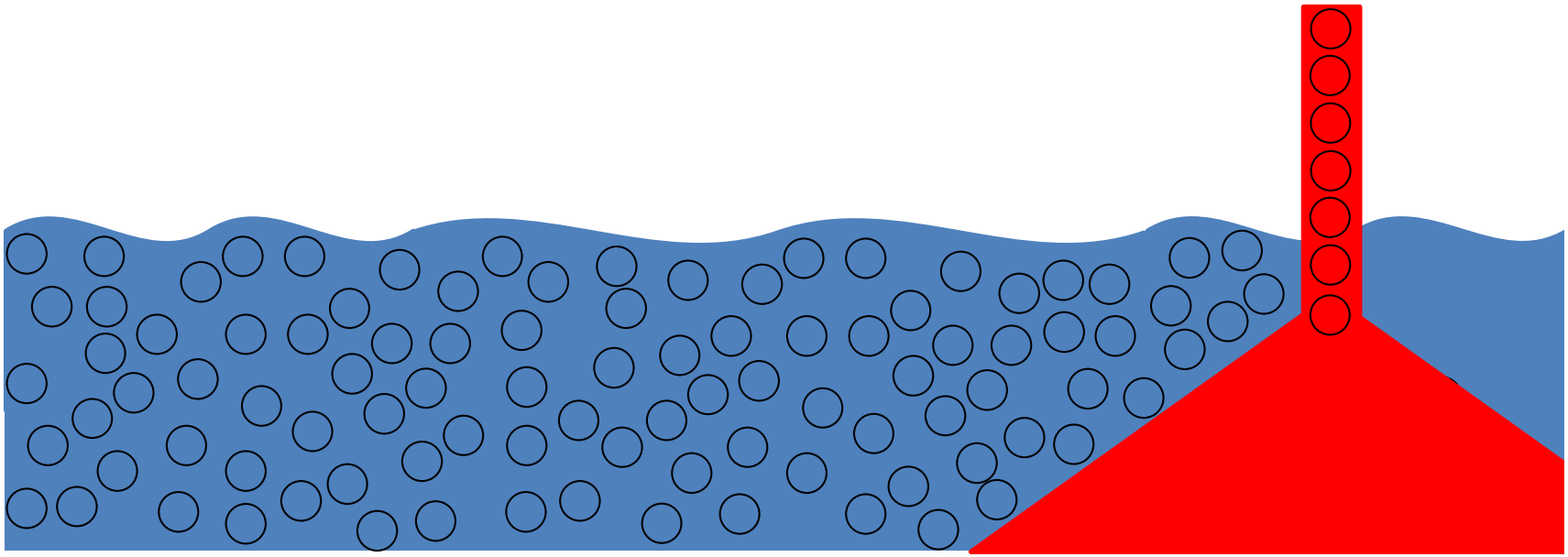
PRESSURE

FORCES

FREE-SURFACE ELEVATION



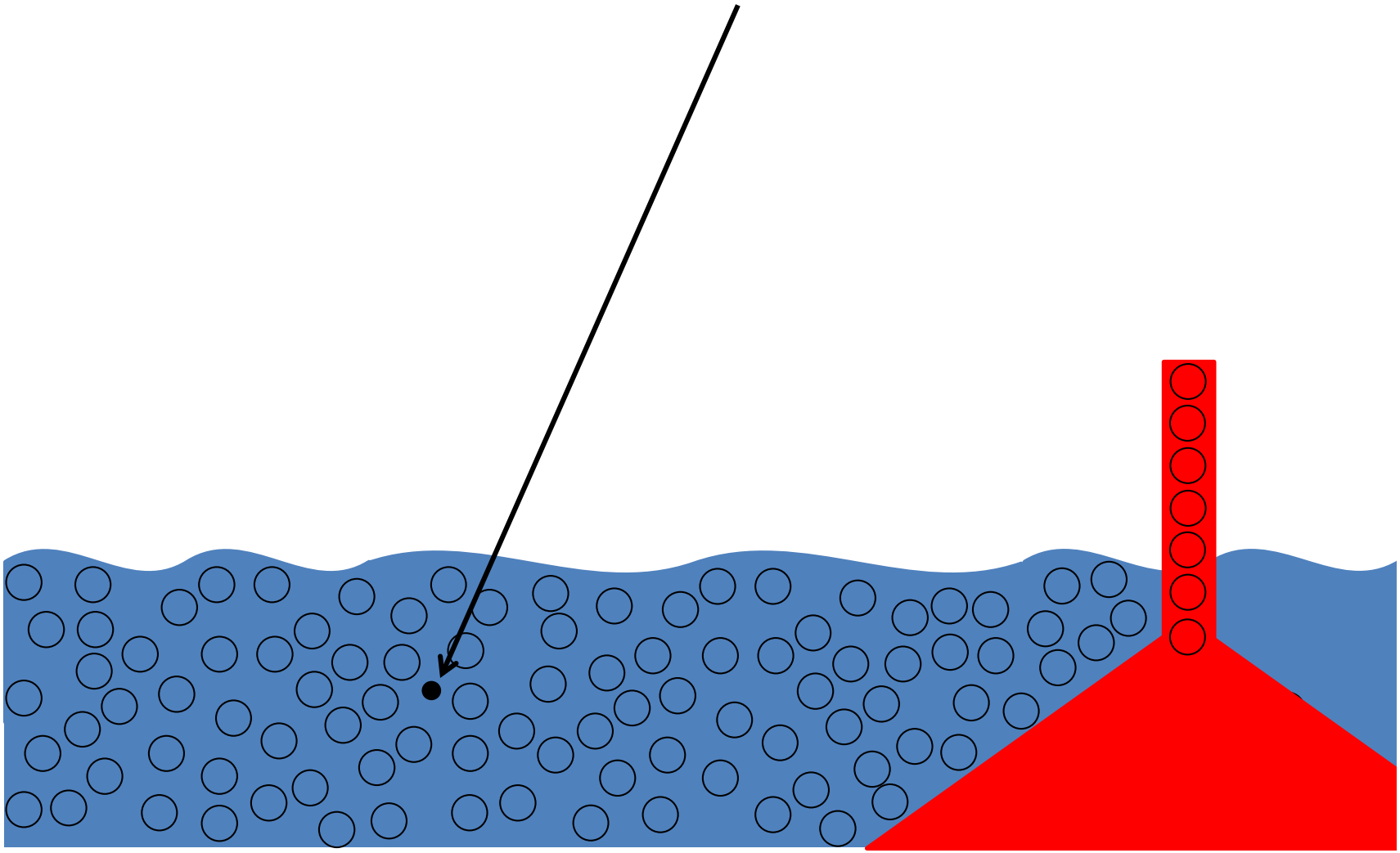
HOW TO NUMERICALLY COMPUTE VELOCITY



HOW TO NUMERICALLY COMPUTE VELOCITY

DualSPHysics

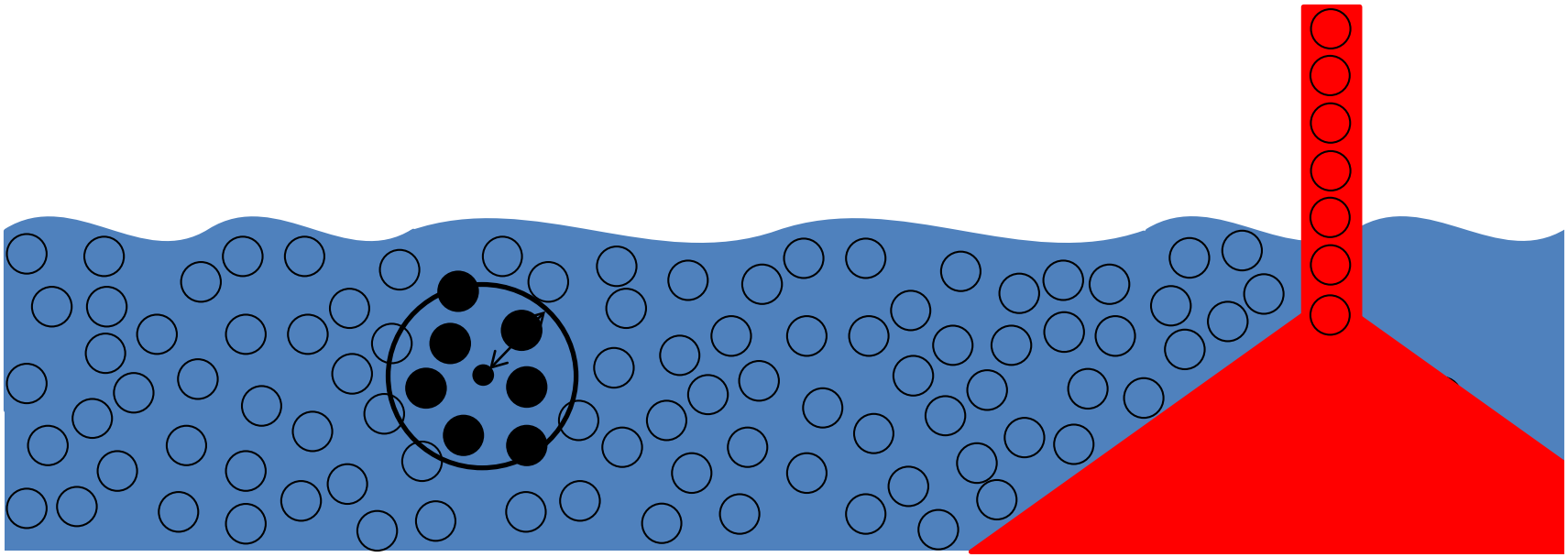
1) For a given location



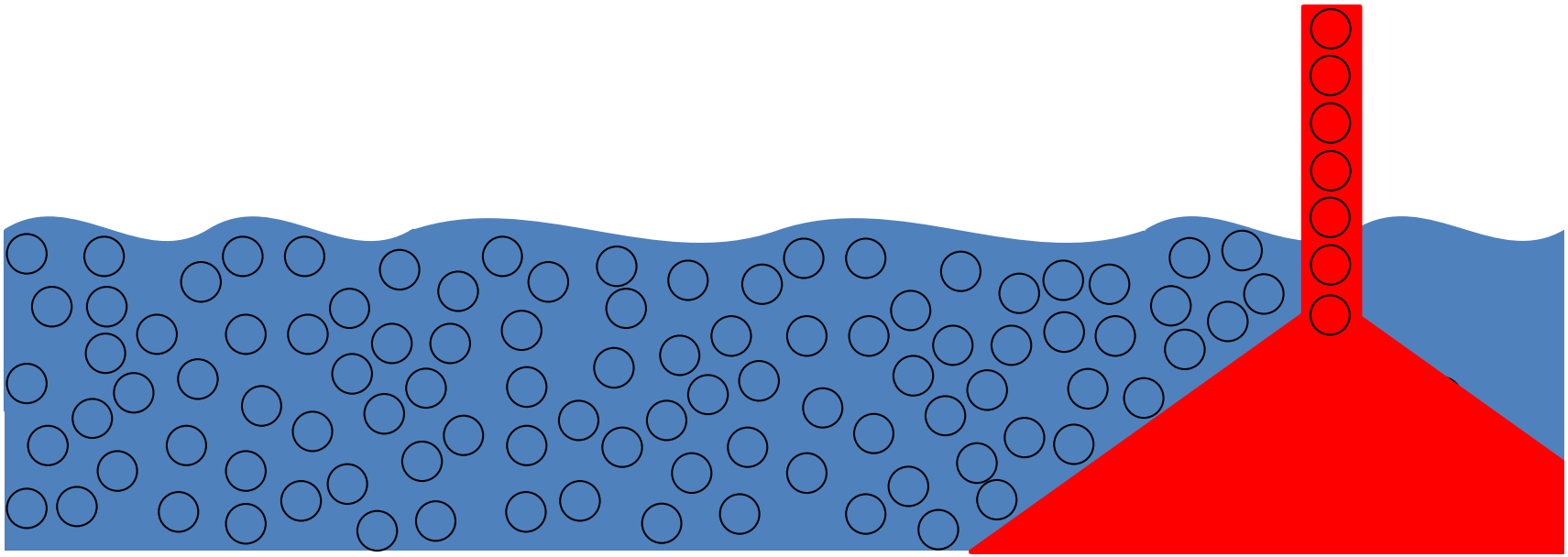
HOW TO NUMERICALLY COMPUTE VELOCITY

1) For a given location
2) We compute numerical VELOCITY
   using VELOCITY values of neighbouring fluid particles

$$\boldsymbol{V}_a = \frac{\sum_b \boldsymbol{V}_b W_{ab}}{\sum_b W_{ab}}$$

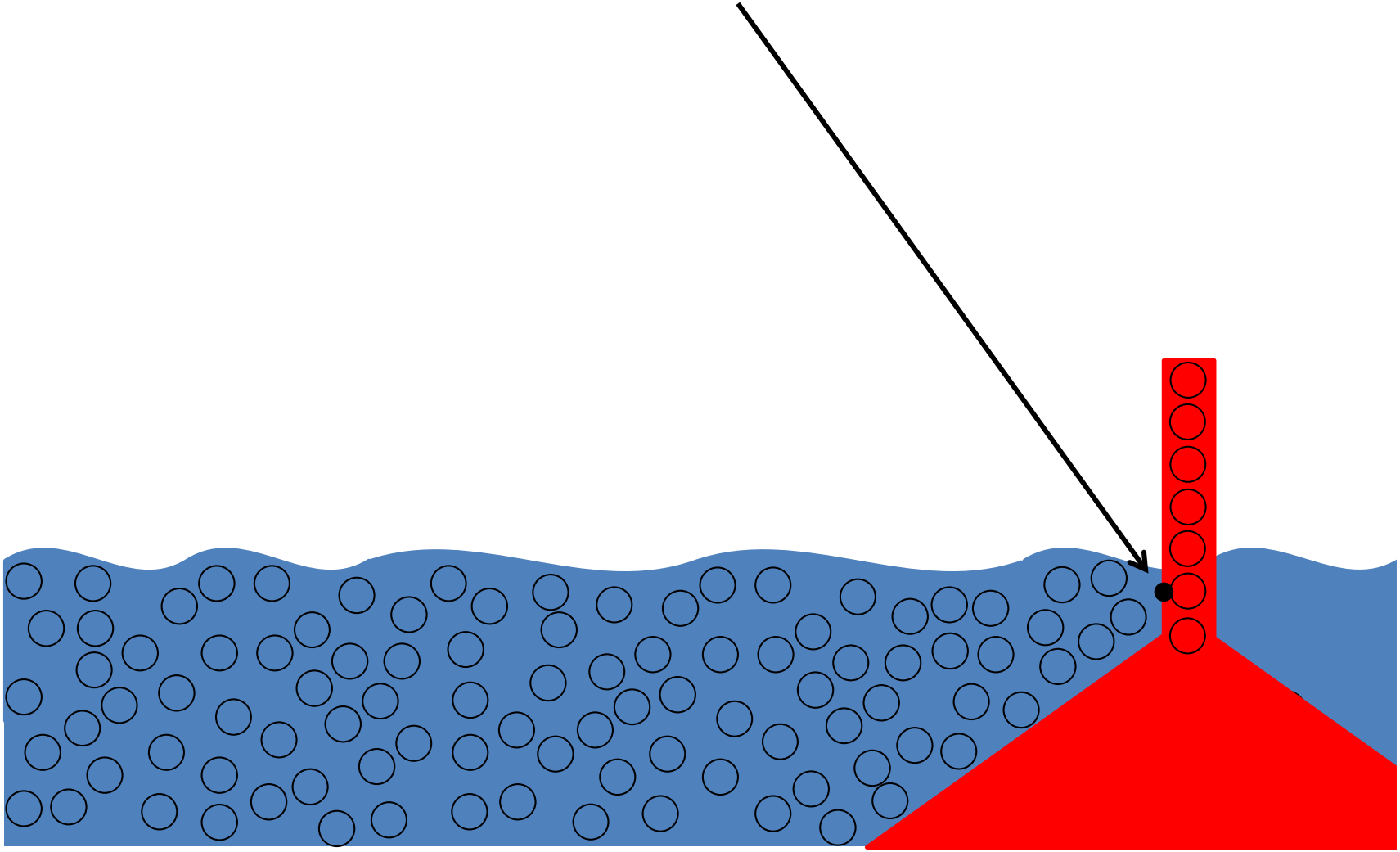


HOW TO NUMERICALLY COMPUTE PRESSURE



HOW TO NUMERICALLY COMPUTE PRESSURE

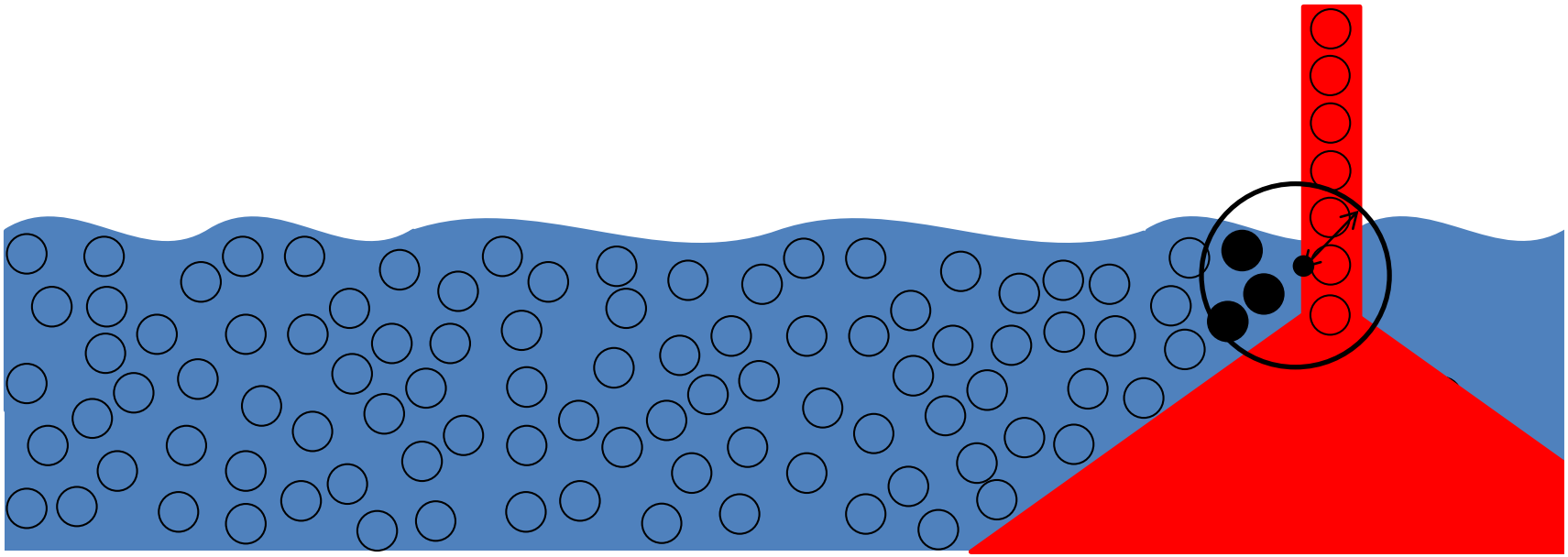
1) For a given location



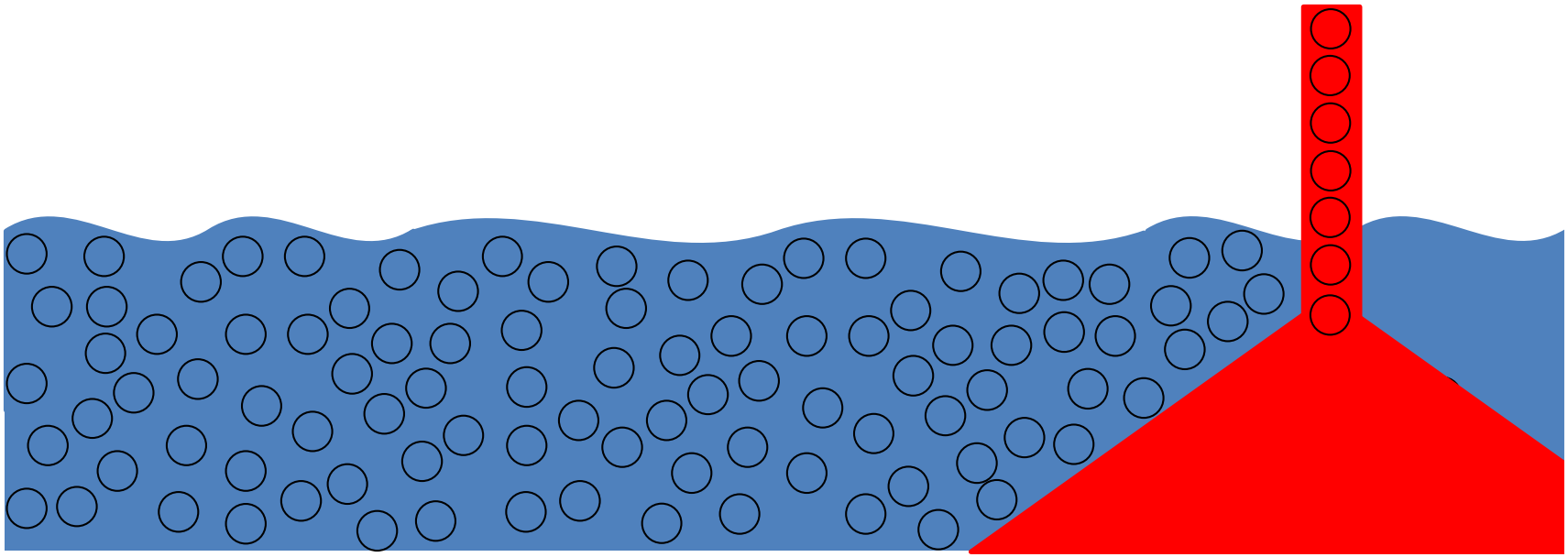
HOW TO NUMERICALLY COMPUTE PRESSURE

1) For a given location
2) We compute numerical PRESSURE
   using PRESSURE values of neighbouring fluid particles

$$\boldsymbol{P}_a = \frac{\sum_b \boldsymbol{P}_b W_{ab}}{\sum_b W_{ab}}$$

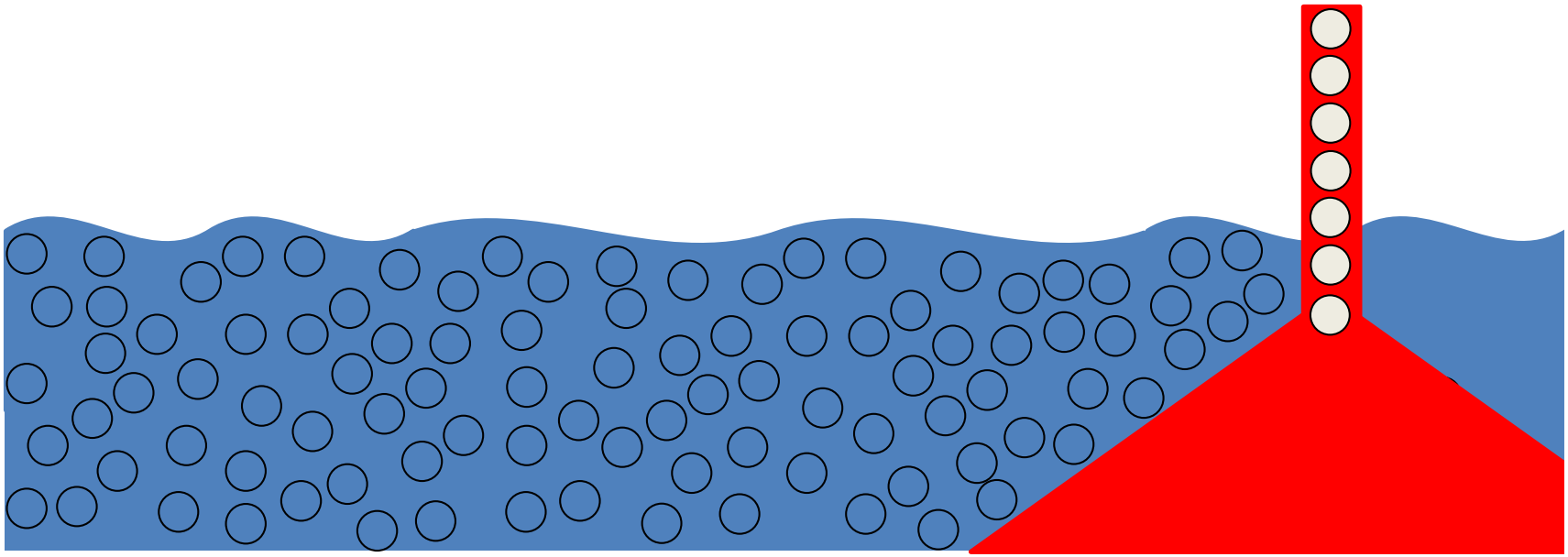


HOW TO NUMERICALLY COMPUTE FORCES



HOW TO NUMERICALLY COMPUTE FORCES

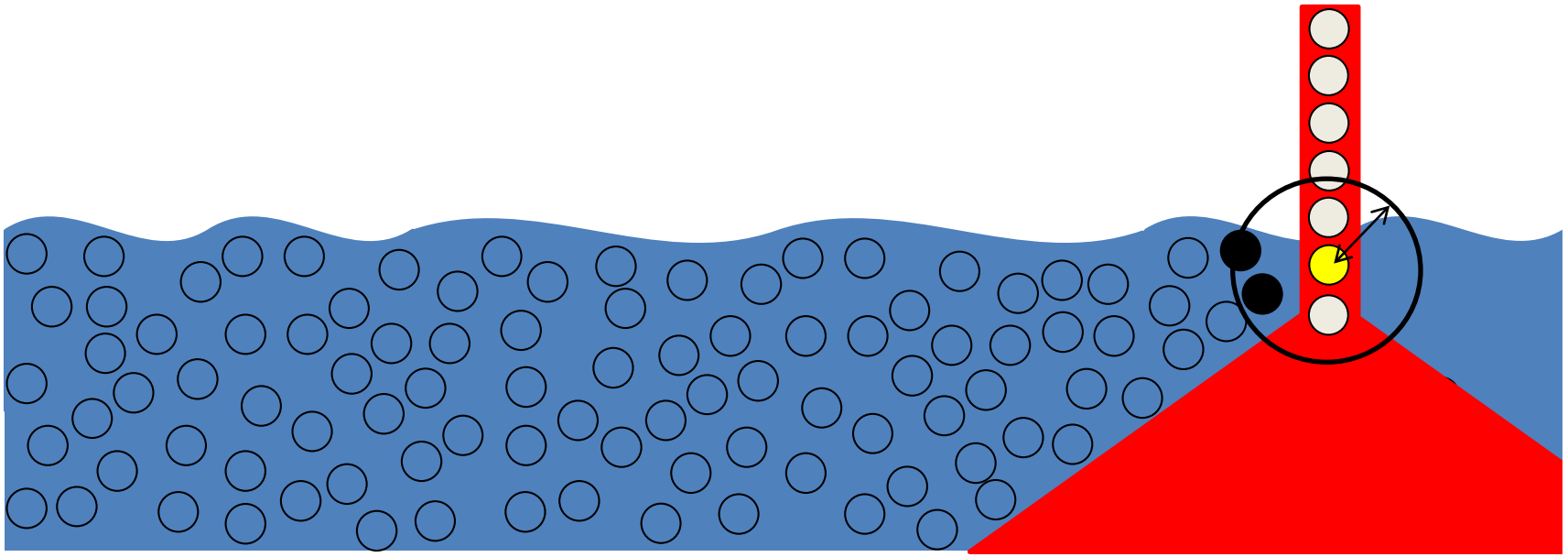
1) For a range of boundary particles



HOW TO NUMERICALLY COMPUTE FORCES

1) For a range of boundary particles
2) We compute numerical ACCELERATION of those boundary particles solving the particle interactions with fluid neighbouring particles

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \nabla_a W_{ab} + \mathbf{g}$$



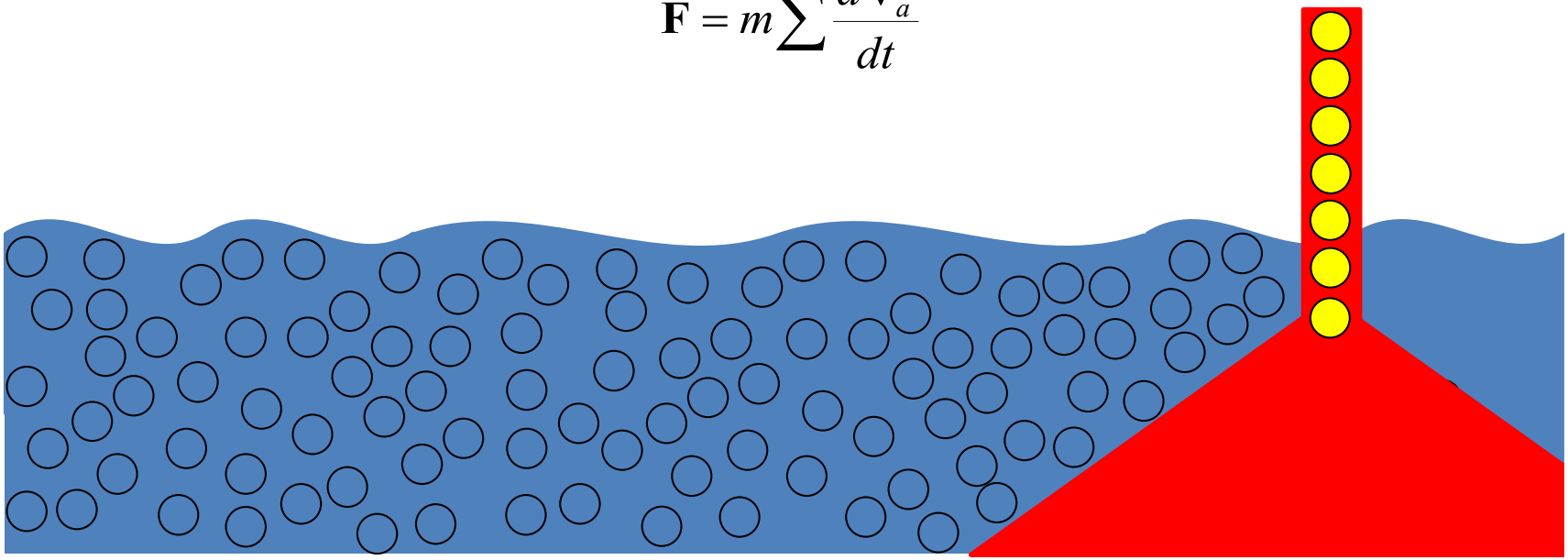
HOW TO NUMERICALLY COMPUTE FORCES

1) For a range of boundary particles
2) We compute numerical ACCELERATION of those boundary particles solving the particle interactions with fluid neighbouring particles

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \nabla_a W_{ab} + \mathbf{g}$$

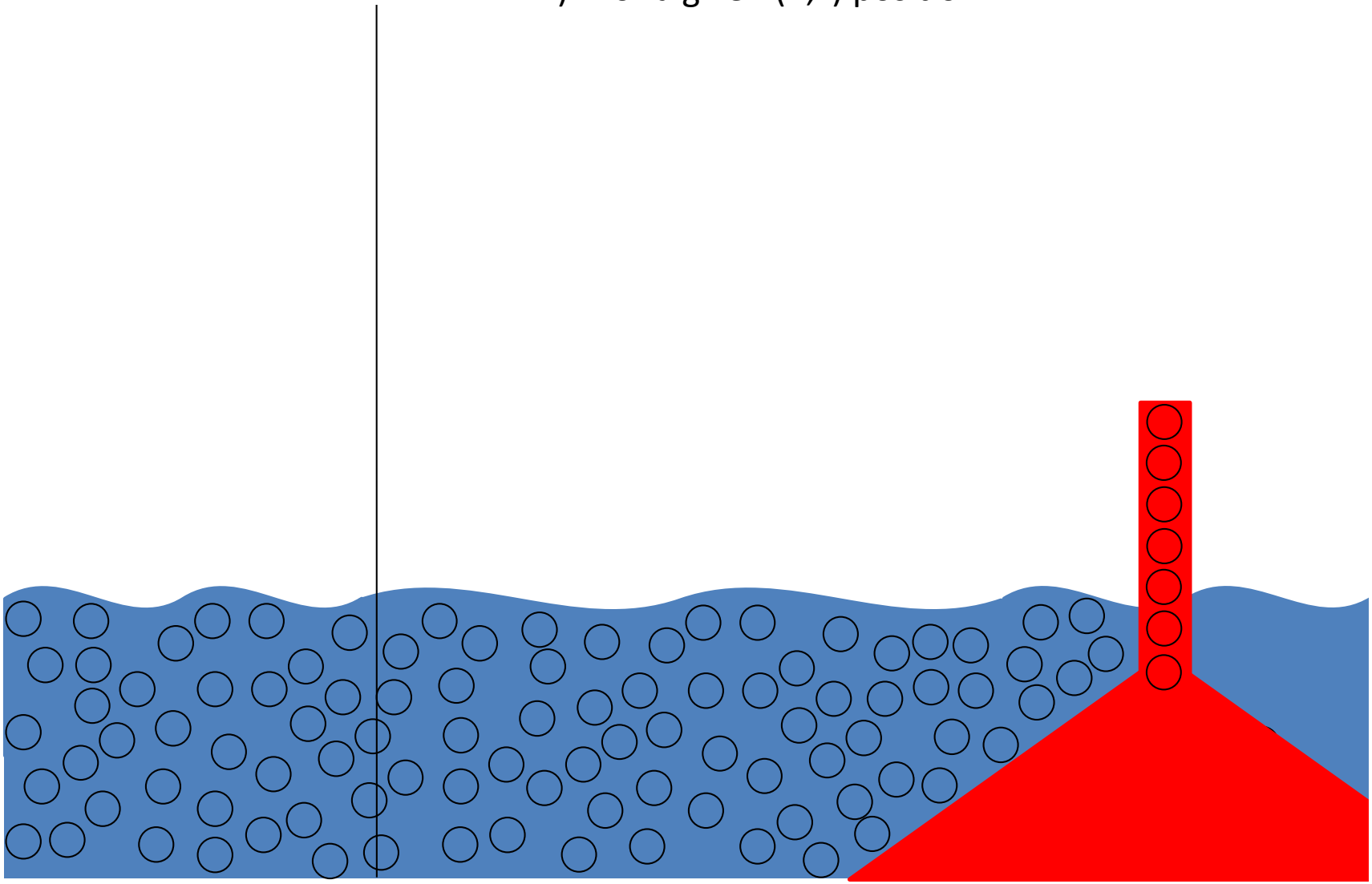
3) We do the summation of ACCELERATION values of those boundary particles

$$\mathbf{F} = m \sum \frac{d\mathbf{v}_a}{dt}$$



HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

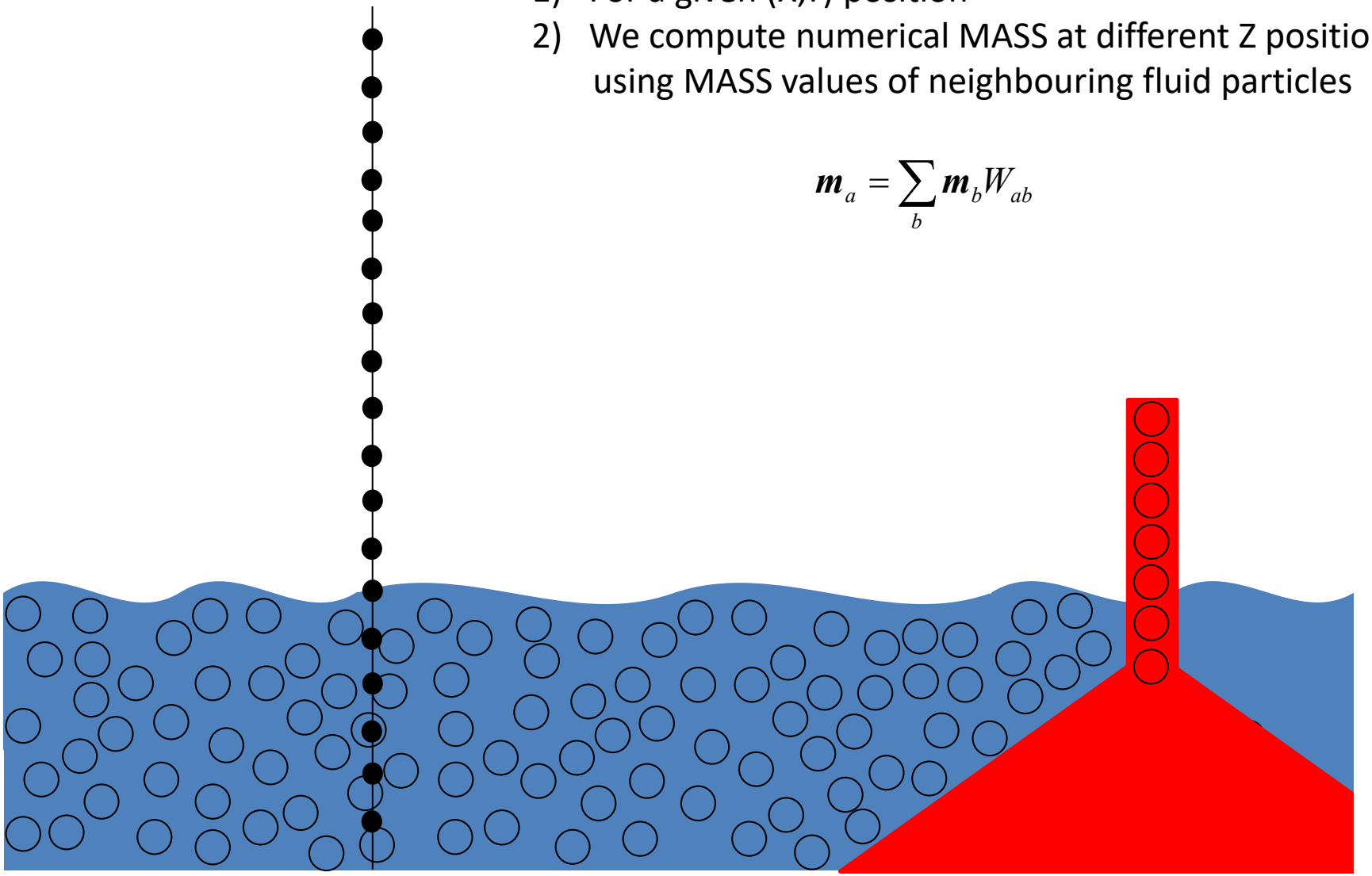
1) For a given (X,Y) position



HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

1) For a given (X,Y) position
2) We compute numerical MASS at different Z positions using MASS values of neighbouring fluid particles

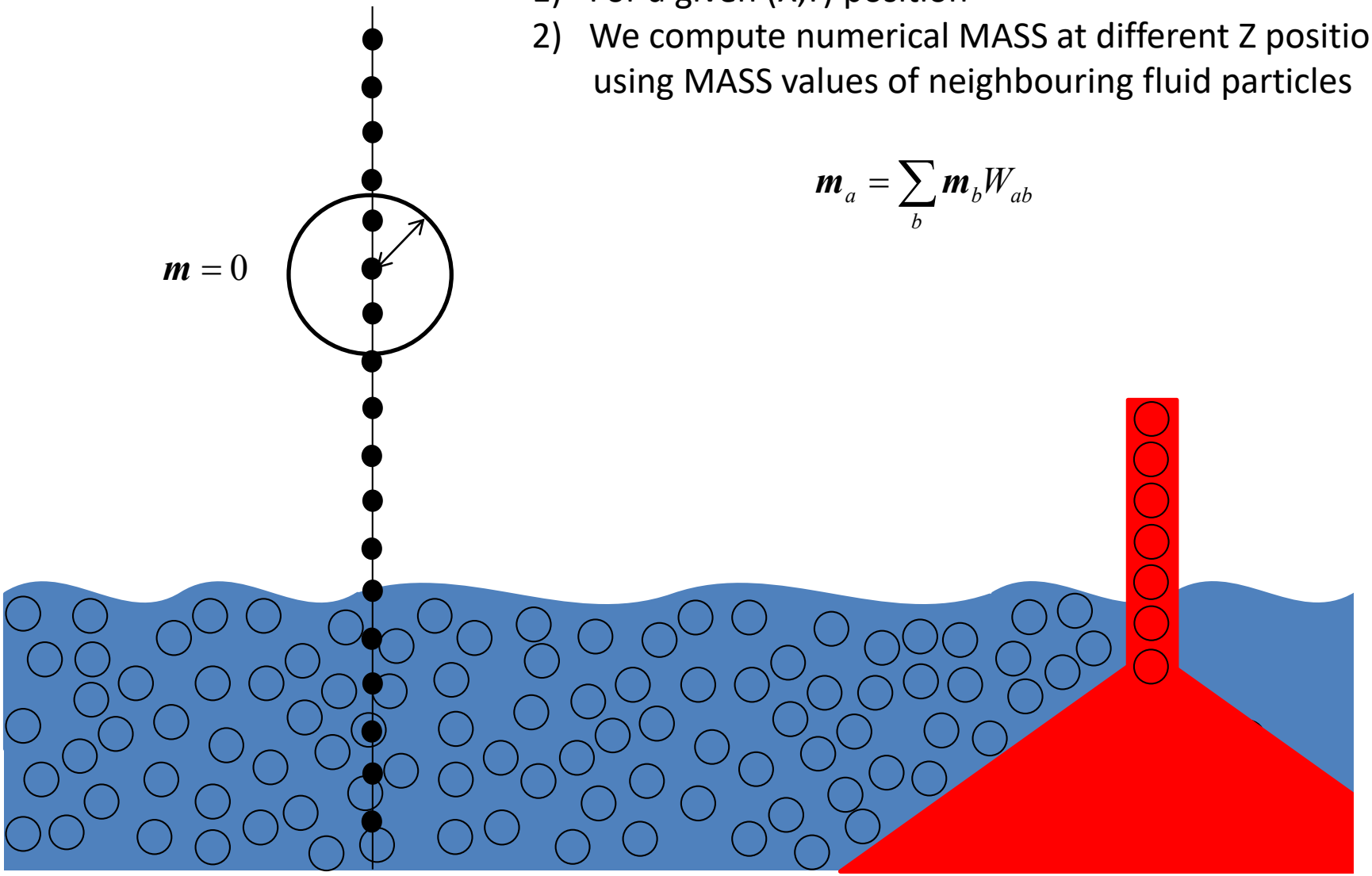
$$\boldsymbol{m}_a = \sum_b \boldsymbol{m}_b W_{ab}$$



HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

1) For a given (X,Y) position
2) We compute numerical MASS at different Z positions using MASS values of neighbouring fluid particles

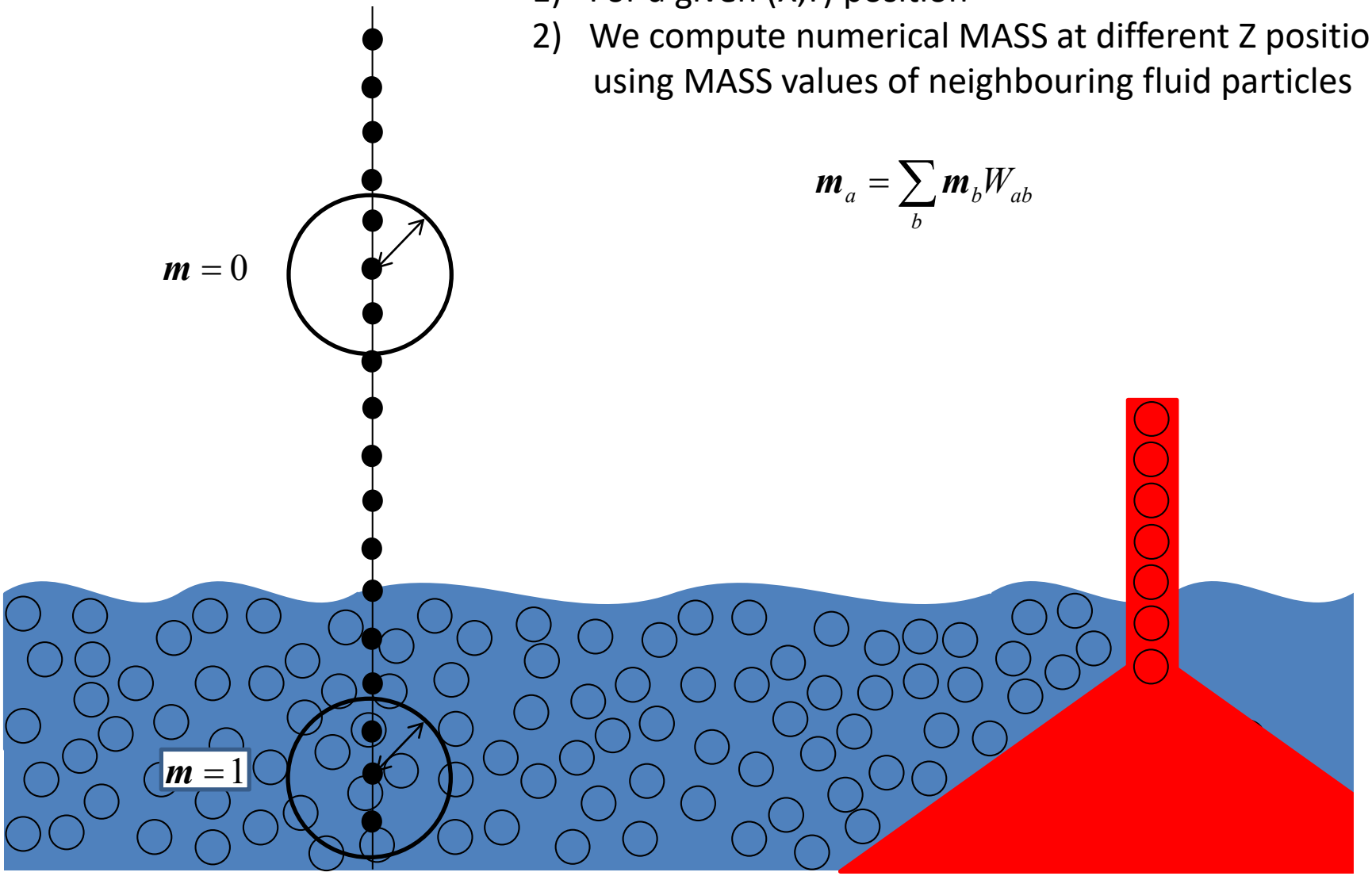
$$m_a = \sum_b m_b W_{ab}$$



HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

1) For a given (X,Y) position
2) We compute numerical MASS at different Z positions using MASS values of neighbouring fluid particles

$$m_a = \sum_b m_b W_{ab}$$

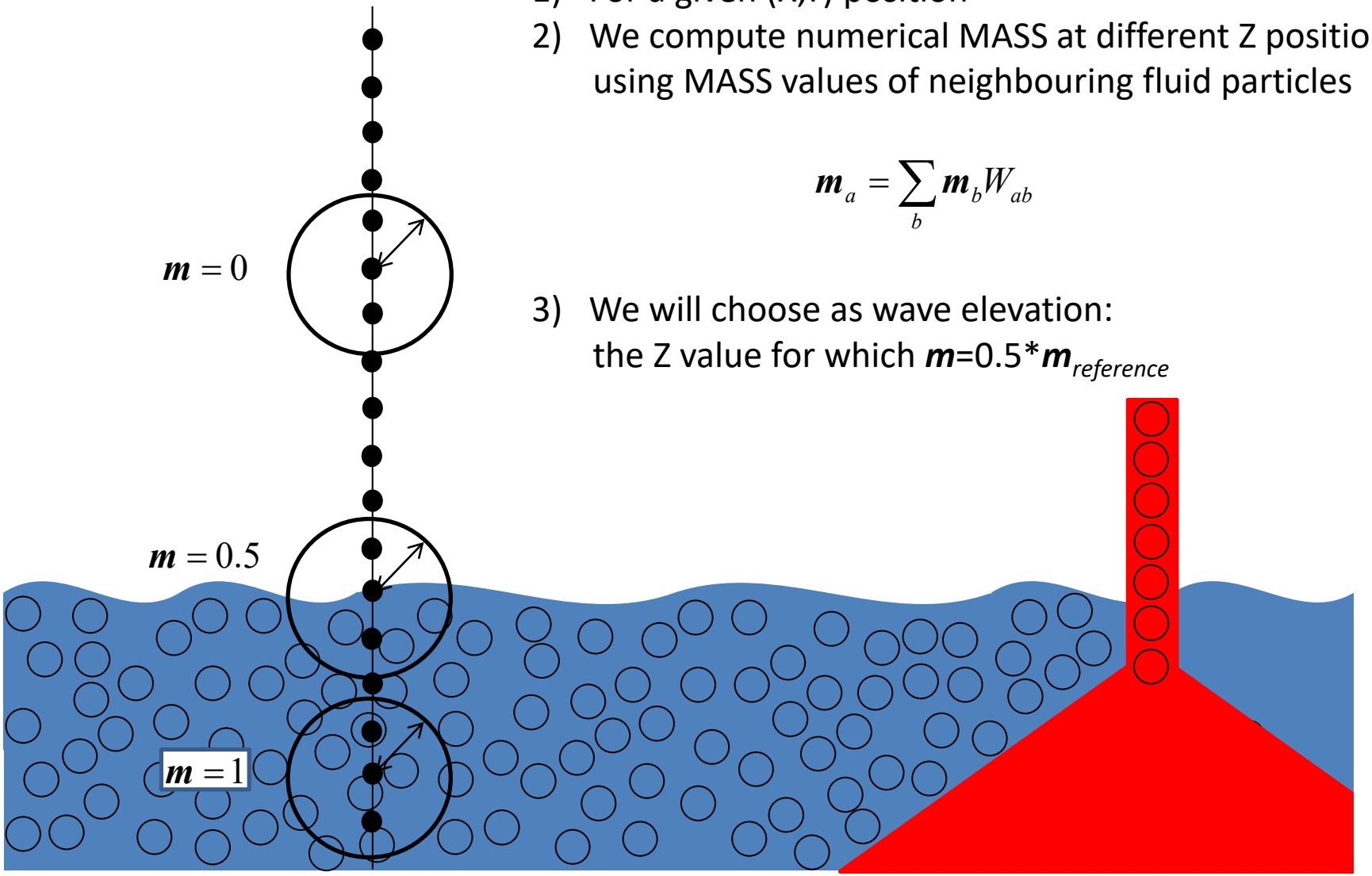


HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

1) For a given (X,Y) position
2) We compute numerical MASS at different Z positions using MASS values of neighbouring fluid particles

$$m_a = \sum_b m_b W_{ab}$$

3) We will choose as wave elevation: the Z value for which m=0.5*$m_{reference}$

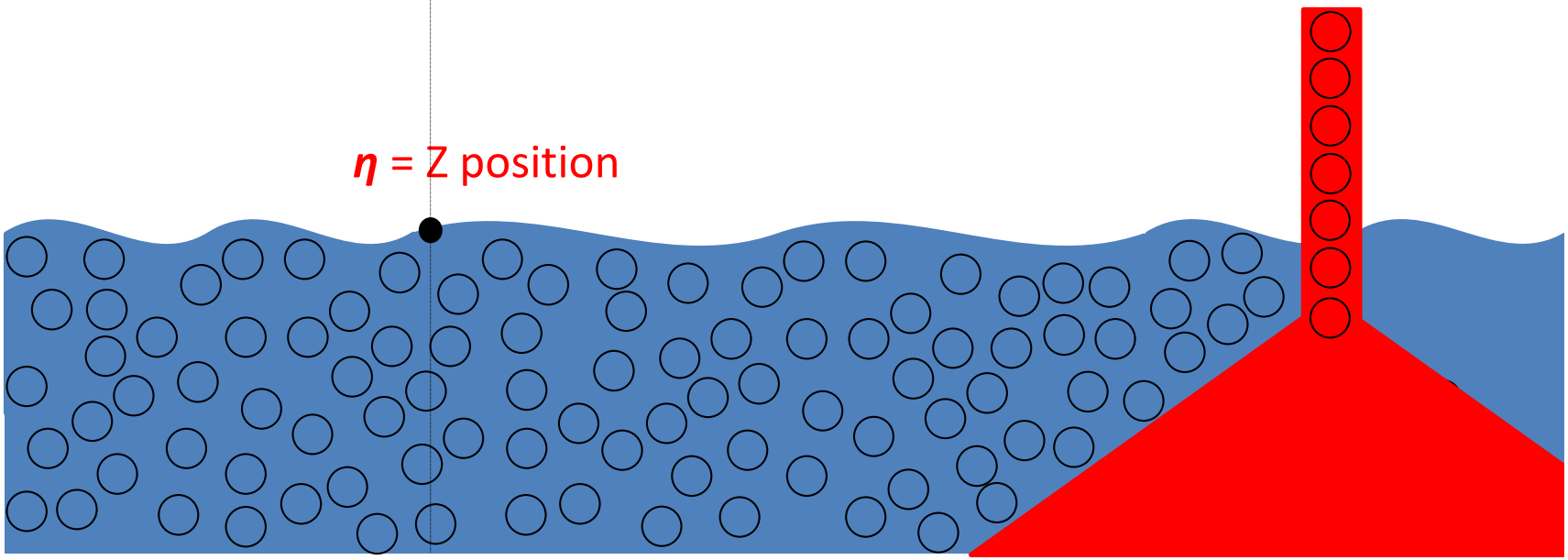


HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

1) For a given (X,Y) position
2) We compute numerical MASS at different Z positions using MASS values of neighbouring fluid particles

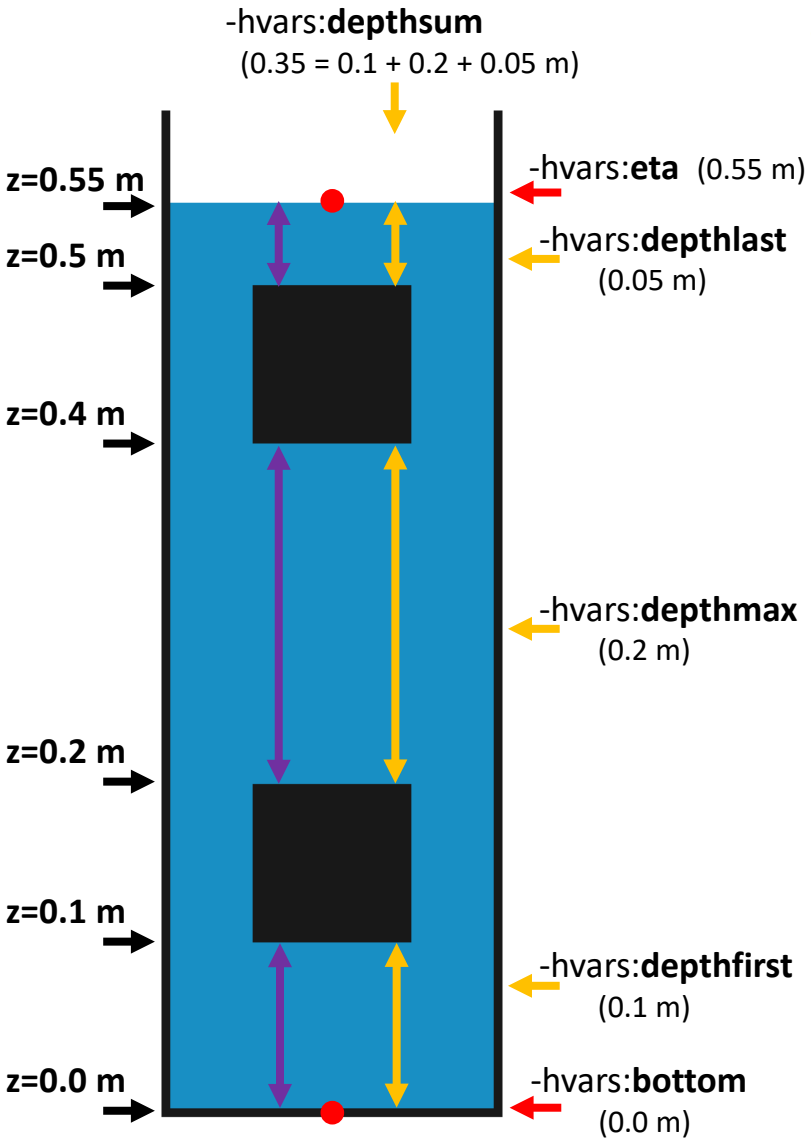
$$\boldsymbol{m}_a = \sum_b \boldsymbol{m}_b W_{ab}$$

3) We will choose as wave elevation: the Z value for which $\boldsymbol{m}=0.5*\boldsymbol{m}_{reference}$



HOW TO NUMERICALLY COMPUTE FREE-SURFACE ELEVATION

Options for water level and depth calculation in MeasureTool




```
-hvars[:<values>] Defines height values to be computed
                  (+ means include, - means do not include)
   +/-all:        To choose or reject all options
   +/-eta:        Elevation for each column x,y (equal to -elevation)
   +/-bottom:     Fluid bottom for each column x,y
   +/-depthsum:   Sum of depths for each column x,y
   +/-depthmax:   Maximum depth for each column x,y
   +/-depthlast:  Last depth (near surface) for each column x,y
   +/-depthfirst: First depth (near bottom) for each column x,y
   +/-depthrhop:  Total depth calculated by integral of density at x,y
```

$$DepthRhop = \sum_{i}^{n} \frac{\rho_i dz}{\rho_0} \approx DepthSum$$

n measuring points separated by *dz*