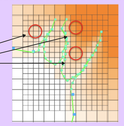
Diagram

Local VEGF - VEGFR binding via ODEs

Each cell edge segment models binding reactions with its local environment (or cellular junction) via its own set of localised ODEs



Inputs

VEGF = VEGF of grid site(s) that edge's nodes are located within.
VEGFR = edge's unbound VEGFR value
VEGF.VEGFR = edge's VEGF-VEGFR complex value

Reaction equations

v1 (forward reaction of VEGF binding to VEGFR):
k1 * VEGF * VEGFR
v2 (inverse reaction of VEGF.VEGFR complex dissociation):
k-1 * VEGF * VEGFR

dxdt

VEGF ' = v2 - v1
VEGFR ' = v2 - v1
VEGF.VEGFR ' = v1 - v2

Solver

Runge-Kutta Cash-Karp method via Boost's Odeint library.

When signalling, each edge interacts with the grid sites that its nodes are contained within.

Sum of VEGF from grid site(s) nodes lie within is used as VEGF input to ODEs.

Unbound VEGFR and active VEGFR of the edge are also input.

Proteins in both edges and grid are stored in amounts. The inputs to the ODE solver are converted to concentrations using the edge's length.

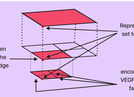
Run ODE solver

Output converted back to amounts from concentrations using edge's length.

Edge unbound VEGFR / active VEGFR set to these new amounts.

Grid site VEGF reduction:

The factor by which VEGF has been changed from ODE is applied to the VEGF level of the grid site(s) the edge is interacting with.



Represent same points so VEGF level is set to same level (determined by ODE)

More fine-grained grid sites encompassed by grid site that has had VEGF changed also have VEGF change factor applied to their VEGF level

Check remeshing

Edge VEGF.VEGFR > activation threshold?

true → Increment edge's remesh timer

remesh timer > remesh threshold AND coarseness level > min

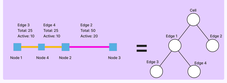
Remeshing

Edges are stored in a tree structure with the cell being the root.

When an edge remeshes it creates two new edges at a finer coarseness level and becomes the parent of these edges in the tree structure.

Protein total and active amounts are split between the two child edges.

The parent edge becomes invalid and is only used as a placeholder in the tree structure. It becomes valid again when it unmeshes and the child edges are deleted.



Grid Remeshing

Edge remeshing also triggers remeshing of grid sites to the same level of granularity as the edge at the edge's location.

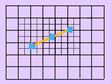
The grid stores sets of locations at different coarseness levels in a tree structure, with the root node being the most coarse-grained grid locations that always cover the entire simulation environment.

Root of grid tree contains most coarse-grained locations

Finer grid sites are created in the area surrounding the newly remeshed edge, these are represented by a new node in the grid tree.

The parent grid locations are required to have their entire area covered by finer grid sites due to diffusion PDE requirements. They cannot be only partially filled with finer grid sites.

Edge's are also required to have grid sites in their nodes' surround moore neighbourhood for signalling purposes.



Root of grid tree contains most coarse-grained locations

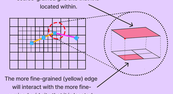
Child node representing set of more fine-grained locations created by remeshing of cell membrane edge.

Finer grid sites are created in the area surrounding the newly remeshed edge, these are represented by a new node in the grid tree.

The parent grid locations are required to have their entire area covered by finer grid sites due to diffusion PDE requirements. They cannot be only partially filled with finer grid sites.

Edge's are also required to have grid sites in their nodes' surround moore neighbourhood for signalling purposes.

The more coarse-grained (purple) edge will interact with the more coarse-grained grid site that it is located within.



The more fine-grained (yellow) edge will interact with the more fine-grained grid site that it is located within.

Edge update

If remeshed then call for each child edge

Remeshed?

Not remeshed → Signalling → Check if needs to be remeshed → Check if needs to be unmeshed

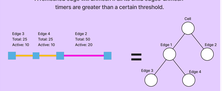
Increment edge's unmesh timer

Edge VEGF.VEGFR > activation threshold?

false → remeshed?

true →

A remeshed edge will unmesh if all its child edges' unmesh timers are greater than a certain threshold.



The child edges will be deleted and the parent edge will become valid again. The parent's protein values are set to the sum of the child edges' proteins.

Cell membrane update

Update edges → Make edge movements → Hooke's law spring force on nodes → Check edge within min / max length range

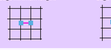
Edge is remeshed? (If remeshed then call for each child edge) — Not remeshed → Edge is a retracting tip node?

Is retracting tip → Check if retraction finished

Not retracting tip → At each coarseness level, an edge's length must stay within a certain range dictated by the specific coarseness level that it is within.

Grid squares at each coarseness level have a fixed side length. An edge's length must be >= and < 2 * this length.

Edge minimum length: Edge maximum length:



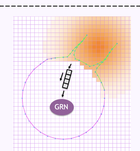
Cell update

Update membrane → Update protein levels → Gene regulation network

Simulation Timestep

Update cells → Update grid

Sum active protein levels from cell's edges → Push aggregate active protein levels onto FIFO stacks to model temporal protein translocation to and from nucleus



Read active protein levels by popping values from protein translocation queues → Run cell level ODE → If total VEGFR value has changed, update all edge total VEGFR values so that cell VEGFR is evenly distributed along membrane → The factor by which active VEGFR has been changed from ODE is applied to the VEGFR level of all edges. This is an attempt to maintain the locality of current VEGFR activation.

Inputs

VEGFR = cell total unbound VEGFR value
VEGF.VEGFR = VEGF-VEGFR complex value popped from top of translocation queue

Constants

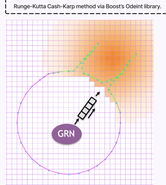
Φ = Degradation rate
Y = Production rate

dxdt

VEGFR ' = -Φ * VEGFR) + Y
VEGF.VEGFR ' = -Φ * VEGF.VEGFR)

Solver

Runge-Kutta Cash-Karp method via Boost's Odeint library.



Grid update

VEGF diffusion modelled via PDEs → Remove inactive grid sections

Diffusion of VEGF in the environment is modelled as time–dependent PDEs; letting u(x, y, z, t) denote the concentration of VEGF at spatial location (x, y, z) and time t, and given the positive diffusion coefficient α, the distribution of VEGF over time is described by Equation 1.

The solution to this equation is approximated using the finite–difference method Forward Time Centered Space (FTCS). FTCS is an explicit scheme, meaning that the solution at the next time step is calculated as a function of the solution at the current time step (Equation 2).

The solution at a single point in three–dimensional space at time step n + 1 is computed from the values of the solution at at time step n at the point itself and six of its neighbouring points (Equation 3); this is an example of a seven–point stencil computation.

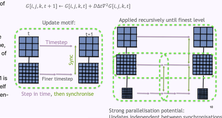
$$\frac{\partial u}{\partial t} = \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) \quad (1)$$

$$U(t+\Delta t) = f(U(t)) \quad (2)$$

$$u^{n+1}_{x,y,z} = u^n_{x,y,z} + (-6r)u^n_{x,y,z} + r(u^n_{x+1,y,z} + u^n_{x-1,y,z} + u^n_{x,y+1,z} + u^n_{x,y-1,z} + u^n_{x,y,z+1} + u^n_{x,y,z-1}) \quad (3)$$

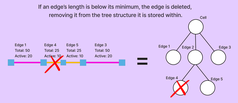
Time step:

$$G[i,j,k,t+1] \leftarrow G[i,j,k,t] + D\delta t\nabla^2 G[i,j,k,t]$$



Edge deletion

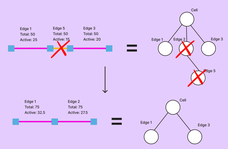
If an edge's length is below its minimum, the edge is deleted, removing it from the tree structure it is stored within.



Protein totals are always evenly distributed across the cell membrane. An edge being deleted causes protein totals to be redistributed across the deleted edge's sibling edges in the tree structure. This redistribution will be applied down the tree structure if any sibling edges have child edges.

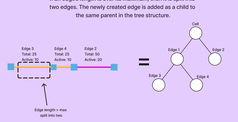
Active proteins of the deleted edge are redistributed to local edges.

If a parent edge has its last child edge deleted, then the parent edge is also removed.



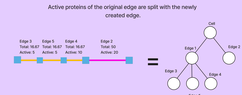
Edge splitting

If an edge's length is over its maximum, then it is split into two edges. The newly created edge is added as a child to the same parent in the tree structure.



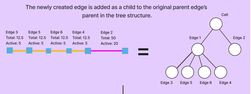
Protein totals are always evenly distributed across the cell membrane. An edge being split causes protein totals to be redistributed across the split edge's sibling edges in the tree structure. This redistribution will be applied down the tree structure if any sibling edges have child edges.

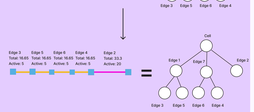
Active proteins of the original edge are split with the newly created edge.



A parent edge can have a maximum of 4 child edges. When it reaches this maximum, the parent edge is split into two with half of its child edges being moved to the newly created edge.

The newly created edge is added as a child to the original parent edge's parent in the tree structure.





Edges splitting when going above their maximum length is crucial to how filopodia form in the model.

As a tip node extends, it elongates the connected tip edges. When these edges become too long, they split into two leaving behind a 'stem' edge.

A stem edge is focally adhered to its environment so its nodes cannot move. New filopodia cannot be initiated from any edge already in a filopodia.

Hooke's law spring forces

Adhered? — Not adhered → Apply spring forces acting on node and update position accordingly.

The force **F** on a node **P** of its S Springs is given by:

$$F = \sum_{i=0}^{S} k\,(\overline{PN_i} - \overline{SN_i}),$$

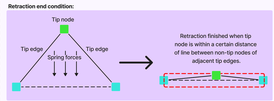
where **N** is the vector position of the neighbour node at the other end of the spring and **k** is the specific spring constant

Currently, only tip nodes have the ability to become unadhered. This allows a filopodia to retract via spring forces pulling it back towards the adjacent edge's in the filopodia stem.

When a tip edge's length goes belong its minimum for its specific coarseness level, the tip edge will be deleted and the adjacent stem edge will acquire its tip state.

Check if retraction finished

Retraction end condition:



Retraction finished when tip node is within a certain distance of line between non-tip nodes of adjacent tip edges.

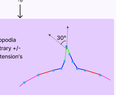
If retraction end condition met → Remove this filopodia by resetting filopodia state variable of all edges it is comprised of

Attempt to extend

Cell has actin available? — Actin available → Edge VEGF.VEGFR > extension threshold? — VEGFR activation level meets extension requirement → Edge is tip to not in a filopodia?

Tip:

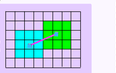
Extension angle after filopodia initialisation will be an arbitrary +/- 15°variation on previous extension's angle.

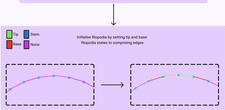


Not in filopodia:

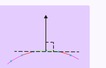
Decide which node will become tip node.

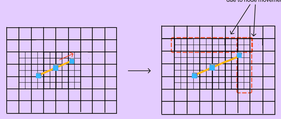
Tip node will be node which has the highest concentration of VEGF in its Moore neighbourhood of grid sites





Initially, tip extends perpendicular to line between two base connections.



Move tip node to new position according to extension angle + distance and decrement cell actin

If an edge's node extends into an area of the environment which does not contain grid sites at the edge's coarseness level, then the set of refined grid site locations that the edge is currently within is expanded to cover that new area.