M9. Neutrophil module

M9.1 Slot, status and schedule

Summary. Neutrophils are a small agent type that moves by its own chemotactic rule, kills or takes up extracellular bacteria, secretes TNF and may stand in caseum. No neutrophil exists in the default configuration, but under the probabilistic option of M2 the neutrophil recruitment predicate still runs and consumes one draw at every open source slot. Neutrophils can be created by Init-section entries, random placement at initialisation, recruitment and injection (M8.9).

What the slot controls

M9 specifies the neutrophil agent type: the routes by which a neutrophil comes to exist, where it may stand, its movement and recruitment, what it does with bacteria, its secretion, how its death enters the killing count, the neutrophil-only statistics and the neutrophil parameters. Mechanisms that neutrophils share with other agents are specified by their owners and linked from here. M9 states only how neutrophils differ.

Injection, the one-time placement of neutrophils inside the granuloma at a chosen timestep, is an option of M8. M8.9 owns its trigger, candidate set, admission rule and parameters. M9 owns everything an injected neutrophil does afterwards.

Options and how they are selected

No single switch selects the module. Neutrophils are held absent by the values of four independent creation paths, each of which remains executable. A study enabling neutrophils must open each path it intends to use and close the others.

Path	Controlled by	Default setting	See
Init-section `Neutrophil` entries	Presence of an entry in the parameter file	No entry	M9.3
Random placement at initialisation	`Core_Neutrophil_initDensity`, or `Core_Neutrophil_initNumber` if the density is not read	Density 0.0, giving a count of 0	M9.3
Recruitment at vascular sources	The depletion window, `Recruit_Prob_Neutrophil_depletionTimestep` and `Recruit_Prob_Neutrophil_depletionEndTimestep`	0 and -1, so recruitment is refused at every timestep	M9.5
Injection	A positive `Core_Neutrophil_injectionTimestep`	0, so injection never fires	M8.9

Init-section `Neutrophil` entries are honoured in every configuration. Unlike fibroblast entries when M4 is not selected, they are not ignored with a warning.

Recruitment is the only path held closed by a time window rather than by a zero count. Both recruitment probability bounds, `Recruit_Prob_Neutrophil_minRecProb` and `Recruit_Prob_Neutrophil_maxRecProb`, are also 0. Recruitment would therefore create no neutrophil even with the window open, because the recruitment probability is identically zero (M9.5).

What runs although no neutrophil exists

Every timestep of the default configuration runs this neutrophil code:

- **The neutrophil recruitment predicate**, under the probabilistic option of M2, at every open slot of every source that is neither full nor caseous (M9.5). Each evaluation computes the full recruitment signal, including both fatal consistency checks; evaluates the crowding rule, consuming one uniform draw; and increments up to four neutrophil recruitment statistics, three of them generally non-zero (M9.9). The probability draw that would decide recruitment is not reached, because the depletion test returns first.
- **The neutrophil movement, secretion, next-state and state-commitment passes**, over an empty list.
- **The injection test** at the end of the timestep (M8.9).

The crowding draw is part of the random number stream of the default configuration. A reimplementation that omitted neutrophil code would diverge from the first recruitment block onward, although no neutrophil is ever created.

Place in the cellular timestep

Block	Neutrophil activity	Order relative to other types	See
Split half-step	Secretion of TNF; under M4, binding and secretion of TGF-β	After macrophages, T cells and fibroblasts; before caseum secretion	M9.7
Movement	`Core_Neutrophil_movesPerTimestep` attempts per neutrophil	After macrophages and dendritic cells; before Tgam, Tcyt, Treg and fibroblasts	M9.4
Recruitment	Neutrophil predicate at every open slot	Second branch of the cascade (S8.11)	M9.5
Next-state computation	Senescence, lifespan reduction, kill or uptake, intracellular growth and decay	Last: after macrophages, dendritic cells, the three T cell types and fibroblasts	M9.2, M9.6, M9.8
State commitment	Pending state committed; dead neutrophils removed	Last	M9.2

The global type orders belong to spine 3. The movement and next-state orders differ: a neutrophil moves before any T cell, but computes its next state after every other type. Cross-agent effects are written at once (spine 2.3). A neutrophil computing its next state therefore sees every death written earlier in the pass, caseation cascades included.

M9.2 The neutrophil agent

Size class and occupancy

A neutrophil is a small agent occupying one of the two slots of its compartment (spine 2.3). A compartment may therefore hold two neutrophils, a neutrophil beside a T cell, or a neutrophil beside one large agent. The size class is a compile-time constant; no parameter changes it.

A neutrophil counts toward the occupant count n(p). It therefore enters every quantity built on that count: the occupied predicates, the cell density that gates vascular delivery and defines the threshold granuloma, connected-region eligibility, the neutrophil extent and the per-type caseous-surroundings statistic (S9).

`Core_Neutrophil_probMoveToBigOrNeut` changes how a neutrophil treats compartments it enters (M9.3). It does not change how other agents see a neutrophil, which is always as a small agent.

Construction

A neutrophil is constructed with a birth time, a position and a state. Its death time is its birth time plus `Core_Neutrophil_maxAge`.

Construction passes the neutrophil's own receptor parameters to the common agent constructor; S6.3 owns the draw mechanism.

Quantity	Mean	Half-width
Surface TNFR1	`Molecular_TNF_Neutrophil_meanTNFR1Neut`	`Molecular_TNF_Neutrophil_stdTNFR1Neut`
Surface TNFR2	`Molecular_TNF_Neutrophil_meanTNFR2Neut`	`Molecular_TNF_Neutrophil_stdTNFR2Neut`
Surface IL-10 receptor	`Molecular_IL10_Neutrophil_meanIL10RNeut`	`Molecular_IL10_Neutrophil_stdIL10RNeut`

The membrane TNF synthesis rate is set to 0, and the TACE release rate to `Molecular_TNF_Neutrophil_kTaceNeut`. The common constructor also makes its NF-κB draw when `NFkB_meanNFkB` is positive (S6.3).

The receptor counts are stored but have no dynamical effect in any configuration (M9.7). The draws nonetheless consume the random number stream, so the receptor parameters must be supplied with valid values.

Each neutrophil gets an intracellular bacterial container, created empty. Construction therefore creates no bacterium and makes no growth-rate draw, whatever the randomisation setting of M8. The randomisation flag and the intracellular growth-rate bounds are passed to the container but have no effect: bacteria taken up later keep the growth rates they carry (M9.6).

Birth time and initial state by route

Write M for `Core_Neutrophil_maxAge` and t for the current timestep. Every route creates the neutrophil active.

Route	Birth time	Resulting death time
Init-section entry	Uniform on $\{-M, \dots, -2\}$	$\{0, \dots, M-2\}$

Route	Birth time	Resulting death time
Random placement at initialisation	Uniform on $\{-M, \dots, -2\}$	$\{0, \dots, M-2\}$
Recruitment	t minus a value uniform on $\{1, \dots, M-1\}$	$t + \{1, \dots, M-1\}$
Injection	t	t + M

Each draw is defined only for $M \geq 2$ (M9.10.1).

A recruited neutrophil is added during recruitment, which precedes next-state computation, so it computes a next state on its arrival timestep. An injected neutrophil is added after next-state computation, so it first acts on the next timestep.

States

A neutrophil has two states, active and dead. It is created active, and its only transition is to dead.

There is no infected state. A neutrophil holding live intracellular bacteria remains active. Rules that distinguish infected neutrophils test whether the container holds at least one live bacterium (M9.5, M9.9). The hybrid update of spine 2.3 applies as for other agents.

Next-state computation

```
next-state computation for neutrophil a at timestep t:
    if a is dead or dead-next:
        set pending state dead; return
    if t >= death time of a:                                  // senescence
        if position not inside the previous timestep's region
                and live intracellular count > 0:
            increment the outside-granuloma death and release statistics
        release all intracellular bacteria, live and dead, into the
            extracellular container of a's compartment
        u <- U(0,1)                                           // always drawn
        if u < Core_Neutrophil_probabilityThatDeathContributesToCaseation:
            increment the killing count at a's compartment, naming a
                (kills a at once; may trigger the caseation cascade)
        set pending state dead
    else:                                                     // state is active
        run the active handler of a
    if live intracellular count > 0:
        increment the neutrophils-holding-bacteria statistics
    if live intracellular count > Core_Neutrophil_maxNrIntMtbAllowed:
        increment the cap-exceeded statistic
```

The active handler runs, in this order:

1. the lifespan reduction in damaged tissue (M9.8);

2. kill or uptake on the extracellular bacteria of the neutrophil's own compartment (M9.6);
3. growth of the intracellular population (M9.6);
4. decay of the intracellular dead list (M9.6).

The handler writes no pending state that could override a death, so an active neutrophil stays active unless killed.

The outside-granuloma test uses the circular approximation to the connected region built at the end of the previous timestep (S9.8).

The neutrophil's cumulative intracellular-holding time is never incremented, so its contribution to the intracellular-time statistic is always zero.

Death and removal

A neutrophil dies by senescence or by the caseation cascade only.

- **Senescence**, at its death time, which damaged tissue may have brought forward (M9.8). Its intracellular bacteria are released into its own compartment. Its death contributes to the killing count with probability `Core_Neutrophil_probabilityThatDeathContributesToCaseation`.
- **The caseation cascade** (S5.3). A neutrophil killed by the cascade does not release its intracellular bacteria, and its death contributes to the killing count deterministically (M9.8).

A dead neutrophil is removed in the state-commitment pass. Removal increments the death counter of its compartment, frees its slot, and destroys the container together with any bacteria still in it.

No TNF-induced apoptosis test is made for a neutrophil, and no neutrophil proliferation route exists.

M9.3 Placement and admission

The admission predicate and the crowding rule decide where a neutrophil may stand. They differ in what they test and in whether they draw. Each placement route uses one or both.

The admission predicate

A neutrophil is admitted to compartment p if and only if n(p) < N_slots, with N_slots = 2. The predicate makes no caseation test, no large-agent test and no draw. It refuses only a full compartment.

Agent addition applies this predicate to neutrophils in place of the caseation test and type predicate it applies to every other type. Every route that adds a neutrophil therefore admits caseous compartments, unless the route itself tests caseation first. Of the placement routes, only recruitment does.

The crowding rule

Movement, random placement at initialisation and recruitment use the crowding rule. It returns a movement value of 1 or 0 and a blocked flag.

```
crowding rule at compartment p, returning (moveValue, blocked):
    u <- U(0,1)                                    // always drawn, first
    b <- some occupant of p is a large agent
    q <- some occupant of p is a neutrophil
    if n(p) < N_slots:
        if (not b and not q)
                or Core_Neutrophil_probMoveToBigOrNeut == 1
                or u < Core_Neutrophil_probMoveToBigOrNeut:
            return (1, false)
        else:
            return (0, true)
    else:
        return (0, true)
```

The rule makes no caseation test. Its draw is consumed whatever the outcome, including at a full compartment and when the parameter equals 1. In the default setting the parameter is 1.0, so the outcome never depends on the draw, but the draw is still made.

At a parameter value of 0 the neutrophil behaves as a large agent that also refuses other neutrophils; at 1, as a small agent.

Placement routes compared

Route	Test applied	Caseous compartment	Draws for admission	Failure
Init-section entry	Admission predicate	Admitted	None	Fatal
Random placement at initialisation	Crowding rule, then admission predicate	Admitted	One crowding draw per attempt, after the position draws	Position redrawn
Recruitment	Caseation skip by the source loop, crowding rule, then admission predicate	Refused	One crowding draw per open slot	No recruitment at that slot
Injection	Admission predicate, inside the granuloma approximation	Admitted	None	Requires fewer neutrophils than candidates (M8.9)
Movement	Crowding rule, then admission predicate	Admitted	One crowding draw per attempt	Neutrophil stays

Where a route applies the crowding rule first, the admission predicate always succeeds, since the rule has already required a free slot. An injected neutrophil may therefore stand where the crowding rule would have refused a randomly placed one, and in caseum.

Init-section entries

Init-section entries are processed in the element order of spine 5.4. Each `Neutrophil` entry takes its position from the entry, given as absolute coordinates, as an offset from the grid centre, or as the centre itself. It then draws a birth time (M9.2), constructs an active neutrophil, appends it to the neutrophil list and adds it to the grid. No module switch is tested.

Random placement at initialisation

Random neutrophil placement follows the other initial placement loops, in the order given by spine 5.4. The number placed is `Core_Neutrophil_initNumber`, computed from `Core_Neutrophil_initDensity` when the density is read (M9.10.3). Placement is by rejection sampling over the whole grid, with no border exclusion.

```
remaining <- Core_Neutrophil_initNumber
if remaining > free-compartment counter:
    fatal error
while remaining > 0:
    p <- uniformly random grid position
    (moveValue, blocked) <- crowding rule at p    // one draw
    if moveValue == 1:
        draw a birth time (M9.2)
        create an active neutrophil at p
        remaining <- remaining - 1
        decrement the free-compartment counter
```

The free-compartment counter starts at the number of grid compartments and is decremented once for each agent placed by the earlier random-placement loops. Init-section placements are not counted. The counter bounds the request by compartments, not slots, so it does not guarantee that enough admissible positions exist.

The loop has no exit other than placing the full count. If no admissible compartment remains, the loop repeats without end. This happens when every compartment is full, or, with `Core_Neutrophil_probMoveToBigOrNeut` = 0, when every compartment that is not full holds a large agent or a neutrophil. With the parameter positive, any compartment that is not full is eventually admitted.

With a count of zero the loop is not entered and no draw is made, as in the default setting.

Random placement precedes the parameter-file initialisation of compartment fields, so no compartment is caseous at placement.

M9.4 Movement

Schedule and opportunity

Each neutrophil, in list order, makes `Core_Neutrophil_movesPerTimestep` movement attempts per cellular timestep (spine 3). It applies no age-interval test and no state dependence, so it attempts movement on every timestep. A value of 0 means a neutrophil never moves.

Each move is committed at once. Neutrophils processed later, and the T cells that move after them, see the new occupancy.

```
movement of neutrophil a:
    repeat Core_Neutrophil_movesPerTimestep times:
        k <- destination routine for a             // one draw
        q <- compartment at Moore ordinal k from a.pos (periodic)
        (moveValue, blocked) <- crowding rule at q // one draw
        if blocked:
            increment the neutrophil blocked-move statistic
        if moveValue == 1:
            add a at q; remove a from a.pos; a.pos <- q
```

Secretion precedes movement (M9.7), so a neutrophil secretes only into its start-of-timestep compartment. Compartments it passes through receive nothing.

Moves into its own compartment

Ordinal 4, the neutrophil's own compartment, is a possible destination. The crowding rule sees the neutrophil itself as a neutrophil occupant.

- **Alone in its compartment**, the neutrophil finds a free slot. Entry succeeds outright when the parameter equals 1, and otherwise with probability `Core_Neutrophil_probMoveToBigOrNeut`. A successful entry changes its slot, not its position.
- **With a second occupant**, the compartment is full. The rule reports blocked and the neutrophil stays.

Either way the neutrophil ends where it began, but the blocked-move statistic may be incremented.

Caseous compartments

Neither test checks caseation, so a neutrophil may move into caseum and remain there. The killing-count term of the destination routine draws neutrophils toward damaged tissue, caseum included, where their lifespan is shortened (M9.8).

The destination routine

The neutrophil does not use the shared routine of S1.3. Its own routine weights three profiles over the periodic Moore neighbourhood of its position. For each ordinal $k = 0, \dots, 8$, with q_k the corresponding compartment:

- L_k is the live extracellular count at q_k;
- K_k is the killing count at q_k;
- C_k is 0 if C_CCL5(q_k) < `Core_minChemotaxis`, `Core_maxChemotaxis` if C_CCL5(q_k) >

`Core_maxChemotaxis`, and C_CCL5(q_k) otherwise.

Each profile is normalised over the neighbourhood by a guarded division that returns 0 when the total is not strictly positive:

$$\ell_k = \frac{L_k}{\sum_j L_j}, \qquad \kappa_k = \frac{K_k}{\sum_j K_j}, \qquad c_k = \frac{C_k}{\sum_j C_j}.$$

With w_L = `Core_Neutrophil_liveCount_MovWeight` and $w_K = w_C = \frac{1}{2}(1 - w_L)$, the combined weight is

$$g_k = w_L \ell_k + w_K \kappa_k + w_C c_k, \qquad G = \sum_k g_k.$$

If G > 0, one uniform draw r on [0, 1) is made, and the destination is the first ordinal whose cumulative probability is at least r. Otherwise one ordinal is drawn uniformly from the nine. Either way the routine makes exactly one draw.

While every weight is non-negative, r = 0 selects ordinal 0 whatever its weight. If `Core_Neutrophil_liveCount_MovWeight` exceeds 1, the first cumulative value can be negative and this no longer holds.

If rounding leaves the final cumulative value below r, the destination is the lower-left neighbour, ordinal 0's compartment.

Properties of the destination rule

The weights renormalise over the profiles present. Each normalised profile sums to 1 if any entry is positive and to 0 otherwise, so

$$G = w_L[\textstyle\sum L > 0] + w_K[\sum K > 0] + w_C[\sum C > 0].$$

An absent profile drops out, and the remaining weights are rescaled to sum to 1. With w_L = 1, killing count and CCL5 carry no weight. A neighbourhood without live extracellular bacteria then gives G = 0 and a uniform move, even where damage and CCL5 are present.

Each profile is normalised separately. The magnitudes never compete, so one bacterium in a compartment pulls as strongly as a hundred.

The CCL5 window clamps. It does not switch chemotaxis off, and it is evaluated at every compartment of the neighbourhood. This differs from S1.3, where a value outside the window at the agent's own compartment removes the chemokine from the weighting. The bounds are the shared chemotaxis parameters of S1.

No bonus factor and no sensitivity to CCL2 or CXCL9. CCL5 is a fixed multiple of CCL2 (S6.14), so the CCL5 profile has the spatial pattern of CCL2.

The weight is not range-checked. Above 1, w_K and w_C are negative, the cumulative distribution is not monotone, and the cumulative values do not form a probability distribution.

The blocked-move statistic

The neutrophil entry of `NrTimesWhereMoveBlockedByBigCell` is incremented on every blocked attempt: into full compartments, into the neutrophil's own full compartment, and refused by the crowding draw. It counts blocked attempts of every kind, not only blocks by a large agent. S9 owns the statistic.

M9.5 Recruitment

Scope and position

This section specifies neutrophil recruitment under the probabilistic option of M2. S8 owns the source loop, slot accounting and recruitment cascade (S8.2, S8.3, S8.11). M9 specifies the neutrophil predicate, its signal and the creation of a recruited neutrophil. Neutrophil recruitment under the alternative option of M2 is not described here (M2.9).

The source loop skips full and caseous sources before any predicate, and evaluates every predicate at every open slot (S8.3). The neutrophil predicate therefore runs, and consumes its crowding draw, at every open slot of every non-full, non-caseous source on every timestep. This holds whatever the depletion window and the recruitment probability.

The recruitment signal

The signal is built from four saturation factors. Each is a sum over the periodic Moore neighbourhood of the source, divided by the largest value it could take there.

Factor	Numerator	Denominator
s_ext	Live extracellular bacteria	$M_N \cdot$ `Mtb_growthExtMtbBound`
s_Mi	Live intracellular bacteria held by agents of the macrophage type	$M_N \cdot$ `Core_Mac_nrIntMtbBurstCInf`
s_CCL5	Windowed CCL5, summed	$M_N \cdot$ `Recruit_Prob_Neutrophil_maxThresholdRecChemokine_neut`
s_neut	Neutrophils holding at least one live intracellular bacterium	$M_N \cdot 2$

M_N is 9 in two dimensions and 27 under M5. The factor 2 is the number of small agents a compartment can hold. The macrophage type in s_Mi includes dendritic cells, were they present. At most one agent of that type stands in a compartment.

Windowed CCL5. Each compartment's CCL5 value is clamped before summing. A value below `Recruit_Prob_Neutrophil_minThresholdRecChemokine_neut` contributes 0. A value above `Recruit_Prob_Neutrophil_maxThresholdRecChemokine_neut` contributes that maximum. Any other value contributes itself. This window is distinct from the chemotaxis bounds of M9.4.

Divisions and clamps. Every division is guarded and returns 0 when the denominator is not strictly positive.

- s_ext is set to 1 when its numerator exceeds its denominator, so it lies in [0, 1].
- The other factors are not clamped. A fatal error is raised if the numerator of s_Mi or of s_neut exceeds its

denominator.

- The numerator of s_CCL5 cannot exceed its denominator, by construction of the window.

In the default configuration s_neut is identically 0, and the other factors are generally non-zero near the lesion.

The s_Mi check aborts the run when the live intracellular bacteria of macrophage-type agents, summed over the neighbourhood, exceed $M_N \cdot$ `Core_Mac_nrIntMtbBurstCInf`. It runs in every probabilistic-recruitment configuration, the default one included. A single macrophage can exceed the burst threshold at recruitment time, since an infected macrophage becoming chronically infected is not tested for bursting until the next timestep.

The weights and the signal

With ω_{ext} = `Recruit_Prob_Neutrophil_liveCount_RecWeight`, ω_{neut} = `Recruit_Prob_Neutrophil_neutCount_RecWeight` and $\omega_{\text{Mi}} = \omega_{\text{CCL5}} = \frac{1}{2}(1 - \omega_{\text{ext}} - \omega_{\text{neut}})$:

$$S = \frac{\omega_{\text{ext}} s_{\text{ext}} + \omega_{\text{Mi}} s_{\text{Mi}} + \omega_{\text{CCL5}} s_{\text{CCL5}} + \omega_{\text{neut}} s_{\text{neut}}}{\omega_{\text{ext}} + \omega_{\text{Mi}} + \omega_{\text{CCL5}} + \omega_{\text{neut}}}.$$

The equal split of the residual weight is fixed in the code. The denominator equals 1 by construction, so the normalisation has no effect beyond rounding. Nothing enforces $\omega_{\text{ext}} + \omega_{\text{neut}} \leq 1$; above 1 the derived weights are negative and S may leave [0, 1].

The probability and the eligibility test

The recruitment probability is $P_{\text{neut}} = S(p_{\max} - p_{\min}) + p_{\min}$, with p_min = `Recruit_Prob_Neutrophil_minRecProb` and p_max = `Recruit_Prob_Neutrophil_maxRecProb`. It is the only recruitment probability with a possible non-zero floor.

A source is eligible if and only if $C_{\text{CCL5}}(p) \geq r_5 \cdot$ `Recruit_Prob_Mac_thresholdRecChemokine` at the source, a non-strict test made with the significant-figure comparison of spine 2.1. Here r_5 is the CCL5-to-CCL2 ratio (S6.14).

- No neutrophil chemokine threshold exists. The test reuses the macrophage threshold and coincides with the CCL5 term of the macrophage eligibility test.
- Eligibility does not depend on the signal, and the signal is computed whether or not the source is eligible.
- In the default configuration eligibility is frequently true near the lesion. The nullity of recruitment comes from the depletion window and the probability bounds.

The neutrophil predicate

```
neutrophil predicate at source p, slot s, timestep t:
    (eligible, P) <- neutrophil signal at p          // no draw; may be fatal
    if eligible and s == 0:
        increment the neutrophil eligible-source statistic
    (moveValue, blocked) <- crowding rule at p       // one draw, always
    if blocked:
        increment the neutrophil blocked-source statistic
```

```
    if n(p) < N_slots and a large agent is at p:
        increment the neutrophil large-agent statistic
    if depletion window is in force at t:
        return false
    else if not caseated(p) and not blocked:
        return eligible and U(0,1) < P              // drawn only if eligible
    else:
        return false
```

The crowding rule is called for its blocked flag, and its movement value is discarded. Its draw precedes every test, so it is consumed inside the depletion window. The probability draw is reached only outside the window, at a non-blocked source, when the source is eligible. The large-agent statistic counts slots where a large agent shares the source with a free slot, whether or not the crowding rule blocked.

The depletion window

Neutrophil recruitment is refused at timestep t while

$$d_0 \geq 0 \ \wedge \ t \geq d_0 \ \wedge \ (d_1 < 0 \ \vee \ t \leq d_1)\,,$$

with d_0 = `Recruit_Prob_Neutrophil_depletionTimestep` and d_1 = `Recruit_Prob_Neutrophil_depletionEndTimestep`. The window is closed at both ends, and a negative end means it never closes.

- In the default setting $d_0 = 0$ and $d_1 = -1$. The window is in force at every timestep, so recruitment is refused for the whole run.
- Recruitment is unrestricted when $d_0 < 0$, whatever d_1, or when d_0 exceeds the final timestep.
- Both parameters are declared with a minimum of 0. The negative values that give them meaning are accepted at load, so the declared minimum is not enforced (spine 7.0, "Validity ranges").
- A load-time check rejects, as a fatal error, a non-negative end at or before the start: $d_1 \geq 0$ with $d_1 \leq d_0$.

Selection and creation

If the predicate returns true, the neutrophil competes in the S8.11 cascade as one category, beside the macrophage and the T cell categories.

A selected neutrophil is created as follows:

1. the source's total and neutrophil recruitment counters are incremented;
2. a birth time $t - \{1, \ldots, M-1\}$ is drawn (M9.2);
3. an active neutrophil is constructed at the source and added to the grid and the neutrophil list;
4. the cumulative, recruited and recruited-this-timestep neutrophil counts are incremented.

The new neutrophil occupies a slot and is seen by later slots and sources (S8.3). It computes a next state on the same timestep (M9.2).

Statistics written by recruitment

Each is written per open-slot evaluation unless stated otherwise; M9.9 gives definitions.

- **Extracellular exceedance count**: whenever s_ext was clamped.
- **Eligible-source count**: once per source, at slot 0, when the source is eligible.
- **Blocked-source count**: whenever the crowding rule reported blocked.
- **Large-agent count**: whenever a large agent shares the source with a free slot.
- **Active-source count**: whenever the predicate returned true.
- **Recruitment counts**: whenever a neutrophil is created.
- **Recruitment rate**: recomputed every recruitment block as the cumulative count divided by the current timestep.

The exceedance, eligible-source and large-agent counts are generally non-zero in the default configuration, although no neutrophil is recruited. The blocked-source count is zero in practice there, with `Core_Neutrophil_probMoveToBigOrNeut` = 1, because the crowding rule then blocks only at full sources and the source loop never evaluates one.

M9.6 Bacteria and interactions with other agents

Order of the active handler

The active handler's order is given in M9.2. Because neutrophils compute their next states after every other type, the extracellular container a neutrophil reads has already been acted on by any macrophage in the same compartment on that timestep. It has not yet been grown, reclassified or decayed by the extracellular pass, which follows the agent pass (S4).

Kill or uptake

The rule mirrors the extracellular rule of the resting macrophage (S2), with neutrophil parameters and an uptake cap. Write U_n for `Core_Neutrophil_nrExtMtbUptakeNeut`, p_n for `Core_Neutrophil_probKillExtMtbNeut`, and C_n for `Core_Neutrophil_maxNrIntMtbAllowed`.

```
kill or uptake by neutrophil a:
    E <- extracellular container at a.pos
    if E.liveCount == 0:
        return                                          // no draw
    if E.biomassSum < U_n:
        kill all live bacteria in E                     // no draw
        add the number killed to the neutrophil kill statistic
        pending state <- current state
    else if U(0,1) < p_n:
        kill min(U_n, E.liveCount) live bacteria of E  // shuffles only if
                                                        // fewer than all
        add the number killed to the neutrophil kill statistic
        pending state <- current state
    else:
        m <- min(U_n, E.liveCount)
        h <- live intracellular count of a
        if h + m > C_n:
            m <- max(min(C_n - h, m), 0)
        transfer m live bacteria from E to a, chosen at random
```

```
                                                    // shuffles even if m == 0
```

The biomass test. The first branch compares the biomass sum, not the count, with U_n. A compartment whose bacteria together carry less biomass than U_n is cleared outright. With U_n = 1, this means a single bacterium below unit biomass.

The cap. C_n limits uptake only, not growth, so the intracellular count can exceed C_n. Once it does, m is 0 and nothing more is taken up.

What uptake changes. Uptake does not change the neutrophil's state. It writes nothing the macrophage rule writes: no exposure memory, no uptake-in-caseum statistic and no cumulative infected count. Transferred bacteria keep their growth rates, including any stored rate under M8 randomisation.

Caseum. The rule makes no caseation test, so a neutrophil in caseum can kill or take up non-replicating bacteria there. It is the only agent that does so from within a caseous compartment. Activated macrophages kill in caseous neighbours but cannot stand in caseum (S2).

Intracellular growth and decay

The intracellular population grows by the S4 intracellular rule, with the M8 post-adaptive factor and randomisation flag, read from the simulation on every call. Bacteria taken up on a timestep grow on that timestep, because the extracellular pass comes later. Daughters and the dead list follow S4.

A neutrophil applies no intracellular killing, bursting threshold, chronic-infection threshold or drug killing. Its intracellular bacteria grow without limit until it dies. Starvation cannot occur while `Mtb_Agents_deathThreshold` does not exceed both `Mtb_Agents_initialBiomass` and $\min(\phi, 1-\phi)\cdot$ `Mtb_Agents_divisionThreshold`, with ϕ = `Mtb_Agents_divisionFraction`, which the default values satisfy (S4).

Fate of intracellular bacteria on death

Death route	Live and dead intracellular bacteria	Counted in bacterial death statistics
Senescence	Released into the extracellular container of the neutrophil's own compartment, with no dispersal, as macrophage senescence does (S2); dead bacteria keep their decay clocks	Not applicable
Caseation cascade	Destroyed with the neutrophil at removal	No; the caseum-killing statistic is incremented by the cascade for macrophages only (S5, S9)

Interactions with other agents

Rule	Effect on or of neutrophils
Treg contact suppression (S3.10)	Each active cognate Treg tests every agent in its periodic Moore neighbourhood that is not dead, not dead-next and not a non-cognate T cell. The neighbourhood includes the Treg's own compartment, so co-resident agents are tested, neutrophils included. Each test consumes one draw; the suppression has no effect on a neutrophil.
Fas-mediated and cytotoxic killing (S3)	Neutrophils are never targets; target selection enumerates macrophages only.
IFN-γ action, interaction counting, exhaustion (S3, M8)	Read macrophages only.
Macrophage exposure memory (S2.10)	A neighbouring neutrophil holding bacteria confers no exposure. Bacteria it releases on senescence confer exposure through the extracellular container.
NF-κB stimulation by neighbourhood bacteria (S2)	Reads extracellular bacteria only.
Macrophage movement (S1)	A compartment holding a lone neutrophil admits a macrophage.
T cell movement (S1)	The T cell's own movement routine enters an occupied compartment only if the occupant is a macrophage or a T cell, each with its own probability. A compartment holding a lone neutrophil is refused without a draw. The grid predicate used for T cell addition (not caseous, n(p) < 2) does not refuse it.
T cell recruitment (S8)	At a source holding a lone neutrophil, T cell recruitment into the free slot proceeds with probability `Core_Tcell_probMoveToTcell`, as for a lone T cell.
Macrophage recruitment (S8)	A source holding a lone neutrophil admits a macrophage.
T cell daughter placement (S3)	Placement tests free slots alone, with no type test, so a T cell daughter may be placed beside a lone neutrophil.
Oxygen consumption (S7.6)	Neutrophils consume no oxygen.

Drugs

Under M6 the neutrophil holds no intracellular drug concentration, applies no drug killing to its intracellular bacteria, and updates no efflux state for them. Neutrophil-held bacteria are also excluded from the random assignment of resistance at the start of treatment, which enumerates extracellular and macrophage-held bacteria only. M6.18 states the same.

M9.7 Secretion and cytokine effects

Schedule

Neutrophil secretion is called once in each split half-step, with argument $\Delta t_d/2$ (spine 3). It runs at the start-of-timestep position, since movement follows the diffusion block. A recruited or injected neutrophil first secretes on the timestep after its arrival. The M8 depletion flags and the M3 and M4 selections are passed through; secretion does not depend on state.

What is secreted

Species	Rate per half-step	Applies when
TNF	$0.1 \cdot$ `Core_Neutrophil_dTNF` $\cdot (\Delta t_d/2)$, times the logistic IL-10 suppression factor of the shared TNF helper (S6)	Coarse-grained TNF (M3); TNF not fully depleted (M8)
IL-10	0	Always
CCL2, CCL5, CXCL9	None; no chemokine helper is called	Never
Latent TGF-β	`Core_Neutrophil_dTGFBInactive` $\cdot (\Delta t_d/2)$	Either active option of M4; TGF-β not depleted (M8)

TNF. The state factor 0.1 is fixed in the neutrophil class. Fractional and full TNF depletion apply as specified in M8. TGF-β does not modulate neutrophil TNF secretion under any M4 option. The same call stores $0.1 \cdot$ `Molecular_TNF_Neutrophil_kSynthNeut` and $0.1 \cdot$ `Molecular_TNF_Neutrophil_kRNANeut` (0 under full depletion). These would feed M3 receptor equations, which neutrophils never solve, so they have no effect.

Under fine-grained TNF (M3) the shared helper writes no TNF to the field, and neutrophils solve no receptor equations. Under that option a neutrophil therefore secretes no TNF.

IL-10. The IL-10 helper is called with a state factor of 0. The field increment and the stored rate are therefore 0 under every depletion and receptor option, and `Core_Neutrophil_dIL10` and `Molecular_IL10_Neutrophil_IkSynthNeut` have no effect in any configuration.

TGF-β. Under either active M4 option the neutrophil first binds active TGF-β through the shared M4 routine (M4.3). It binds `Core_Neutrophil_tgfbBindingRate` $\cdot (\Delta t_d/2)$, capped at the local active TGF-β, into a cumulative bound quantity that no neutrophil rule reads. TGF-β depletion suppresses latent secretion, but not binding.

No uptake of TNF or IL-10

A neutrophil removes no TNF and no IL-10 from its compartment. The coarse receptor sink of S6 iterates the macrophage and T cell lists only, and the neutrophil's own sink routine does nothing. The M3 molecular loop also omits neutrophils (M3.2). S6.1 lists every agent type the sink does not cover.

M9.8 Caseation and entrapment

S5 owns the killing count, the caseation predicate, the cascade, entrapment and healing. Neutrophils are the only agents that can stand in caseous compartments (M9.3, M9.4), which is the source of most of their differences.

Contribution to the killing count

A neutrophil's senescent death adds one increment with probability `Core_Neutrophil_probabilityThatDeathContributesToCaseation`. One uniform draw is made on every senescent death, whatever the outcome. This is the only probabilistic contribution to the killing count in the model; for every other type the route of death fixes it. A neutrophil killed by the caseation cascade adds one increment always, with no draw.

On a successful senescence draw, the increment names the neutrophil, which is killed at once, and the S5.3 caseation test and cascade follow. A contributing death in a compartment that is already caseous therefore kills every other neutrophil standing there. So does any other increment there, for example a T cell daughter discarded at commitment (S3). Each death adds one.

Entrapment

The entrapment sweep (S5.4) skips neutrophils before its kill. A neutrophil in a compartment caseated by entrapment survives, adds nothing to the killing count, and may stay or move on.

Lifespan reduction in damaged tissue

At the start of its active handler, on every timestep, a neutrophil at compartment p, with killing count K(p) there and death time D, tests

$$K(p) > \texttt{Core_Neutrophil_caseumReducesNeutLifespan_threshold}.$$

If the strict test holds, its death time becomes

$$D \leftarrow t + \lfloor f \cdot (D - t) \rfloor, \qquad f = \texttt{Core_Neutrophil_caseumReducesNeutLifespan_byHowMuch}.$$

- **f is the fraction of remaining life retained, not removed.** At f = 1 the rule leaves the death time unchanged. At f = 0 the neutrophil dies of senescence on the next timestep.
- **The reduction compounds.** While the neutrophil remains in damaged tissue, its remaining life R at successive timesteps satisfies $R' = \lfloor fR \rfloor - 1$. For f < 1 it therefore reaches 0 within at most about ln R / ln(1/f) timesteps, and the neutrophil dies on the next timestep.
- **The test reads the count, not the caseation predicate.** The threshold may lie below the caseation

threshold, so partly damaged tissue shortens life. A compartment caseous only by entrapment shortens life only if its own count exceeds the threshold.

The reduction is applied before kill or uptake, and never on the timestep of senescence. A neutrophil that leaves damaged tissue keeps the death time it has reached.

Healing

Neutrophils initiate no healing. `Core_Neutrophil_probHealCaseation` is declared and read but used by no rule. Healing is initiated only by activated macrophages (S2, S5).

Consequences for the killing count

Nothing caps the killing count. Apart from neutrophils, the only increments a caseous compartment receives come from T cell daughters discarded there at commitment (S3), since no other agent can stand there. Neutrophils add one for each contributing senescent death, and one for every other neutrophil killed by any increment there. Healing halves the count once per event (S5.8), so a compartment charged in this way may remain caseous after healing.

M9.9 Observation

S9 owns the statistics, their reset codes and the output streams. M9.9 lists the statistics written only by or for neutrophils, or with a neutrophil-specific meaning.

Statistics zero in the default configuration

Statistic	Content	Reset	Writer
`NrOfAgents` for neutrophils; `NrOfNeutrophils` by state	Neutrophils present, by state	Agent pass	State commitment
`TotalNeutrophil` by state	Neutrophils recruited over the run	Cumulative	Recruitment only
`NrOfNeutrophilsUnscaled`, `NrOfNeutrophilsScaledRing`, `NeutrophilsScaledRingFactor`, `NrOfNeutrophilsScaledExtent` by state	Neutrophils within the granuloma (S9)	Agent pass	Observation block
`NrOfNeutrophilsInGran`	Neutrophils in threshold-granuloma compartments	Timestep	Grid sweep
`NeutrophilCaseationRec`	Neutrophils in caseous surroundings (S9)	Timestep	Observation block
`TotIntMtb` for neutrophils	Live bacteria held by neutrophils	Timestep	State commitment
`TotUptakeKillNeut`	Extracellular bacteria killed by neutrophils	Cumulative	Kill or uptake
`NeutrophilWithIntMtb`	Neutrophils holding live bacteria	Timestep	Next-state computation
`NeutrophilWithIntMtb_cumulative`	Neutrophil-timesteps holding live bacteria	Cumulative	Next-state computation

Statistic	Content	Reset	Writer
`NeutrophilWithIntMtb_ExceedsMax`	Neutrophils holding more than `Core_Neutrophil_maxNrIntMtbAllowed` live bacteria	Timestep	Next-state computation
`NrNeutsDieOutsideGran`, `NrExtMtbOutsideGranFromNeuts`	Senescent neutrophils outside the granuloma approximation, and the live bacteria they release	Timestep	Senescence
`NrTimesWhereMoveBlockedByBigCell` for neutrophils	Blocked movement attempts (M9.4)	Timestep	Movement
`NrRec`, `RecRate`, `NrRecruited` for neutrophils	Recruitment count, rate and per-timestep count	As in S9.17	Recruitment
`NrSourcesActive` for neutrophils	Slots at which the predicate returned true	Timestep	Recruitment
`NrSourcesBlocked` for neutrophils	Open-slot evaluations at which the crowding rule reported blocked (M9.5)	Timestep	Recruitment

Statistics non-zero in the default configuration

Statistic	Content	Reset
`NrSources` for neutrophils	Sources passing the eligibility test, once per source at slot 0	Timestep
`NrSourcesWhereRecBlockedByBigCell` for neutrophils	Open-slot evaluations at which a large agent shares the source with a free slot	Timestep
`NrSourcesWhereLiveExtMtbExceedsMax` for neutrophils	Open-slot evaluations at which s_ext was clamped	Timestep

These are written by the neutrophil predicate although no neutrophil exists, so a non-zero value is not evidence of recruitment. The exceedance count is distinct from `NrCompartmentsWhereLiveExtMtbExceedsMax`, which belongs to the extracellular pass and counts compartments above `Mtb_growthExtMtbBound` (S4).

Properties that affect interpretation

Cumulative counts. `TotalNeutrophil` counts recruited neutrophils only; Init-section, randomly placed and injected neutrophils are never added. `NeutrophilWithIntMtb_cumulative` counts neutrophil-timesteps, not distinct neutrophils.

Cognate counts. The cognate and non-cognate recruitment counts and rates are always zero for neutrophils.

Written columns. Several neutrophil statistics are maintained but not written to the general statistics file: `TotalNeutrophil`, `NrSourcesActive` for neutrophils, and `NrRec` and `RecRate` for every type. The neutrophil recruitment count written is the per-timestep `NrRecruited`. `NrSourcesWhereLiveExtMtbExceedsMax` is written for neutrophils only. S9 holds the full column inventory.

Agent dump. The agent dump writes one row per neutrophil: time, identifier, agent type, state and position. It writes no intracellular count. Because the dump follows the removal of dead agents, every neutrophil row records the active state. Neutrophils are absent from the T cell interaction file. They appear in the molecular-tracking file only when all agents are tracked, with zeros in the macrophage and T cell columns.

Bacterial burden. Neutrophil-held bacteria are included in the total intracellular statistic, and therefore in the bacterial burden and in the bacterial criterion for the onset of T cell recruitment (S8). They are also written separately as the neutrophil intracellular column of the general statistics file. On the timestep a neutrophil dies by the cascade, its intracellular bacteria are still counted in the burden, although they are destroyed at removal. A senescent neutrophil has already released its bacteria, so it adds nothing.

Injection. Injection timing and its effect on the statistics are specified in M8.9.

M9.10 Parameters

Range-build and omitted-entry rules are specified in spine 7.0 ("Validity ranges" and "Resolution"). In the Value column, "—" marks a value that is read but has no effect while no neutrophil exists.

M9.10.1 The Core_Neutrophil group

The group has 21 entries. Only `initNumber` and `initDensity` carry declared defaults, 0 and 0.0. `Core_Neutrophil_injectionTimestep` and `Core_Neutrophil_injectionCount` are tabled in M8, which owns injection.

Identifier	Value or range	Units	Meaning
`Core_Neutrophil_maxAge`	—	integer, timesteps	Lifespan; bounds every birth-time draw (M9.2). Declared minimum 0, not checked[1]
`Core_Neutrophil_initDensity`	0.0	fraction of compartments	Initial density; primary of the density-count pair (M9.10.3). [0, 1], checked
`Core_Neutrophil_initNumber`	0, computed	integer, count	Number placed at random (M9.3). Minimum 0, not checked
`Core_Neutrophil_movesPerTimestep`	—	integer, attempts per timestep	Movement attempts (M9.4). Minimum 0, not checked; a negative value behaves as 0
`Core_Neutrophil_liveCount_MovWeight`	—	—	Movement weight w_L (M9.4). Minimum 0, not checked[2]
`Core_Neutrophil_nKillings_MovWeight`	—	—	Read; used by no rule. Minimum 0, not checked
`Core_Neutrophil_CCL5_MovWeight`	—	—	Read; used by no rule. Minimum 0, not checked
`Core_Neutrophil_probMoveToBigOrNeut`	1.0	probability	Crowding-rule entry probability (M9.3). [0, 1], checked
`Core_Neutrophil_nrExtMtbUptakeNeut`	—	integer, bacteria	U_n, and the kill-all biomass threshold (M9.6). No range[3]
`Core_Neutrophil_probKillExtMtbNeut`	—	probability	p_n (M9.6). [0, 1], checked
`Core_Neutrophil_maxNrIntMtbAllowed`	—	integer, bacteria	C_n, and the cap-exceeded threshold (M9.6, M9.9). No range
`Core_Neutrophil_probabilityThatDeathContributesToCaseation`	—	probability	Senescent contribution to the killing count (M9.8). [0, 1], checked
`Core_Neutrophil_caseumReducesNeutLifespan_threshold`	—	integer, killings	Threshold of the strict lifespan-reduction test (M9.8). Minimum 0, not checked[4]

Identifier	Value or range	Units	Meaning
`Core_Neutrophil_caseumReducesNeutLifespan_byHowMuch`	—	fraction	f, fraction of remaining life retained per timestep (M9.8). [0, 1], checked
`Core_Neutrophil_probHealCaseation`	—	probability	Read; used by no rule (M9.8). [0, 1], checked
`Core_Neutrophil_dTNF`	—	molecules/s	TNF secretion rate, applied with factor 0.1 (M9.7). Minimum 0, not checked
`Core_Neutrophil_dIL10`	0.02	molecules/s	Read; multiplied by 0 (M9.7). Minimum 0, not checked
`Core_Neutrophil_dTGFBInactive`	—	molecules/s	Latent TGF-β secretion under M4 (M9.7). Minimum 0, not checked
`Core_Neutrophil_tgfbBindingRate`	—	—	Active TGF-β binding under M4 (M9.7). Declared [0, 1] and checked; applied as an amount per second (M4.3)

None of these constraints is checked at load:

[1] `Core_Neutrophil_maxAge` must be at least 2 for the birth-time draws of Init-section entries, random placement and recruitment to be defined.

[2] `Core_Neutrophil_liveCount_MovWeight` must not exceed 1, or the derived movement weights are negative (M9.4).

[3] `Core_Neutrophil_nrExtMtbUptakeNeut` must be non-negative; it is both a biomass threshold and an integer count.

[4] `Core_Neutrophil_caseumReducesNeutLifespan_threshold` is independent of the caseation threshold. Setting it at a value the count cannot reach disables the reduction.

M9.10.2 The Recruit_Prob_Neutrophil and neutrophil molecular groups

The Recruit_Prob_Neutrophil group has 10 entries, none defaulted, all read at load in every configuration. Under probabilistic M2, all except the two unused weights enter the neutrophil predicate at every open slot (M9.5).

Identifier	Value or range	Units	Meaning
`Recruit_Prob_Neutrophil_minRecProb`	0	probability	p_min. [0, 1], checked[5]
`Recruit_Prob_Neutrophil_maxRecProb`	0	probability	p_max. [0, 1], checked
`Recruit_Prob_Neutrophil_depletionTimestep`	0	integer, timestep	d_0; negative disables depletion. Minimum 0, not enforced[6]
`Recruit_Prob_Neutrophil_depletionEndTimestep`	-1	integer, timestep	d_1; negative means no end. Minimum 0, not enforced[6]

Identifier	Value or range	Units	Meaning
`Recruit_Prob_Neutrophil_liveCount_RecWeight`	—	weight	ω_{ext}. [0, 1], checked[7]
`Recruit_Prob_Neutrophil_neutCount_RecWeight`	—	weight	ω_{neut}. [0, 1], checked[7]
`Recruit_Prob_Neutrophil_Mi_RecWeight`	—	weight	Read; used by no rule. [0, 1], checked, so a file must supply a value in range
`Recruit_Prob_Neutrophil_CCL5_RecWeight`	—	weight	Read; used by no rule. [0, 1], checked, so a file must supply a value in range
`Recruit_Prob_Neutrophil_minThresholdRecChemokine_neut`	—	molecules per compartment	Lower bound of the CCL5 window. Minimum 0, not checked[8]
`Recruit_Prob_Neutrophil_maxThresholdRecChemokine_neut`	—	molecules per compartment	Upper bound of the CCL5 window; per-compartment denominator of s_CCL5. Minimum 0, not checked[8]

[5] $p_{\min} \leq p_{\max}$ is not checked. If inverted, the probability decreases as the signal rises, within [0, 1].

[6] Checked at load: $d_1 \geq 0$ with $d_1 \leq d_0$ is a fatal error (M9.5).

[7] $\omega_{\text{ext}} + \omega_{\text{neut}} \leq 1$ is required for non-negative derived weights and S in [0, 1]; each weight is checked, their sum is not.

[8] The program does not check that the lower bound is at most the upper. The clamp tests the lower bound first, so an inverted window gives every value at or above the lower bound the upper bound. An upper bound of 0 gives a zero denominator and s_CCL5 = 0.

Molecular_TNF_Neutrophil (10 entries, none defaulted). The TNFR1 and TNFR2 means and half-widths are read at construction and stored. `kTaceNeut` is stored at construction. `kSynthNeut` and `kRNANeut` are scaled by 0.1 and stored at each secretion call. The receptor means and half-widths, `kTaceNeut`, `kSynthNeut` and `kRNANeut` are each declared with minimum 0, not checked. `thresholdNFkBTNF` (no range), `saturationFracNFkBTNF` ([0, 1], checked) and `kNFkB` (no range) are never read.

Molecular_IL10_Neutrophil (7 entries, none defaulted). The IL-10 receptor mean and half-width are read at construction and stored. `IkSynthNeut` is multiplied by 0 at each secretion call. `halfSatIL10mRNA` is never read. The receptor mean and half-width, `IkSynthNeut` and `halfSatIL10mRNA` are each declared with minimum 0, not checked. `saturationFracSTAT3IL10`, `thresholdSTAT3IL10` and `kSTAT3IL10` (no range) are never read.

The construction-read molecular parameters determine draws whenever a neutrophil is created. Each receptor half-width must be strictly positive; a zero half-width makes the draw degenerate.

M9.10.3 Constants, computed values and inert parameters

Fixed structural constants: small size class; movement split $w_K = w_C = \frac{1}{2}(1 - w_L)$; recruitment split $\omega_{\text{Mi}} = \omega_{\text{CCL5}} = \frac{1}{2}(1 - \omega_{\text{ext}} - \omega_{\text{neut}})$; TNF state factor 0.1; IL-10 state factor 0; 2 small agents per compartment; M_N = 9 in two dimensions, 27 in three.

Computed values. `Core_Neutrophil_initNumber` and `Core_Neutrophil_initDensity` form a density-count pair (spine 7.0, "Computed parameters"). With the density read, the count is $\lfloor$ `Core_Neutrophil_initDensity` $\cdot N_{\text{grid}} \rfloor$ and any count in the file is overwritten. With only the count read, the reverse derivation yields a density of 0 for any count below N_grid; no rule reads it. If neither member is read, loading fails with a fatal error (spine 7.0, "Computed parameters").

Read but without effect: `Core_Neutrophil_nKillings_MovWeight`, `Core_Neutrophil_CCL5_MovWeight`, `Core_Neutrophil_probHealCaseation`, `Recruit_Prob_Neutrophil_Mi_RecWeight`, `Recruit_Prob_Neutrophil_CCL5_RecWeight`, the never-read molecular entries, and `Core_Neutrophil_dIL10` and `Molecular_IL10_Neutrophil_IkSynthNeut`, which are multiplied by 0.

M9.11 Interfaces

With	Provides or receives	See
Spine 2.3	Receives the occupancy rules and the hybrid update; provides the neutrophil size class and the neutrophil admission conditions, which spine 2.3 links	M9.2, M9.3
Spine 3	Receives the global type orders and the split half-step	M9.1, M9.7
Spine 5.4	Receives the Init-section element order, position forms and placement-loop order	M9.3
Spine 2.1, 2.4	Receives the significant-figure comparison and the derived chemokine ratio	M9.5
Spine 7.0	Receives the range-build, omitted-entry and density-count-pair rules ("Validity ranges", "Resolution", "Computed parameters")	M9.10
Spine module-slot table	Provides the M9 summary	M9.1
S1	Receives `Core_minChemotaxis` and `Core_maxChemotaxis` for the per-compartment CCL5 clamp; provides neutrophil movement and admission, which S1 links	M9.3, M9.4
S2	Receives the resting-macrophage extracellular rule that kill or uptake mirrors, and `Core_Mac_nrIntMtbBurstCInf` for s_Mi; neutrophils confer no exposure memory	M9.5, M9.6
S3	Treg suppression tests neutrophils with no effect; T cell daughters may be placed beside a lone neutrophil; Fas and cytotoxic killing exclude neutrophils	M9.6
S4	Receives the intracellular growth rule and `Mtb_growthExtMtbBound`; neutrophil kill or uptake precedes the extracellular pass	M9.5, M9.6
S5	Receives the killing count, cascade, entrapment sweep and healing; provides the neutrophil contributions, entrapment exemption and lifespan reduction	M9.8

With	Provides or receives	See
S6	Receives the receptor draw mechanism, the TNF and IL-10 helpers, the derived fields and `Core_Mac_dCCL2`, `Core_Mac_dCCL5` (via r_5); provides TNF into a field neutrophils never consume	M9.2, M9.5, M9.7
S7	Neutrophils count toward cell density at sources and consume no oxygen	M9.2, M9.6
S8	Receives the source loop, slot accounting, cascade and `Recruit_Prob_Mac_thresholdRecChemokine`; provides the neutrophil predicate, its crowding draw per open slot, the signal and the creation rule; neutrophil-held bacteria enter the T cell onset criterion	M9.5, M9.9
S9	Receives the statistic definitions, reset codes, column inventory and the previous-timestep region; provides the neutrophil statistics list	M9.9
M1	Advances the TNF and TGF-β fields neutrophils secrete into; neutrophil rules read fields after the diffusion block	M9.7
M2	Probabilistic option: the predicate and signal run at every open slot; alternative option: neutrophil recruitment not described here (M2.9). Default: predicate runs, draw consumed, recruitment refused	M9.5
M3	Neutrophils omitted from the receptor sink and molecular loop under every option; no TNF under fine-grained TNF. Default: coarse option	M9.7
M4	Binding and latent secretion under either active option; Init-section entries honoured whatever the selection. Default: not selected	M9.3, M9.7
M5	Sets M_N in the saturation denominators and the destination neighbourhood. Default: two-dimensional, M_N = 9	M9.4, M9.5
M6	No drug concentration, killing, efflux update or resistance assignment for neutrophil-held bacteria; M6.18 links. Default: not selected	M9.6
M7	No neutrophil rule depends on it; neutrophils form no anti-TNF complexes	—
M8	Owns injection and its parameters (M8.9); supplies the TNF depletion flag and `Core_depletionFractionTNF`, the TGF-β depletion flag, the randomisation flag with `Mtb_Agents_growthRateIntMtbMin`, `Mtb_Agents_growthRateIntMtbMax` and `Mtb_Agents_growthRateIntMtb`, and the post-adaptive factor; exhaustion does not read neutrophils. Default: all options closed	M9.2, M9.6, M9.7
Appendix B	Indexes the creation paths, the closed-path work, the depletion window and its check, and the inert parameters, linking M9	M9.1, M9.5, M9.10