



EvoLife v0.6

Uni 102



Materials

Materials

- Rocks
 - Can be static
 - Generated at simulation start
 - Added by user
- Gas bubbles
 - Inspired by deep sea gas vents
 - Spawned by spawner fields
 - Contained energy is set by spawner field
- Dead cells
 - When cells die they became edible food
 - Contained energy is set by dying cell
- Biomaterials (0..25)
 - Produced by cells
 - Contained energy is set at simulation start

Material interactions

Cells can

- Stick to
 - Rocks
 - Biomaterials
 - Other cells
- Produce
 - Biomaterials
- Break down
 - Gas
 - Dead cells
 - Biomaterials
- Siphon
 - Other cells
 - Dead cells

Biomaterials can

- Stick to
 - Rocks
 - Gas
 - Living cells
 - Dead cells
 - Biomaterials



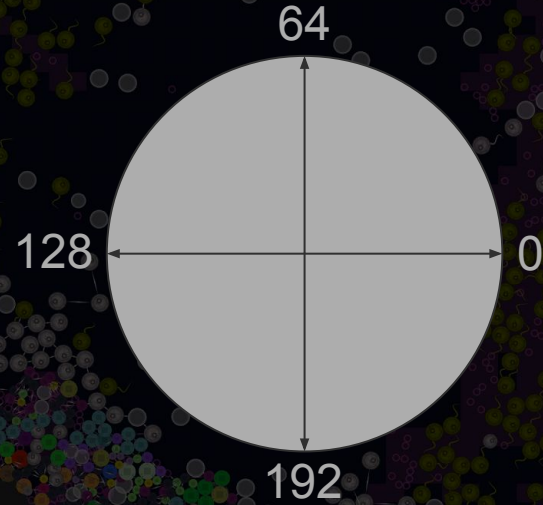
Cell internals

Cell internals

- Age
- Energy / max energy (E)
- Hitpoints / max hitpoints (hp)
- 16 behavioral slots
 - Organelles
 - Behavioral logic
- Spawn cooldown (0..9 game ticks)
- Species
 - Surface proteins (15 bits)
 - Color
 - DNA sequence
 - DNA pieces

Behavioral slots

- 16 per cell
- Opcode (DNA piece)
 - Executes as DNA, gets copied to a slot
 - Math
 - Sequence
 - Organelle
- Can read other output registers as inputs
- Output register, defines slot
 - Default value
 - Valid bit: valid or invalid
 - Data: 0..255
 - Scalar or direction



Cell healing

- Every 100 game ticks
- If $E \geq \max E / 2$
- If $E \geq 1000$
- If $hp < \max hp$
- $Hp += 1$
- $E -= 500$

Cell death

- Happens if either
 - Age \geq max age
 - $E \leq 0$
 - $H_p \leq 0$
 - Programmed cell death
- Dead cell energy = $E + 250 * h_p$
- Connections initiated by the cell disconnect

Cell during one game tick

- Heals if age modulo 100 == 0
- Executes the current DNA piece
- Executes the 16 behavioral slots
 - Output register reads contain data from previous game tick
 - Result of this game tick can be read in the next one
- Checks E and hp, can become dead



DNA pieces

DNA pieces

- Nop: nothing
- Math: Arbitrary math in a behavioral slot
- Sequence: Periodic sequences and higher level logic in a behavioral slot
- Organelle: grow some capability in a behavioral slot
- Egg: new fresh cell
- Fission: copy cell
- Jump: jump in DNA execution
- End: programmed cell death
- Wait: for event or time

DNA piece: nop

- Time to execute: 50 game ticks
- Energy per game tick: 1
- Does nothing

DNA piece: math

- Occupies a behavioral slot
- If slot already contains the exact same opcode:
 - Time to execute: 20 game ticks
 - Energy per game tick: 1
- Else:
 - Time to execute: 50 game ticks
 - Energy per game tick: 2
- Energy cost in a behavioral slot: 0
- Operation: math operation performed
- Post processing
 - Wrap: modulo 256
 - Clamp: to 0..255
- Validity mode
 - Valid: updated every tick, results can be invalid
 - Enable: updated only if result will be valid
- Validity type
 - Table: truth table of valid bits
 - Compare: comparison of data values
- Input 1: register
- Input 2
 - Register
 - Constant

DNA piece: sequence

- Occupies a behavioral slot
 - If slot already contains the exact same opcode:
 - Time to execute: 20 game ticks
 - Energy per game tick: 1
 - Else:
 - Time to execute: 50 game ticks
 - Energy per game tick: 2
 - Energy cost in a behavioral slot: 0
- Prio list: output is the highest priority input that is valid
 - Const: constant value
 - Attribute: output calculated from cell stats
 - Rand: random number in a range, updated on valid input
 - Beat: valid if timer is reached maximum, timer restarted automatically and on valid input
 - Hold: stays valid for a time after input become invalid
 - Delay: stays invalid for a time after input became valid
 - Sine: wave, input sets phase
 - Square: wave, input sets phase
 - Triangle: wave, input sets output
 - Sawtooth: wave, input sets output

DNA piece: organelle

- Occupies a behavioral slot
- If slot already contains the exact same opcode:
 - Time to execute: 20 game ticks
 - Energy per game tick: 1
- Else:
 - Time to execute: 50+ game ticks
 - Energy per game tick: 2+
 - Increased by the organelle properties (td, Ed)
- Energy cost in a behavioral slot: 0+
 - Increased by organelle properties (Eu), different organelle types require energy at different times
- 65536 organelles with different parameters
- Can take inputs
- Can provide outputs
- Organelles will be detailed later

DNA piece: egg

- Time to execute: 50 game ticks
- Energy per game tick: 2
- Cell division
- Direction
 - Absolute
 - Behavioral slot output register
- Connection
 - Use an existing connection type organelle to connect with the new cell
- Address
 - New cell will start DNA execution from here
- Creates a fresh cell
- Can mutate, evolution will be discussed later
- Age of new cell is reset to 0
- All behavioral slots are empty
- If the original cell has only 1 hp or 1 E then nothing happens
- New cell
 - $Hp = \min(\text{original cell hp} / 2, 10)$
 - $E = \min(\text{original cell E} / 2, 5000)$
- Original cell
 - $Hp = \text{original cell hp} - \text{new cell hp}$
 - $E = \text{original cell E} - \text{new cell E}$

DNA piece: fission

- Time to execute: 50 game ticks
- Energy per game tick: 2
- Cell division
- Direction
 - Absolute
 - Behavioral slot output register
- Connection
 - Use an existing connection type organelle to connect with the new cell
- Address
 - New cell will start DNA execution from here
- Creates a copy of the cell
- Age of new cell is same as the original
- All behavioral slots are copy of the original
 - Connection types are unconnected
- If the original cell has only 1 hp or 1 E then nothing happens
- New cell
 - $Hp = \text{original cell hp} / 2$
 - $E = \text{original cell E} / 2$
- Original cell
 - $Hp = \text{original cell hp} / 2$
 - $E = \text{original cell E} / 2$

DNA piece: jump

- Time to execute: 50 game ticks
- Energy per game tick: 2
- DNA piece located at the selected address will be executed next
- Can be conditional
 - Checks selected output register valid bit

DNA piece: end

- Time to execute: 50 game ticks
- Energy per game tick: 2
- Programmed cell death
- The cell dies

DNA piece: wait

- Time to execute: settable game ticks
- Energy per game tick: 1
- Waits for a settable time
- And / or
- A register valid bit to have a set value



Organelles

Organelles

- Id: 0..65535
- Type
- Type parameters
 - Unsigned integers 0..255 (increment is 1)
 - Unsigned fixed point 0..12.75 (increment is 0.05)
 - Signed fixed point -6.40..6.35 (increment is 0.05)
- Additional time to develop (td)
- Additional energy to develop (Ed)
- Energy to use (Eu)
 - Subtracted from cell energy at type specific events

Organelle type: spikes

- Spike to poke every touching cell causing damage
- Parameters
 - Atk (0..255, generating 0..63): Damage the cell cause to all other cells touching it
 - $\text{Damage} = \text{atk} - \text{def}$
- Inputs
 - None
- Energy used
 - Never
- Output register
 - No change

Organelle type: size

- Organelle to change cell size and mass
- Parameters
 - Delta radius (-6.40..6.35): Changes the radius of the cell. Default is 2.0, minimum is 1.0, maximum is 2.475
 - Delta mass (-6.40..6.35): Changes the mass of the cell. Default is 1.0, minimum is 0.1, maximum is 20.0
 - Max hp = 10 * mass
 - Max E = 5000 * mass
- Inputs
 - None
- Energy used
 - Never
- Output register
 - No change

Organelle type: cell wall

- Thick cell wall to protect from damage
- Parameters
 - Def (0..255): Reduces incoming damage
 - $\text{Damage} = \text{atk} - \text{def}$
- Inputs
 - None
- Energy used
 - Never
- Output register
 - No change

Organelle type: cell color

- The cell grows pigments, recoloring itself
- Parameters
 - Red (0..255): component of the cell color
 - Green (0..255): component of the cell color
 - Blue (0..255): component of the cell color
- Inputs
 - None
- Energy used
 - Never
- Output register
 - No change

Organelle type: active spikes

- Retractable spikes
- Parameters
 - Atk (0..255, generating 0..63): Damage the cell cause to all other cells touching it
 - $\text{Damage} = \text{atk} - \text{def}$
- Inputs
 - On/off source
- Energy used
 - When it is engaged/disengaged
- Output register
 - Valid 255 when engaged
 - Invalid 0 when disengaged

Organelle type: flex

- Organelle to change cell size
- Parameters
 - Maximum delta radius (-6.40..6.35): Changes the radius of the cell. Default is 2.0, minimum is 1.0, maximum is 2.475
- Inputs
 - Flex amount, $\text{delta radius} = (\text{input} / 255) * \text{maximum delta radius}$
- Energy used
 - If input is valid
- Output register
 - Last valid input

Organelle type: heal

- Additional healing
- Parameters
 - Cooldown (0..255, generates so cooldown * Eu > 500)
 - When cooldown reaches 0 and hp < max hp then hp is increased by 1 and cooldown resets
- Inputs
 - On/off source, enables cooldown and healing
- Energy used
 - If organelle is enabled and cooldown is not 0
- Output register
 - Valid 255 if hp += 1 happened
 - Invalid 0 else

Organelle type: flagellum

- Propels the cell forward, remembers last valid direction and force pair
- Parameters
 - Maximum force (0..12.75)
- Inputs
 - Direction source
 - Force source
 - Organelle direction and force is updated if both inputs are valid
 - Flagellum propel forward force = maximum force * (force input / 256)
- Energy used
 - If $Eu == 0$ then 0
 - If force == 0, then 0
 - Else $\max(1, Eu * (\text{force input} / 256))$
- Output register
 - Current direction

Organelle type: produce smell

- Parameters
 - Smell id (0..7): which smell to produce
 - Maximum amount (0..12.75): how much to produce
- Inputs
 - Emit amount
 - Smell amount emitted = maximum amount * (input / 256)
- Energy used
 - If Eu == 0 then 0
 - If emit == 0, then 0
 - Else max(1, Eu * (emit input / 256))
- Output register
 - Emit amount

Organelle type: produce biomaterial

- Parameters
 - Biomaterial id (0..25): which biomaterial to produce
 - Cooldown (0..255, generates so cooldown * Eu > biomaterial E)
- Inputs
 - Direction source
 - If cooldown is 0 then a biomaterial is emitted in the direction, resetting the cooldown
- Energy used
 - If cooldown is not 0
- Output register
 - Valid 255 if ready to emit, cooldown is 0
 - Invalid 0 else

Organelle type: break down

- Break down a dead cell, a gas bubble or a biomaterial
 - Cell will only break down material if $\text{max energy} - \text{current energy} > 100$
 - Takes 100 energy out of the material or less if less is remaining
 - If the broken down particle has remaining 0 energy it despawns
- Parameters
 - Material id: which material to break down
 - Cooldown (0..255)
 - For gas bubbles and dead cells it generates 0..31
 - For biomaterials it generates 0..255
- Inputs
 - On/off source
- Energy used
 - If cooldown is not 0
- Output register
 - Valid 255 if ready to break, cooldown is 0
 - Invalid 0 else

Organelle type: siphon

- Siphon energy out of a dead or alive cell
 - Cell will only siphon if $\text{max energy} - \text{current energy} > 25$
 - Takes 50 energy out of a dead cell and receives 25
 - Takes 50 - defense energy out of a living cell and receives up to 25
- Parameters
 - Surface protein threshold (0..15): do not siphon living cell if more surface protein bits match than defined here
 - Cooldown (0..255)
- Inputs
 - On/off source
- Energy used
 - If cooldown is not 0
- Output register
 - Valid 255 if ready to siphon, cooldown is 0
 - Invalid 0 else

Organelle type: sense direction

- Parameters
 - What to sense
 - Fluid: direction of the flow of the fluid surrounding the cell
 - Cell: direction of the movement of the cell
- Inputs
 - On/off source
- Energy used
 - If enabled
- Output register
 - Valid direction if enabled
 - Invalid 0 else

Organelle type: sense smell

- Parameters
 - Smell id (0..7): which smell to sense
 - Mul (0..255): output scaling
 - Div (0..255): output scaling
 - $\text{Output} = \text{sensed smell value} * \text{mul} / \text{div}$
- Inputs
 - On/off source
- Energy used
 - If enabled
- Output register
 - Valid amount if enabled
 - Invalid 0 else

Organelle type: sense touch

- Sense the direction of the last touching object
- Parameters
 - None
- Inputs
 - On/off source
- Energy used
 - If enabled
- Output register
 - Valid direction if enabled
 - Invalid 0 else

Organelle type: sense damage

- Sense the direction of the last spike poke or siphon event
- Parameters
 - None
- Inputs
 - On/off source
- Energy used
 - If enabled
- Output register
 - Valid direction if enabled
 - Invalid 0 else

Organelle type: sense color

- Sense the color of the first object in a straight line (raycast)
- Parameters
 - Range (0..12.75): length of the detection from the center of the cell
 - Color: What color component to sense
 - Red, green, blue
 - Gray: average of the 3
- Inputs
 - Direction source, which direction the cell looks
- Energy used
 - If enabled
- Output register
 - Valid color component if the ray hit something
 - Invalid 0 if not enabled or ray hit nothing

Organelle type: sense cells

- Sense the number of alive cells in range, even through object
- Parameters
 - Range (0..15): range of detection will be $\sim \text{range} * 5.0$
- Inputs
 - On/off source
- Energy used
 - If enabled
- Output register
 - Valid number of alive cells in range, including the cell using this organelle
 - Invalid 0 if not enabled

Connection organelles

- Common parameters
 - Length (0..12.75): length of the connection
 - Strength (0..12.75): strength of the connection
 - Rod / rope joint
 - What it connects to
 - Rock
 - Surface protein count: at least this number of surface protein bits match
 - Other alive cells
 - Biomaterials

Connection organelle type: stick

- The cell sticks to something, can unstuck
- Special parameters
 - None
- Inputs
 - On/off source
- Energy used
 - Never
- Output register
 - Valid connection direction if connected
 - Invalid 0 if not

Connection organelle type: muscle

- Special parameters
 - Length act as maximum length
- Inputs
 - Length source
 - $\text{Connection length} = \text{maximum length} * (\text{input} / 256)$
- Energy used
 - If input is valid
- Output register
 - Last valid length input if connected
 - Invalid 0 if not

Connection organelle type: transfer

- Connect only to other alive cells
- Special parameters
 - Transfer hp (0..3): transfer hp to the connected cell
 - Have to have more than 1 remaining
 - Transfer energy (0..15): transfer energy to the connected cell
 - Have to have more than 100 remaining
- Inputs
 - Transfer data source
- Energy used
 - Never
- Other cell
 - Transfer data destination
 - Transfer source data to the connected cell destination output register
 - This overwrites connected cell behavioral slot output
- Output register
 - Transferred data if connected
 - Invalid 0 if not



Evolution

Species creation

- User created species
- Spawned species
 - Randomised DNA
 - Randomised parameters
 - Adds energy to the simulation
- Mutating existing species
 - 100% chance
 - One DNA add / del / change
 - Color rgb value step by 1
 - Can happen
 - More than one DNA change
 - Change in default output registers
 - Change in surface protein bits
 - Change in mutation chance

Species lifecycle

- Species is spawned
 - 5 cells with 10 hp and 5000 energy
 - Always spawned on top of a gas spawner field
- If pop count reaches 50, a mutated version is generated
- If 5 mutated egg is spawned a new mutation is generated
- If population count hits 0
 - Sum spawned species cell count is less than 100000 the species is deleted
 - If more it is kept as a relic of a once successful species
 - Species won't be deleted if it is in the GUI list of species
- DNA tree visualization coming up in a future release



Overcrowding

Overcrowding

- Rocks merge
 - Mass adds up
 - Become static
- Gas bubbles, dead cells, biomaterials merge
 - Each with its own type
 - Contained energy adds up
 - Up to 10000
- Cells take damage
 - Only if $\text{def} < 20$



Congratulations!