M6. Antibiotics and pharmacology module

M6.1 Slot, activation and schedule

Summary. Antibiotics are dosed into a whole-body plasma model and delivered at vascular sources, diffuse and bind in tissue, and enter macrophages by partitioning. Bacteria are killed with a per-timestep probability set by the drug concentration they experience, modified by efflux and by resistance acquired at division. Naming a fluoroquinolone also switches off density-dependent death in caseum.

What the slot controls

The module adds antibiotic pharmacokinetics, tissue transport and pharmacodynamics to the model. It acts on no host cell directly. It kills bacteria, alters per-bacterium growth rates and drug sensitivities, and moves drug among pools. Its effects on host cells arise only through the resulting changes in bacterial load and distribution.

One rule outside the module is altered by drug selection: density-dependent killing of bacteria in caseum (S4, S5) is switched off when certain drugs are named (M6.14).

The module is not selected in the default configuration. The rest of this document describes it as it behaves when selected.

Selection and the drug list

The module is selected by naming one or more drugs in the command-line option `--antibiotics-to-include`, as three-letter abbreviations separated by commas or white space. Nothing else creates the module's objects.

The parameter declarations cover eleven drugs: INH, RIF, MXF, LVX, GFX, EMB, PZA, BDQ, PTM, LZD and RPT.

For each named drug the parameter file must contain a group `Drugs.XXX`, where XXX is the abbreviation, holding every per-drug attribute (M6.17). A missing group or attribute is a fatal error at startup. A drug absent from the declarations may be named if its group holds every attribute; it is then treated as a declared drug, without range checks (M6.17).

For each named drug, in the order given, the program creates one per-drug object (M6.2) and assigns the drug the bit value 2^i, where i is its zero-based position in the list. The order is part of the configuration. It fixes:

- the order of every per-drug loop, including loops that consume random draws (M6.13);
- the bit positions by which the interaction methods index drug combinations (M6.10).

Naming the same drugs in a different order is therefore a different configuration.

Gates

The module uses three gates. Other sections refer to them as named, dosing and treatment start.

Gate	True when	Governs
Named	At least one per-drug object exists	Concentration objects on agents and bacteria; mutation at division; reset of per-drug statistics; resistance and efflux counts, concentration accumulations and above-MIC comparisons
Dosing	Named, and $t \geq$ Drugs_dosageStartTime	Consumption, partitioning, dosing, pharmacokinetics, delivery, diffusion, killing, efflux, dispersal on death, the cell-density cache, per-compartment exposure statistics
Treatment start	Named, and t = Drugs_dosageStartTime	Seeding of resistant bacteria

Two operations test $t \geq$ Drugs_dosageStartTime directly, without testing whether drugs are named: the suppression of caseum killing (M6.14), which also requires a particular drug to be named, and the regional concentration windows (M6.15), which loop over the drug list.

Drug-interaction sub-slot

The rule that combines the effects of several drugs is selected with `--drug-interaction-method`: 0, none (default); 1, INDIGO; 2, Loewe; 3, Bliss (M6.10). Methods 1 to 3 require `--drug-interaction-file`, which names a file for INDIGO and a directory of lookup tables for Loewe and Bliss. A method value of 4 or more is fatal. The options are otherwise checked against one another at startup, and a rejected combination is fatal.

Drugs named	Method	File or directory given	Accepted
No	0	No	Yes; module not selected
No	0	Yes	No
No	1 to 3	Either	No
Yes	0	No	Yes
Yes	0	Yes	No
Yes	1 to 3	No	No
Yes	1 to 3	Yes	Yes

Initialisation order

In a run started at time zero, the per-drug objects and the drug-kill object are created after the timestep synchronisation checks (M1.2) and before any agent or extracellular bacterium is placed. Every agent and bacterium created thereafter carries one concentration object per drug. Loading a saved state follows a different path (M6.16).

Behaviour before dosing begins

When drugs are named, these occur from the first timestep, irrespective of Drugs_dosageStartTime:

- every macrophage and every bacterium carries concentration objects, created with zero concentration and no resistance, without consuming random draws;
- every bacterial division performs resistance and compensatory mutation trials, which consume random draws and may produce resistant bacteria (M6.13);
- per-drug statistics are reset each timestep, and resistance counts, efflux counts, concentration accumulations and above-MIC comparisons are evaluated (M6.15).

A run that names drugs therefore diverges from a drug-free run with the same seed from the first bacterial division, even if dosing never begins within the simulated interval.

Status in the default configuration

No drug is named by default, the interaction method is 0 and no interaction file is given. Module initialisation returns before creating anything. No per-drug or drug-kill object exists, and the dosing and treatment-start gates are false at every timestep. Every concentration list is empty, so every ungated per-drug loop, including the mutation trials at each division, executes zero times and consumes no random draws. The cached cell-density grid and the two fractional inhibitory concentration grids stay zero, and the three interaction-factor averages are zero at every timestep (M6.15). Caseum killing operates unmodified. The declared drug parameters are read and validated, but none is consulted. The module's code paths are reached and return without changing state or consuming draws: they are live code with nil effect. Appendix B indexes the module's options, its status and its inert parameters.

Place in the cellular timestep

The block names are those of the spine schedule (spine Element 3).

Block	Operation	Gate	See
Start of timestep, after the next-grid copy	Seeding of resistant bacteria	Treatment start	M6.13
Statistics reset	Reset of per-drug per-timestep statistics	Named	M6.15
Before the diffusion loop	Refresh of the cached cell-density grid	Dosing	M6.5
Split half-step, after oxygen consumption and before the molecular substeps	Extracellular consumption; intracellular consumption and degradation; partitioning	Dosing	M6.6, M6.7
Split half-step, after the molecular substeps and before immunomodulation and oxygen flux	Dosing; vascular delivery; plasma pharmacokinetics	Dosing	M6.3, M6.4
Diffusion solve; smoother	Diffusion of free drug; degradation of bound drug	Dosing	M6.5

Block	Operation	Gate	See
Next-state computation, macrophage handlers and death routes	Intracellular killing; intracellular efflux update; dispersal of intracellular drug on death	Dosing	M6.11, M6.12, M6.8
Next-state computation, macrophage handlers	Mutation at division of intracellular bacteria; accumulation of intracellular concentration statistics	Named	M6.13, M6.15
Extracellular pass, per compartment	Dispersal of intracellular drug from macrophages killed by entrapment	Dosing	M6.8
Extracellular pass, per compartment	Mutation at division during growth; average-concentration accumulation; resistance and efflux counts	Named	M6.13, M6.15
Extracellular pass, per compartment	Suppression of caseum killing	Named, and $t \geq$ Drugs_dosageStartTime	M6.14
Extracellular pass, per compartment	Extracellular killing; extracellular efflux update	Dosing	M6.11, M6.12
State commitment, macrophage statistics	Counts of intracellular concentration above MIC; intracellular resistance and efflux counts	Named	M6.15
Statistics block	Intracellular and total concentration grids (named); per-compartment exposure statistics (dosing); regional concentration windows ($t \geq$ Drugs_dosageStartTime); interaction-factor averages (always)	As stated	M6.15
Output	Dump of total drug grids	Named, if requested	M6.15

In the two split half-step blocks each operation is applied once per half-step with argument $\Delta t_d/2$, so twice per diffusion step, accumulating to Δt_d. In the consumption block the order is extracellular consumption over the whole grid, then the macrophage pass, then partitioning. In the vascular block the order is dosing, then delivery, then pharmacokinetics. The order of operations within each compartment of the extracellular pass is given in M6.11.

Reads and writes

Reads:

- global parameters `Drugs_*` and per-drug attributes `Drugs_XXX_*` (M6.17);
- the command-line drug list, interaction method and interaction file or directory;
- the diffusion and molecular timesteps and the ODE stepper selection;
- the vascular source list and its order;
- the killing count and caseation status per compartment;

- cell density per compartment;
- macrophage presence and state per compartment;
- live extracellular and intracellular bacterial counts, and the container location designation;
- bacterial generation time;
- the Mtb extent scaling factor from the previous timestep's statistics;
- Core_nrKillingsCaseation;
- the granuloma area and caseated compartment count from the statistics block.

Writes:

- free, caseum-bound and epithelium-bound drug, per drug, in its current and next buffers;
- plasma state per drug;
- the intracellular drug concentration of each macrophage;
- per-bacterium resistance flag, compensation flag, Mtb-specific C50 values, efflux flag and efflux window;
- per-bacterium intracellular and extracellular growth rates;
- bacterial deaths by drug killing;
- free drug in the Moore neighbours of dying macrophages;
- the cached cell-density grid and the two fractional inhibitory concentration grids;
- per-drug statistics, `TotDrugKill`, the pre-drug counts and the interaction-factor averages.

The module writes no cytokine, chemokine or oxygen field, and no state of any host cell other than the intracellular drug concentration.

M6.2 State

The module adds state at three levels:

- one per-drug object for each named drug, owned by the grid;
- one concentration object per drug on every macrophage;
- one concentration object per drug on every bacterium.

A single drug-kill object implements the selected interaction method (M6.10).

Per-drug object

Each per-drug object holds the drug's parameters (M6.17), its tissue fields, its plasma state and its statistics (M6.15).

Tissue fields are held in a pair of buffers, current and next. Each buffer holds three per-compartment fields.

Field	Storage	Diffuses	Meaning
Free drug	Padded, with auxiliary u and v arrays (M1)	Yes	Unbound extracellular drug

Field	Storage	Diffuses	Meaning
Caseum-bound drug	Unpadded	No	Drug bound to caseous material
Epithelium-bound drug	Unpadded	No	Drug bound to non-caseous, unoccupied tissue

All three are concentrations in mg/L of the compartment volume V, so they may be summed directly. Accessors that modify free drug preserve the invariant C = (u + v)/2; under the spectral solver the auxiliary arrays are maintained but not read.

The buffer pair belongs to the per-drug object, not to either main grid. Both main grids hold pointers to the same per-drug objects, so the current-versus-next copying of the main grids (spine 2.2) does not apply to drug fields. The two drug buffers are exchanged only by the drug diffusion step and by the smoother (M6.5). Every other operation of the module reads and writes the current drug buffer: consumption, partitioning, vascular delivery, dispersal on death, killing, efflux and observation.

Plasma state comprises these quantities per drug, all zero at creation.

Quantity	Units	Role
Blood concentration	mol/L	Read by vascular delivery (M6.4); advanced by the PK model (M6.3)
First absorption compartment	mg/kg	Receives doses
Second absorption compartment	mg/kg	Used by PK model 1 only
Peripheral compartment	mg/kg	Distribution compartment
Lung compartment	mg/kg	Never advanced; remains zero
Intracellular macrophage concentration	mg/L	Written by no operation; remains zero
Integration vector	Five entries	Working state for the PK step

Free drug may be initialised from the Init section of the parameter file. An entry whose element name is the drug abbreviation, with a non-negative `value` attribute, sets free drug at the given position, or at every compartment if `pos` is `all`. A negative or missing value, or an invalid position, is fatal. Bound drug always starts at zero. This initialisation occurs when the per-drug object is created and is not repeated when a saved state is reloaded (M6.16).

Concentration objects

Every macrophage and every bacterium carries a list of concentration objects, one per drug, in drug-list order. T cells carry none. Each object refers to its per-drug object and holds the following fields.

Field	Initial value	Used on macrophages	Used on bacteria
Concentration, mg/L	0	Intracellular drug concentration	Not used by any active rule
Resistance flag	false	Not used	Resistance to this drug (M6.13)
Compensation flag	false	Not used	Compensation of the fitness cost (M6.13)
Mtb-specific C50: extracellular, intracellular, non-replicating	The drug's `c50Extra`, `c50Intra`, `c50Nonrep`	Not used by any active rule	Half-saturation in the kill rate (M6.9) and the efflux band (M6.12)
Efflux flag	false	Never set; read by the lookup-table methods (M6.10)	Efflux pump status (M6.12)
Efflux window	Drugs_effluxInductionStatInterval entries, all false	Never updated	Rolling induction record (M6.12)

On a macrophage the object carries a concentration. On a bacterium it carries a resistance and efflux phenotype; the concentration the bacterium experiences is computed from its surroundings (M6.9). The macrophage efflux flag is never set, so any rule that reads it reads false.

Creation and inheritance

Concentration objects are created when a macrophage or bacterium is constructed, from the drug list as it then stands, without consuming random draws. This covers initial, recruited, seeding and reinfecting macrophages, and every bacterium created other than by division.

A daughter bacterium is a deep copy of its parent made at division. It therefore inherits every field of every concentration object: resistance and compensation flags, the three C50 values, and the efflux flag and window. Mutation trials are applied to the daughter after copying (M6.13).

Uptake, release, dispersal and transfer of bacteria between containers move the bacterium objects themselves, so each bacterium's phenotype travels with it. A macrophage's intracellular concentration travels with the macrophage and is not inherited by any other agent; its fate on death is given in M6.8. When a saved state created without drugs is loaded with drugs named, concentration objects are added at load time (M6.16).

Drug-kill object

A single drug-kill object is created after the per-drug objects. It is of one of three kinds, according to the interaction method: none, INDIGO, or lookup table, the last serving both Loewe and Bliss. It holds the method and, for the non-default methods, the interaction table or tables loaded at startup. None of its state changes during a run.

M6.3 Plasma pharmacokinetics and dosing

The plasma model is a whole-body compartmental model, expressed per kilogram of body mass. Each drug's plasma state evolves independently of every other drug and of the tissue grid. It is advanced inside the vascular routine of each split half-step, under the dosing gate, with argument $\Delta t_d/2$. Within one call the operations occur in this order:

```
vascular routine(t, diffusion step k, half-step length h):

    tau <- 600 * t + 60 * k                  // elapsed seconds, integer
    if tau < 600 * Drugs_dosageStartTime:
        return                               // never true under the dosing gate

    for each drug d, in list order:          // dosing
        add dose (d, tau)

    for each vascular source s:              // delivery, M6.4
        for each drug d, in list order:
            deliver (d, s, h)                // reads blood(d) before the PK step

    for each drug d, in list order:          // pharmacokinetics
        advance PK (d, h)
```

Delivery in a half-step therefore uses the blood concentration reached at the end of the previous half-step. On the first dosing half-step blood is zero and no drug enters tissue.

Dosing

Let $\tau_0 = 600\times$ Drugs_dosageStartTime and $I = 600\times$ `dosageInterval`, both in seconds. In each call, half of `dose` (mg/kg) is added to the first absorption compartment if

$$((\tau - \tau_0) + I) \bmod I = 0.$$

Within a cellular timestep τ takes the values 600t + 60k for diffusion steps $k = 0, \dots, N_d - 1$, and both half-steps of a diffusion step share the same τ. For an interval of at least one timestep, 60k is not a multiple of I for $k = 1, \dots, 9$, so the condition holds only at k = 0, and only on timesteps for which $t-$ Drugs_dosageStartTime is a multiple of `dosageInterval`. On each such timestep the two half-doses of diffusion step 0 deliver exactly one full dose. The first dose falls at t = Drugs_dosageStartTime.

Doses accumulate in the first absorption compartment. There is no limit on the number of doses and no end of treatment other than the end of the simulation. A `dosageInterval` of zero makes I zero, and the modulus by zero terminates the run.

Pharmacokinetic models

The model is selected per drug by `PKModel`. Write A_1, A_2 for the absorption compartments, P for the peripheral compartment and C for the central compartment, all in mg/kg; k_a = `kAbs` (per hour); Q = `q` and CL = `clearance` (L/(h kg)); V_c = `volDistCentral` and V_p = `volDistPeriph` (L/kg).

Model 0, two compartments with one absorption compartment:

$\frac{dA_1}{dt} = -k_a A_1, \qquad \frac{dP}{dt} = Q\left(\frac{C}{V_c} - \frac{P}{V_p}\right), \qquad \frac{dC}{dt} = k_a A_1 - Q\left(\frac{C}{V_c} - \frac{P}{V_p}\right) - CL\frac{C}{V_c}.$

Model 1, two compartments with two transit absorption compartments:

$\frac{dA_1}{dt} = -k_a A_1, \qquad \frac{dA_2}{dt} = k_a(A_1 - A_2), \qquad \frac{dP}{dt} = Q\left(\frac{C}{V_c} - \frac{P}{V_p}\right), \qquad \frac{dC}{dt} = k_a A_2 - Q\left(\frac{C}{V_c} - \frac{P}{V_p}\right) - CL\frac{C}{V_c}.$

In model 0 the second absorption compartment is neither read nor written. The choice of model is not tied to the drug: any drug may use either. A `PKModel` outside {0, 1} is rejected at startup for every declared drug, named or not (M6.17); for an undeclared drug it is fatal when the PK step is first reached.

Each PK step performs one integration step:

1. Blood B (mol/L) is converted to the central amount $C = B \times$ `MW` $\times V_c$, which is in mg/kg when `MW` is in mg/mol.
2. The state is advanced by one step of the ODE stepper selected by `--odesolver` (default fourth-order Runge-Kutta), of length $(\Delta t_d/2)/3600$ hours.
3. Each absorption and peripheral value is reset to zero if at or below 10^{-12}.
4. Blood is recovered as $B = C/$ (`MW` $\times V_c$) and reset to zero if at or below 10^{-12} mol/L.

With two calls per diffusion step and N_d diffusion steps per cellular timestep, the plasma model advances by exactly Δt_a per cellular timestep, in 2N_d = 20 steps of 30 s. Its step is fixed by the diffusion module; it does not use the molecular substep.

Plasma as a forcing function

Plasma concentration is independent of the tissue. Drug delivered into the grid is not subtracted from the central compartment, and drug returning from tissue when the local concentration exceeds the blood concentration (M6.4) is not added to it. Each drug's blood concentration is therefore a deterministic function of its PK parameters, dose, interval and start time alone. It is identical in every simulation sharing those settings, irrespective of lesion size, vascular source count, partitioning or consumption.

The lung compartment is never advanced. It remains zero, so the lung concentration statistics of M6.15 are identically zero.

M6.4 Vascular delivery

Drug enters the tissue grid only at vascular sources (S7, S8). Each source adds free drug to its own compartment by a permeability-limited flux between blood and tissue. Drug reaches all other compartments by diffusion (M6.5).

Delivery is the second stage of the vascular routine of each split half-step (M6.3). It takes argument $\Delta t_d/2$ and runs after that half-step's consumption and partitioning. The blood concentration it reads is the one reached at the end of the previous half-step.

Order

Sources are visited in the order of the vascular source list, and at each source the drugs are visited in list order. The source list is shuffled once per cellular timestep, immediately before recruitment (S8.2). That shuffle comes after the diffusion block, so delivery in timestep t uses the order produced by the shuffle of timestep $t-1$. On the first timestep it uses the order of initial placement (spine 5.2).

Delivery at one source changes free drug only in that source's own compartment. That compartment enters the local average of a second source only if the two share a face, so the result depends on source order only where two sources share a face.

Flux

For a source at compartment s, write B for the drug's blood concentration (mol/L), $\bar{C}$ for the local tissue concentration (mol/L), P = `vascularPermeability` (cm/s), ϕ = `permCoeff` (dimensionless), M = `MW` (mg/mol), A_cmpt for the lateral surface area of a compartment, 1.6×10^{-5} cm^2, and V for the compartment volume, 8×10^{-12} L (spine 2.1). The computed change in free drug, in mg/L, is

$$\Delta C = P\, A_{\text{cmpt}}\, \frac{\phi B - \bar{C}}{1000}\, \frac{\Delta t_d}{2}\, \frac{M}{V}.$$

The factor 1000 converts mol/L to mol/cm^3. The units then close to mg/L only with `MW` in mg/mol, as declared. The coefficient ϕ multiplies the blood concentration only, and so acts as an effective partition coefficient between plasma and tissue.

The local concentration $\bar{C}$ is the mean of free drug over the face-sharing neighbours of s, divided by M to give mol/L. The mean excludes s itself, and bound and intracellular drug do not enter it. It uses the boundary-clipped Von Neumann neighbourhood, so an edge source would average over fewer than four compartments; with Core_sourceBorderSize = 1 no source lies on the edge. The three-dimensional neighbour set is given in M5.3.

When $\phi B < \bar{C}$ the flux is negative and drug leaves the tissue. Drug removed in this way is not added to blood (M6.3), and so leaves the system.

Gating by compartment state

The change applied depends on the state of the source compartment. Here $\delta(s)$ is the cell density of the source (S9.3).

Source compartment	Change applied	Order of operations
Caseated	None	No drug is delivered or removed
Not caseated, $\delta(s) < 0.8$	$\max(\Delta C, -C(s))$	Add the change; then reset C(s) to zero if at or below 10^{-12}
Not caseated, $\delta(s) \geq 0.8$	$\max(v\Delta C, -C(s))$, with v = `vesselActivity`	Reset C(s) to zero if at or below 10^{-12}; then add the change

The threshold 0.8 is fixed in the code. Cell density is evaluated afresh at each call; it cannot change within the diffusion block, because no agent moves, is created or dies there and no extracellular bacterium is killed there.

The same threshold gates oxygen delivery (S7), but the two rules differ. A dense source delivers no oxygen, whereas it delivers drug scaled by `vesselActivity`, declared on [0, 1]. Drug delivery at dense sources is suppressed entirely only when `vesselActivity` is zero.

The clamp $\max(\cdot, -C(s))$ prevents a single outward flux from driving free drug below zero.

The two non-caseated branches apply the cutoff in opposite orders. In the high-density branch the cutoff is applied to the value before the increment, and the result is not cut off. A value below 10^{-12} may therefore persist after delivery there. A pre-existing value at or below 10^{-12} is also zeroed before a clamped outward change computed from it is added, which can leave a value as low as -10^{-12}. Both effects are at or below the cutoff and have no practical consequence.

Stability of the flux step

The ratio of the change in mg/L to the blood-tissue difference expressed in mg/L is

$\rho = \frac{P\, A_{\mathrm{cmpt}}\, (\Delta t_d/2)}{1000\, V}$.

A value of ρ above one would let a single step overshoot the difference. Under the default diffusion module, $\Delta t_d/2 = 30$ s and $\rho = 6 \times 10^4 P$, so each test of ρ is a bound of about 1.667×10^{-5} cm/s on P.

- **At creation and regular reload.** Each drug's permeability is tested when its per-drug object is created and on a regular reload (M6.16), with a strict inequality: $P\Delta t_d > 2 \times 1000V/A_{\mathrm{cmpt}}$, equivalent to $\rho > 1$, is fatal.
- **At timestep synchronisation.** The synchronisation checks' drug diffusivity and permeability tests (M1.2) are non-strict, rejecting $\rho = 1$. They run only on a load or restart, against the saved values. In a run started at time zero they run before any drug object exists and pass regardless of P.
- **At every delivery.** $|\Delta C| > M|\phi B - \bar{C}|$, equivalent to $\rho > 1$, is fatal.

Every failure is fatal. All these tests bound P alone; they are independent of ϕ, B and `vesselActivity`.

M6.5 Tissue transport

Free drug diffuses and degrades. Caseum-bound and epithelium-bound drug do not diffuse, but degrade at the same rate as free drug. Intracellular drug is neither diffused nor degraded here; its degradation is specified in M6.6.

Transport is performed by the selected diffusion module (M1), inside its diffusion solve, once per diffusion step, and only under the dosing gate. Before dosing begins, drug fields are neither diffused nor degraded, so free drug placed by Init-section initialisation (M6.2) stays exactly where it was placed until the dosing start time.

Spectral solver

Under the spectral module selected by default, each drug is processed in list order. Here Δt_d is Diffusion_FFT_timestepDiffusion and N_grid = n_x n_y.

```
for each drug d, in list order:
```

```
// free drug: diffusion and degradation together, no-flux boundary
X <- DCT-II of free(d, current)              // interior only
X <- X * m(d) / (4 * N_grid)                 // per-mode multiplier m_ij
free(d, next) <- DCT-III of X
set free(d, next) to zero where <= 1e-12

// bound drug: decay only
caseumBound(d, next)     <- caseumBound(d, current)     * exp(-degRateConst(d) * dt_d)
epitheliumBound(d, next) <- epitheliumBound(d, current) * exp(-degRateConst(d) * dt_d)
set each to zero where <= 1e-12

exchange the current and next buffers of d
```

The per-mode multiplier is computed once, when the per-drug object is created or reloaded, from the drug's `diffusivity` D and `degRateConst` k:

$$m_{ij} = \exp\left(-\left[\frac{2D}{\Delta x^2}\left(2 - \cos\frac{\pi j}{n_y - 1} - \cos\frac{\pi i}{n_x - 1}\right) + k\right]\Delta t_d\right),$$

where i and j index the transform modes and n_x = n_y = 300. The drug multipliers share the construction of the M1.3 propagator, including its eigenvalue arguments and storage stride; a cloned three-dimensional solver builds its no-flux plans as M5.10 describes.

Degradation is folded into the multiplier, so diffusion and first-order decay are advanced together over Δt_d. The constant mode i = j = 0 is multiplied by exactly $\exp(-k\Delta t_d)$, so total free drug is conserved by the diffusion operator and lost only by degradation.

The code evaluates the bracket in the form $(2D/\Delta x^2)(2 + k\Delta x^2/(2D) - \ldots)$. With D = 0 this is not a number, whatever k, and every free-drug value becomes not a number after the first solve. A drug with zero diffusivity is therefore not admissible under the spectral module, although the declared range of `diffusivity` admits zero (M1.3).

Drug has a no-flux boundary, unlike the zero-concentration boundary of the cytokine and chemokine fields (M1.3, "Boundary conditions"). Drug is not lost across the domain edge.

Each drug's buffers are exchanged by that drug's own step, independently of the exchange of the main grids, so the parity of N_d has no bearing on drug fields.

Diffusivity and degradation rate are fixed for the duration of a run. The multiplier is not recomputed, and no rule modifies diffusivity according to local conditions. Inputs that suggest otherwise are inert:

- `minDiffusivity`, described as the diffusivity in fully occupied compartments, is read at creation and never used;
- the cached cell-density grid is refreshed at every compartment at the start of each dosing timestep, before the diffusion loop, and is read by no rule.

Smoother

The smoother of the spectral module (M1.3, "The smoother") is invoked at the end of each split half-step. With Diffusion_FFT_numSmootherSteps = 0 it performs no iterations. With a positive value, each iteration processes each drug in list order: free drug is advanced by one explicit no-flux finite-difference step of length Diffusion_FFT_timestepDiffusionSmoother, with no degradation, into the next buffer; both bound fields are copied unchanged to the next buffer; and the drug's buffers are exchanged.

Alternative solvers

Under the alternating-direction explicit module, free drug is advanced by the no-flux variants of the two directional sweeps on its auxiliary arrays. Their average is then decayed by $\exp(-k\Delta t_d)$, with Δt_d = Diffusion_ADE_timestepDiffusion. Bound drug decays as under the spectral solver with the same timestep, and buffers are exchanged per drug. This module relies on the accessor invariant of M6.2, which every module operation other than the spectral solve preserves.

Under the Crank-Nicolson module, each drug carries its own solver object with a no-flux boundary, constructed at creation from the drug's diffusivity (M1.5).

Accuracy check

The synchronisation checks (M1.2) include a test that Δt_d is below $\kappa/(D_{\max}L^{-2})$, where D_max is the largest drug diffusivity and L^{-2} is the inverse-length term of the cytokine test (M1.2). Like the permeability test of M6.4, it sees drugs only on a load or restart of a state created with drugs. In a run started at time zero no drug diffusivity is checked. Under the spectral module, with $\kappa = 10^4$, this has no practical consequence. Under the alternating-direction explicit module, with $\kappa = 2$, a drug diffusivity above the accuracy limit is not detected in such a run.

M6.6 Consumption and intracellular degradation

Bacteria consume drug from the pool in which they reside: extracellular bacteria consume free drug in their compartment, and intracellular bacteria consume the intracellular drug of their host macrophage. Intracellular drug also degrades. Both operations run in every split half-step under the dosing gate, with argument $\Delta t_d/2$, before partitioning. Extracellular consumption is applied over the whole grid first, then the macrophage pass. Drugs, compartments and macrophages are independent of one another, so neither order affects the result. Neither operation consumes random draws.

Extracellular consumption

At each compartment, write n for the live extracellular bacterial count (a count, not a biomass sum), C for free drug (mg/L), k_c = `consumptionRate` (M/s per bacterium), H = `consumptionHalfsat` (mg/L) and M = `MW` (mg/mol). If $n \neq 0$ and $C \neq 0$, free drug changes by

$$\Delta C = -k_c \, \frac{C}{C+H} \, \frac{\Delta t_d}{2} \, n \, M$$

in mg/L, since M/s $\times$ s $\times$ mg/mol is mg/L. Otherwise it is unchanged.

Every live extracellular bacterium consumes, whether its container is designated replicating or non-replicating and whether or not its compartment is caseated. Dead bacteria do not consume. Bound drug is not consumed, and neither is the intracellular drug of a macrophage in the same compartment.

No floor is applied. If the computed loss exceeds C, free drug becomes negative. The partitioning that follows in the same half-step resolves the negative value in a way that depends on the mode:

- under steady-state partitioning (M6.7), the compartment is re-equilibrated from its total drug mass, so the deficit is drawn from the bound and intracellular pools; only if the total is negative does every pool come out negative and get reset to zero by the cutoff, which adds mass;
- under dynamic partitioning (M6.7), the first integration substep starts from the negative value, and unless that substep restores a non-negative value the run stops with a fatal error.

Intracellular consumption and degradation

Each macrophage is visited in macrophage-list order. Write m for its live intracellular bacterial count, C_i for its intracellular concentration (mg/L of the macrophage volume V_mac) and k_i = `degRateConstInt` (s^{-1}). For each drug:

1. If the macrophage's current state is infected or chronically infected, intracellular drug is consumed:

$$C_i \leftarrow C_i - \min\left(k_c \, \frac{C_i}{C_i + H} \, \frac{\Delta t_d}{2} \, m \, M, \; C_i\right).$$

2. In every state, intracellular drug decays: $C_i \leftarrow C_i \exp(-k_i \Delta t_d / 2)$.

Consumption is capped at the available drug, unlike extracellular consumption. No cutoff is applied to the result.

The state tested is the current state, not the next state. An activated macrophage holding bacteria therefore does not consume intracellular drug, although it degrades it.

The intracellular rule does not test for C_i = 0. With `consumptionHalfsat` equal to zero, an infected or chronically infected macrophage with no intracellular drug evaluates 0/0, and its concentration becomes not a number whatever its bacterial count. The extracellular rule avoids this case because it is skipped when C = 0.

Both pools use the same k_c and H, and in both the term is a concentration change in the volume of the pool concerned. At equal concentration and below saturation, an extracellular bacterium therefore consumes $V/V_{\mathrm{mac}} = 8/4.85 \approx 1.65$ times the mass consumed by an intracellular bacterium.

Uptake of drug into macrophages is performed entirely by partitioning (M6.7).

M6.7 Partitioning

Partitioning distributes drug within each compartment among four pools: free drug, caseum-bound drug, epithelium-bound drug, and the intracellular drug of a resident macrophage. It runs in every split half-step under the dosing gate, after consumption and before vascular delivery. Drugs are processed in list order.

For each drug the mode is selected by `partitionUsingSteadyState`: a non-zero value selects the steady-state mode, and zero the dynamic mode. Drugs in one run may use different modes. Neither mode consumes random draws, and both move drug only within a compartment.

For a compartment p, write K for its killing count, θ = Core_nrKillingsCaseation, F, B_c and E for free, caseum-bound and epithelium-bound drug (mg/L of V), and C_i for the intracellular concentration of the macrophage in p, if there is one. A compartment holds at most one macrophage (spine 2.3). Only macrophages count as hosts; T cells hold no drug and do not affect partitioning. A macrophage that has moved carries its intracellular drug

into its new compartment, where that drug enters the redistribution.

Caseum level and equilibrium ratios

The caseum level is $c = \min(K/\theta, 1)$. It is computed from the killing count, not from caseation status. A compartment caseated by entrapment with $K < \theta$ (S5) therefore binds drug only in proportion to K/θ, and a non-caseated compartment with $0 < K < \theta$ binds drug in proportion to its partial damage.

With u_c = `caseumUnboundFraction`, the unbound fraction measured for solid caseum, the code forms a dilution factor $d = \max(\theta/K, 1)$ when K > 0. The unbound fraction of the compartment is

$$u = \begin{cases} \dfrac{1}{1 - 1/d + 1/(d\,u_c)} & K > 0 \\ 1 & K = 0 \end{cases}$$

so the ratio of caseum-bound to free drug is $r_c = (1-u)/u = c(1-u_c)/u_c$, linear in the caseum level and reaching its solid-caseum value at $K \geq \theta$.

The intracellular uptake ratio is a = `cellUptake` if a macrophage is present and 0 otherwise. At equilibrium the intracellular concentration is a times the free concentration, so `cellUptake` is the intracellular-to-extracellular concentration ratio.

Steady-state mode

In steady-state mode every compartment is set, in each half-step, to the equilibrium distribution of its current total drug. The result does not depend on compartment order.

Unoccupied, undamaged compartments. If p holds no macrophage and K = 0, free and epithelium-bound drug are redistributed, with K_a = `epitheliumBindingKa`:

$$F' = \tfrac{F+E}{1+K_a}, \qquad E' = K_a F'.$$

Caseum-bound drug is neither read nor written in this branch. When healing halves K to zero (S5), caseum-bound drug in the compartment is then neither released nor bound further, and is lost only by degradation (M6.5).

All other compartments. If p holds a macrophage, or K > 0, caseum binding and cellular uptake apply, and epithelium binding does not. The total drug mass in the compartment, in mg, is T = (F + B_c + E)V + C_i V_mac, the last term present only with a macrophage. It is redistributed as

$$F' = \tfrac{T/V}{1 + r_c + a\,V_{\mathrm{mac}}/V}, \qquad B_c' = r_c F', \qquad C_i' = a F', \qquad E' = 0.$$

Epithelium-bound drug present when a compartment first acquires a macrophage or a killing is therefore released at the next partitioning, and the epithelium-bound pool is held at zero thereafter. With u_c = 1, caseum-bound drug is likewise released in full.

In both branches each new value is reset to zero if at or below 10^{-12}.

Mass check. In both branches the program compares the new total mass with the old, before the cutoff, and stops with a fatal error if they differ by more than 10^{-6} mg. The tolerance is absolute: a compartment holding 1 mg/L contains about 8×10^{-12} mg of drug, and the tolerance corresponds to a concentration of order 10^{5} mg/L in one compartment. The check detects an infinite result but not a result that is not a number, since every comparison with such a value is false. Mass removed by the cutoff is not checked.

Dynamic mode

In dynamic mode the pools of each compartment relax toward the steady-state equilibrium at finite rates. For each compartment the rate constants are computed from the compartment's state and the drug's parameters once per half-step, before the substeps.

Constant	Value	Meaning
k_fc	`caseumBindingRateConst` if $u_c \neq 1$, else 0	Caseum binding, per unit caseum level
k_rc	$k_{fc}u_c/(1-u_c)$ if $u_c \neq 1$, else 0	Caseum release
k_fe	`epitheliumBindingRateConst` if no macrophage and K = 0, else 0	Epithelium binding
k_re	k_fe/K_a, with K_a = `epitheliumBindingKa`	Epithelium release
k_out	`cellularKout`	Exit from the macrophage
k_in	a k_out	Entry into the macrophage

Uptake into the macrophage is governed by k_out as well as by a. With `cellularKout` equal to zero there is no exchange between the macrophage and the compartment. Intracellular drug then changes only by consumption, degradation (M6.6) and dispersal on death (M6.8).

The state is (F, B_c, E, I), where I = C_i V_mac is the intracellular drug mass in mg and the other three are concentrations in mg/L of V. Where no macrophage is present, I starts at zero and remains zero.

$$\frac{dF}{dt} = k_{rc}B_c + k_{re}E + k_{out}\frac{I}{V} - \left(k_{fc}\,c + k_{fe} + k_{in}\frac{V_{\mathrm{mac}}}{V}\right)F$$

$$\frac{dB_c}{dt} = k_{fc}\,c\,F - k_{rc}B_c, \qquad \frac{dE}{dt} = k_{fe}F - k_{re}E, \qquad \frac{dI}{dt} = k_{in}V_{\mathrm{mac}}F - k_{out}I$$

The system conserves the total mass (F + B_c + E)V + I exactly. At equilibrium $B_c/F = c(1-u_c)/u_c$, E/F = K_a and I/(V_mac F) = a, so the two modes share an equilibrium wherever the same constants are active.

In each half-step the system is integrated over N_m/2 molecular substeps of Δt_m, which under the selected diffusion module is 5 substeps of 6 s, totalling $\Delta t_d/2$. Each substep:

1. reads the current pools, converting the intracellular pool to mass;
2. performs one step of the ODE stepper selected by `--odesolver` (default fourth-order Runge-Kutta);
3. stops the program with a fatal error if any of the four outputs is negative or not a number;
4. writes the result back, the intracellular pool as I/V_mac when a macrophage is present.

Compartments are processed in grid order and are independent. No cutoff is applied to the results.

The stepper is explicit, and no stability check is made on the rate constants. If the product of the fastest rate and Δt_m exceeds the stability limit of the stepper, about 2 for forward Euler and about 2.8 for fourth-order Runge-Kutta, the integration oscillates and typically produces a negative value and a fatal error.

Differences between the modes

Situation	Steady-state mode	Dynamic mode
Epithelium-bound drug where a macrophage is present or K > 0	Zeroed and redistributed	Frozen (k_fe = 0, hence k_re = 0); lost only by degradation
Caseum-bound drug where no macrophage is present and K = 0	Untouched	Released at rate k_rc, since c = 0 but k_rc is unchanged
u_c = 1	No caseum binding; caseum-bound drug released in full	No binding and no release; caseum-bound drug frozen
0 < u_c < 1	Normal operation	Normal operation
u_c = 0	In every compartment with K > 0, r_c is infinite, F' = 0 and B_c' is not a number; the mass check passes and the non-number persists	Admissible; binding irreversible (k_rc = 0)
K_a = 0	Admissible; no epithelium binding	Fatal at the first partitioning, in every compartment, including those where k_fe = 0: k_re is 0/0 or infinite, and its product with E = 0 is not a number
a = 0	No intracellular uptake; existing intracellular drug released	No entry; intracellular drug exits at rate k_out
`cellularKout` = 0	Uptake by `cellUptake` regardless	No exchange with the macrophage
Small values	Reset to zero at or below 10^{-12}	Not reset
Negative free drug from consumption	Deficit drawn from the other pools	Fatal unless the first substep restores a non-negative value

A u_c in (0, 1] is therefore required in steady-state mode, and a positive `epitheliumBindingKa` for every drug in dynamic mode, even one with no epithelium binding. The declared ranges enforce neither (M6.17).

M6.8 Intracellular drug over the macrophage life cycle

Beginning and movement

Every macrophage begins with zero intracellular concentration for every drug. This applies to macrophages placed at initialisation, the seeding macrophage, recruited macrophages, and a reinfecting macrophage (M8). No drug is transferred to a macrophage at recruitment or on placement; its concentration rises only through partitioning (M6.7). Macrophages do not divide (S2), so no intracellular drug is ever inherited.

When a macrophage moves (S1), its intracellular concentration moves with it and enters the redistribution of the destination compartment at the next partitioning. The departed compartment simply ceases to have a resident macrophage: no drug is left behind, and none is taken along from the compartment's free or bound pools. Uptake of bacteria (S2) moves bacteria between containers and does not move drug.

Dispersal on death

Most death routes call a dispersal routine that returns intracellular drug to the tissue. The routine does nothing unless the dosing gate is open. When it acts, it processes each drug in list order, with C_i the macrophage's intracellular concentration and p its compartment:

1. a concentration increment of C_i/9 is added to free drug in each of the nine compartments of the Moore neighbourhood of p, including p itself;
2. the macrophage's intracellular concentration is set to zero.

The nine compartments are located through the Moore ordinal convention of spine 2.1, with periodic wrapping at the domain edge. The increment is added through the accessor that preserves the auxiliary-array invariant (M6.2), and no cutoff is applied. Every neighbour receives the increment, whatever its state: caseated compartments receive it, and a compartment holding another macrophage receives it as free drug, not into that macrophage's intracellular pool. The dispersed drug is redistributed among pools at the next partitioning.

The increment is a concentration in the compartment volume V, not in V_mac. The drug mass added is $9 \times (C_i/9) \times V = C_i V$, against a mass removed from the macrophage of C_i V_mac. Each dispersal therefore multiplies the dispersed mass by $V/V_{\mathrm{mac}} = 8/4.85 \approx 1.65$, and drug mass is not conserved on macrophage death.

Dispersal takes effect at once in the current drug buffer. It occurs during the next-state pass or the extracellular pass, so it is visible to every later step of the same timestep that reads drug, including the concentration used for extracellular killing (M6.11).

The routines that act on a dying macrophage, and the routes that reach them, are:

Routine	Dispersal of intracellular drug	Fate of intracellular bacteria	Reached from
Senescence branch of the next-state computation	Yes, after the bacteria are released	All released into the compartment	Senescence
Apoptosis routine	Yes, before the bacteria are dispersed	Half dispersed	TNF-induced apoptosis; Fas-mediated killing (S3.8)
Burst routine	Yes, after the bacteria are dispersed	All dispersed	Bursting of a chronically infected macrophage

Routine	Dispersal of intracellular drug	Fate of intracellular bacteria	Reached from
General dispersing kill routine, with a fraction	Yes, after the bacteria are dispersed	The given fraction dispersed	Caseation of the compartment and entrapment, both with fraction 1 (S5); the dispersing cytotoxic kill, with fraction 0.5 (S3.9)
Clean kill routine	No	All killed	Clean cytotoxic kill (S3.9)

The clean kill destroys intracellular drug: the macrophage is removed at state commitment with its concentration objects, and the drug they held leaves the system. The rationale is that a clean kill packages cell contents for degradation by neighbouring cells.

The concentration is zeroed by the first dispersal. A later dispersal call on the same macrophage within the same timestep, which would come from caseation triggered by the macrophage's own death, therefore adds nothing.

M6.9 Pharmacodynamics

Every kill method (M6.10) is built on three components: the drug concentration a bacterium is taken to experience, the modification of that concentration by efflux, and the conversion of a kill rate to a kill decision. The methods differ in how they combine drugs. The lookup-table methods depart from these components as given in M6.10.

Concentration seen

For a bacterium and a drug, the concentration seen is computed from the compartment p in which the bacterium lies; for an intracellular bacterium, p is the host macrophage's compartment. With F, B_c and E the free, caseum-bound and epithelium-bound drug in p:

- if p holds no macrophage, C_seen = F + B_c + E;
- if p holds a macrophage with intracellular concentration C_i for that drug,

$C_{\text{seen}} = \frac{(F+B_c+E)\,V + C_i\,V_{\text{mac}}}{V+V_{\text{mac}}}$.

The same rule is applied to intracellular and extracellular bacteria. Consequently:

- **Bound drug counts as active.** Caseum-bound and epithelium-bound drug contribute to killing in full, so binding to caseum does not reduce the concentration seen by bacteria in that compartment, only the free concentration available for diffusion.
- **An intracellular bacterium never sees the pure intracellular concentration.** It sees the volume-weighted mean of the two pools, with weights $V_{\text{mac}}/(V+V_{\text{mac}}) \approx 0.377$ and $V/(V+V_{\text{mac}}) \approx 0.623$.
- **An extracellular bacterium in a compartment holding a macrophage sees the same blended value.** With `cellUptake` above one, the macrophage raises the concentration seen by extracellular bacteria in its compartment; with `cellUptake` below one it lowers it.

The macrophage is the first agent of macrophage type found in the compartment's slots. The concentration seen is the same for every bacterium in a compartment, for a given drug, at a given moment. It is read from the current drug buffer at the moment of evaluation (M6.11).

Efflux

If the bacterium's efflux flag for the drug is set (M6.12), the concentration seen is multiplied by ε = `concentrationChangeByEfflux` to give the effective concentration $C_{\text{eff}} = \varepsilon C_{\text{seen}}$; otherwise C_eff = C_seen. The flag is read from the bacterium's own concentration object, except under the lookup-table methods for intracellular bacteria (M6.10).

Constants by location

The maximum kill rate, the half-saturation and the Hill coefficient are chosen by the bacterium's location. For extracellular bacteria, the location is the container's designation at the time of killing, which is after the reclassification of the same timestep (M6.11).

Location	Maximum kill rate E_max	Half-saturation C_{50}	Hill coefficient h
Intracellular	`intraActivity`	Bacterium's intracellular C50	`hillIntra`
Extracellular, replicating	`extraActivity`	Bacterium's extracellular C50	`hillExtra`
Extracellular, non-replicating	`nonrepActivity`	Bacterium's non-replicating C50	`hillNonrep`

The maximum kill rate and Hill coefficient are properties of the drug. The half-saturation is a property of the bacterium: it equals the drug's `c50Extra`, `c50Intra` or `c50Nonrep` for a susceptible bacterium, and that value multiplied by `foldIncreaseC50FromResistance` for a resistant one (M6.13).

Kill rate and kill decision

The kill rate for one drug is

$$k = E_{\max} \frac{C_{\text{eff}}^{\,h}}{C_{50}^{\,h} + C_{\text{eff}}^{\,h}}.$$

The kill methods combine single-drug rates, or replace them, to give one rate k per bacterium per evaluation. That rate is converted to a probability of death in the current timestep, $P_{\text{kill}} = 1 - \exp(-k)$, and the bacterium dies if a uniform draw on [0, 1) falls below P_kill.

The rate is not multiplied by any timestep. `intraActivity`, `extraActivity` and `nonrepActivity` are therefore rates per cellular timestep of 600 s, and killing is evaluated once per bacterium per timestep (M6.11).

Values	Kill rate	Effect
C_eff = 0, h > 0, $C_{50} > 0$	0	No killing

Values	Kill rate	Effect
h = 0	E_max/2 at every concentration, including zero	Killing without drug
$C_{50} = 0$, C_eff > 0	E_max	Maximal killing at any positive concentration
$C_{50} = 0$, C_eff = 0, h > 0	Not a number	The comparison with the draw is false; no killing

A Hill coefficient of zero, since 0^0 evaluates to one, produces killing at half the maximum rate in every compartment holding bacteria, whether or not drug has reached it. Under the no-interaction method this applies to any drug whose coefficient is zero. The declared ranges admit zero for the Hill coefficients and the half-saturations (M6.17).

M6.10 Interaction options

The interaction method is selected as given in M6.1. Every method is evaluated once per live bacterium per evaluation, at the call sites of M6.11.

No interaction (method 0, default)

Summary. Drugs act independently. Each bacterium is killed at the rate of whichever named drug kills it fastest at the concentration it sees. No interaction file is needed.

For each live bacterium evaluated, the single-drug kill rate k_d of M6.9 is computed for every drug d in list order, using the concentration seen, the bacterium's efflux flag for that drug and the constants for the bacterium's location. The bacterium is killed with the combined rate k = max(0, max_d k_d), converted to a probability as in M6.9. No concentration threshold is applied, so every drug contributes its rate however low its concentration.

The fractional inhibitory concentration grids are not written, and the pre-drug counts and interaction-factor accumulators of M6.15 are not updated, so the three interaction-factor averages are zero throughout a run under this method. Intracellular and extracellular bacteria are treated alike: for an intracellular bacterium the concentration seen is obtained from the compartment (M6.9).

INDIGO (method 1)

Summary. Drugs at or above a set fraction of their MIC are combined through a fractional inhibitory concentration read from a file for their combination, making them synergistic, additive or antagonistic. If none qualifies, the no-interaction rule applies. For extracellular bacteria the entry used is indexed by the number of qualifying drugs, not by which drugs qualify.

The method follows the laboratory of Sriram Chandrasekaran.

FIC table. The table has 2^n entries for n named drugs. Entry j belongs to the combination whose drugs' bit values (M6.1) sum to j. Entry 0, the empty combination, is given FIC -1; each single-drug entry is given FIC 1; every other entry is left at -1. Each accepted line of the file then overwrites the entry of its combination.

A line has exactly seven fields, separated by commas or white space: six drug abbreviations, of which any may be empty, then the FIC.

Condition	Result
A drug field names a drug that is not selected	The line is skipped without error
All drugs named on the line are selected, and the FIC is a non-negative number	The entry for the sum of their bit values is set to that FIC
No drug is named on the line	Error
The FIC field is not a valid number, or is negative	Error
Fewer than seven fields, on a line not skipped	Error
More than seven fields	Fatal at once
The file cannot be opened, or contains no lines	Fatal

Errors are collected over the whole file, and any error is then fatal. A line naming a single drug overwrites that drug's default FIC of 1, and a later line for a combination overwrites an earlier one. A misspelled abbreviation causes its line to be skipped, and no warning is issued. An FIC of zero passes the file check, which rejects only negative values.

A combination that receives no line keeps FIC -1. This is detected only when that combination is first needed for killing, which is fatal. A file covering fewer combinations than can arise in a run is therefore accepted at startup and may stop the run later.

Qualifying drugs. For each live bacterium evaluated, each drug in list order is tested. A drug qualifies if its effective concentration (M6.9) is at least Drugs_interactThreshold times the drug's `MIC` parameter. For each qualifying drug i the method records its effective concentration c_i, its maximum rate E_i, the bacterium's half-saturation $C_{50,i}$ and the Hill coefficient h_i, chosen by location as in M6.9. The reference drug r is the first qualifying drug, in list order, with the largest maximum rate.

The combination index is formed from the qualifying drugs. For intracellular bacteria it is the sum of their bit values. For extracellular bacteria it is the number of qualifying drugs, not the sum of their bit values.

Combined kill rate. If the index is zero, no drug qualified: the bacterium is evaluated by the no-interaction method, with no threshold, and 1 is added to the interaction factor (M6.15). Otherwise the FIC f of the indexed entry is read, and a negative value is fatal. Each qualifying drug's concentration is converted to the concentration of the reference drug that would produce the same effect alone:

$$\hat{c}_i = \left[\frac{C_{50,r}^{\,h_r}\, c_i^{\,h_i}}{\dfrac{E_r}{E_i}\left(C_{50,i}^{\,h_i} + c_i^{\,h_i}\right) - c_i^{\,h_i}} \right]^{1/h_r}, \qquad \hat{c}_r = c_r.$$

The equivalent concentrations are combined, and the kill rate is that of the reference drug at the combined concentration:

$$c_{\mathrm{comb}}(f) = \left(\sum_i \hat{c}_i^{\,f}\right)^{1/f}, \qquad k(f) = E_r \, \frac{c_{\mathrm{comb}}(f)^{\,h_r}}{C_{50,r}^{\,h_r} + c_{\mathrm{comb}}(f)^{\,h_r}}.$$

The bacterium is killed with probability $1 - \exp(-k(f))$, using one random draw.

An FIC of 1 makes the equivalent concentrations additive. With two drugs of equal equivalent concentration x, the combined concentration is 2^{1/f} x, so an FIC below 1 is synergistic and an FIC above 1 antagonistic. With a single qualifying drug and FIC 1, the method reduces exactly to that drug's single-drug rate.

The ratio k(f)/k(1), or zero if k(1) is not positive, is added to the interaction factor. Before each group of bacteria is evaluated, its live count is added to the pre-drug count for its location. Averages are formed in the statistics block (M6.15).

Threshold behaviour. The threshold makes the method discontinuous in concentration. While any drug qualifies, a drug just below its threshold contributes nothing. Once no drug qualifies, every drug contributes again through the fallback, however low its concentration. The kill rate can therefore rise as a drug's concentration falls below its threshold.

Values	Effect
f = 0, two or more qualifying drugs	c_comb infinite; k not a number; no kill; interaction factor contribution zero
f = 0, one qualifying drug	c_comb = 1 mg/L, and the reference drug's rate at that concentration applies
E_i = 0 for a qualifying drug other than the reference	$\hat{c}_i = 0$; the drug contributes nothing
E_i = 0 for every qualifying drug	No reference is set; k not a number; no kill

Fractional inhibitory concentration grids. After each evaluation the method writes a value to the extracellular or intracellular FIC grid at the bacterium's compartment. The value written is zero, so both grids are zero throughout every run.

Loewe and Bliss lookup tables (methods 2 and 3)

Summary. The kill rate is read from experimentally derived tables, one per drug combination, at the line nearest the concentrations a bacterium sees. Half-saturations, Hill coefficients and resistance play no part in killing. Loewe and Bliss share the code and differ only in the tables supplied.

The lookup tables follow the laboratory of Bree Aldridge.

Loading tables. `--drug-interaction-file` names a directory. Every directory entry whose name ends in `.csv` is read as a table, in the order of the directory listing, which is not specified. A table's first line is a header of six fields separated by white space:

```
method  DRUGS  nrDrugs  D  nrLines  L
```

Here `method` is `loewe` or `bliss`, `DRUGS` is the drug abbreviations joined by hyphens, D is the number of drugs and L the number of data lines; for example, `bliss RIF-PZA nrDrugs 2 nrLines 16`. The header is checked in this order:

1. a method field naming the other lookup method (`loewe` under Bliss, `bliss` under Loewe) is fatal; any

other method field is accepted;

2. the literal fields must read `nrDrugs` and `nrLines`, D must be at least 1 and L at least 2, and there must be exactly six non-empty fields, or the run is fatal;
3. if any drug in `DRUGS` is not selected, the table is skipped without error;
4. if the number of drug names differs from D, or any name is empty, the run is fatal.

The method check precedes the drug check, so a table for the other method is fatal even if it concerns only unselected drugs.

Each data line holds D concentrations, in the order of the header, then three kill rates: intracellular, replicating extracellular and non-replicating extracellular. Fields are separated by commas or white space. Every value must be a non-negative number and each line must have exactly D + 3 fields; violations are collected and then fatal. The number of data lines must equal L, or the run is fatal.

An accepted table is stored at the index given by the sum of its drugs' bit values. If two files describe the same combination, the one read later, in the unspecified directory order, replaces the earlier. After all files are read, every non-empty combination of the selected drugs, single drugs included, must have a table, or the run is fatal at startup.

Kill rate. For each live bacterium evaluated, each drug in list order is included if its effective concentration (M6.9) is not exactly zero; no threshold relative to MIC is applied. If no drug is included, the bacterium is not killed and no random draw is consumed. Otherwise the included drugs' concentrations and the sum of their bit values select a table, and the concentrations are reordered to the table's column order. The table line nearest them in summed absolute difference is found,

$$j^* = \arg\min_j \sum_d |c_d - c_{j,d}| \,,$$

with ties going to the earliest line. The rate in that line's column for the bacterium's location is used as k, and the bacterium is killed with probability $1 - \exp(-k)$, using one random draw. Table rates are therefore rates per cellular timestep. No interpolation is performed, and concentrations outside the table's range map to the nearest boundary line.

Departures from the common components. The lookup methods use the concentration seen and the efflux multiplier of M6.9, but depart from the other components.

Component	No interaction and INDIGO	Lookup methods
Half-saturation	Bacterium's Mtb-specific value	Not consulted
Maximum rate and Hill coefficient	Drug parameters by location	Not consulted
Resistance	Raises the half-saturation	No effect on the kill rate
Efflux flag, extracellular	Bacterium's own	Bacterium's own
Efflux flag, intracellular	Bacterium's own	Host macrophage's, which is never set, so efflux has no effect

Component	No interaction and INDIGO	Lookup methods
Draw when no drug is present	One draw	No draw
Interaction statistics	Updated under INDIGO	Pre-drug counts updated; the factor contributes zero

Under these methods resistance (M6.13) still alters growth rates through its fitness cost but confers no protection from drug, and efflux protects only extracellular bacteria.

M6.11 Drug killing call sites and order

Drug killing is evaluated once per live bacterium per timestep, at the point where that bacterium's container is processed, and only under the dosing gate. Intracellular bacteria are evaluated during the next-state pass, and extracellular bacteria during the extracellular pass, which follows it. At both points the concentration seen (M6.9) is read from the current drug buffer at the moment of evaluation. It therefore reflects all consumption, partitioning, delivery and diffusion of the timestep, and any dispersal from macrophages that died earlier in the same pass (M6.8).

A bacterium killed by drug is moved from its container's live list to its dead list, and its decay clock is set as for any other bacterial death (S4.10). Drug killing does not increment the compartment killing count, so it does not contribute to caseation directly. Each death adds 1 to `TotDrugKill`, which is cumulative over the run.

Evaluation within a container

Within one container, bacteria are evaluated in the order of the live list at the moment of evaluation, each once with its own random draw, under the selected method (M6.10). A bacterium killed is removed from the live list as it is evaluated, and evaluation continues with the next. Where the container has been grown earlier in the same timestep, that order is fixed by growth (S4.4): the live list is shuffled before growth, bacteria that starve are removed, and daughters are appended at the end, so daughters created in the timestep are evaluated after every surviving bacterium that existed before growth. Before the bacteria are evaluated the method may record the container's live count, and afterwards the accumulated interaction factor (M6.15).

Extracellular call site

At each compartment the sequence is:

```
for each compartment p, in grid order:

    entrapment sweep at p                // may kill entrapped agents;
                                         // macrophage deaths disperse drug (M6.8)
    if live extracellular count at p > 0:
        record the container's designation
        grow, with mutation at division  // M6.13
        reclassify the container as replicating or non-replicating
        accumulate average concentration // M6.15, using the designation recorded
```

```
                                    // before reclassification
        if caseated(p) and caseum killing not suppressed:
            caseum killing              // S4.11; M6.14
        drug killing                    // designation after reclassification
        efflux update                   // M6.12
        resistance and efflux counts    // M6.15
    heal caseation at p                 // S5.8
    decay dead bacteria at p            // S4
```

The location passed to the kill method is the container's designation after the reclassification of the same timestep, so a container that has just become non-replicating is killed with the non-replicating constants.

- A compartment whose container holds no live bacteria when it is reached is skipped entirely: no growth, no killing, no efflux update and no counting.
- Drug killing is applied in caseated compartments; no gate on caseation status applies to it.
- Dead bacteria, including those killed by drug in this pass, are decayed later in the same visit, so an extracellular drug kill begins its decay in the timestep of its death.
- Entrapment kills at p disperse intracellular bacteria and drug into the Moore neighbourhood of p (M6.8, S5.4). Bacteria dispersed into p itself, or into neighbours not yet visited, are grown and drug-killed in the same pass; bacteria dispersed into neighbours already visited are not evaluated until the next timestep. Dispersed drug likewise raises the concentration seen in neighbours visited later in the pass, but not in those already visited.

Intracellular call sites

Intracellular killing is evaluated inside the macrophage's next-state computation (S2), for the host's intracellular container, with macrophages visited in macrophage-list order. The routine is reached only in these branches.

Branch of the macrophage's next-state computation	**Sequence for intracellular bacteria**	**Drug killing**	**Efflux update**
Senescence	Released into the compartment	No	No
TNF-induced apoptosis	Half dispersed	No	No
Resting, not activating	Uptake of extracellular bacteria if infected by them (S2.5); growth of the bacteria taken up	No	No
Infected	Probabilistic uptake; growth; concentration accumulation (M6.15); decay; drug killing; efflux update	Yes	Yes
Chronically infected	Growth; concentration accumulation; decay; drug killing; burst test	Yes	No
Activated	Only if the live count is positive: removal of one bacterium by the macrophage (S2.8), then drug killing	Yes	No

Branch of the macrophage's next-state computation	Sequence for intracellular bacteria	Drug killing	Efflux update
Suppressed, infected or chronically infected	Growth; decay; drug killing; threshold tests	Yes	No
Suppressed, resting or activated	None	No	No

- In the infected, chronically infected and suppressed branches, dead intracellular bacteria are decayed before drug killing, so a bacterium killed by drug there begins its decay on the next timestep, one timestep later than at the extracellular call site.
- The activated branch calls no decay routine, so bacteria killed by drug in an activated macrophage do not decay while it remains activated (S2.8).
- An activated macrophage holding exactly one live bacterium removes it before drug killing, which then evaluates an empty list.
- In the infected branch, bacteria taken up in the same timestep are grown and drug-killed with the rest of the container.
- A resting macrophage that becomes infected grows its newly acquired bacteria but does not expose them to drug in that timestep. They are first evaluated on the next timestep, under whichever branch then applies.

Because intracellular killing precedes extracellular killing within the timestep, a bacterium released from a macrophage during the next-state pass may be evaluated twice in one timestep: intracellularly first, if its host's branch evaluates killing before the release, and then extracellularly, if the receiving compartment is visited in the extracellular pass with a non-empty container. Bacteria taken up by macrophages during the next-state pass are evaluated only intracellularly, if at all.

M6.12 Efflux pump induction

Each bacterium carries, for each drug, an efflux flag and a rolling record of recent exposure to an inducing concentration (M6.2). While the flag is set, the concentration the bacterium is taken to experience for that drug is multiplied by `concentrationChangeByEfflux` (M6.9). The flag is updated from the record once per evaluation, under the dosing gate, at the call sites of M6.11. No random draws are consumed.

Induction band

For each drug, the band is formed around a minimum inhibitory concentration derived from the bacterium's own half-saturation and the drug's Hill coefficient, chosen by location as in M6.9. The derived value is the concentration at which the Hill term reaches 90 per cent of its maximum:

$$\mathrm{MIC}_{\mathrm{eff}} = C_{50}\left(\tfrac{0.9}{1-0.9}\right)^{1/h} = C_{50}\,9^{1/h}.$$

With m_min = `minEffluxInduction` and m_max = `maxEffluxInduction`, the band is [m_min MIC_eff, m_max MIC_eff], closed at both ends.

The derived value is not the drug's `MIC` parameter. Because it scales with the bacterium's half-saturation, resistance (M6.13) raises the band by `foldIncreaseC50FromResistance`. If m_min < 0, efflux is disabled for that drug: the update returns without changing the record or the flag, which keeps whatever value it had, false unless inherited from a parent that acquired it under a different parameter value. If m_min > m_max, the band is empty.

Concentration tested

Bacterium	Concentration tested against the band	Half-saturation and Hill coefficient
Extracellular, replicating	Free drug in its compartment	Extracellular C50; `hillExtra`
Extracellular, non-replicating	Free drug in its compartment	Non-replicating C50; `hillNonrep`
Intracellular	Intracellular concentration of the host macrophage	Intracellular C50; `hillIntra`

Bound drug is excluded, no volume blending is applied, and the efflux multiplier is not applied to the tested concentration. For extracellular bacteria the location is the container's designation after the reclassification of the same timestep.

Update rule

Let W = Drugs_effluxInductionStatInterval and N = `numTimeStepsEffluxInduction`. The record is an array of W Boolean entries. At timestep t, for each drug:

```
update efflux (bacterium b, drug d, concentration c, t):

    if minEffluxInduction(d) < 0:
        return
    lo <- minEffluxInduction(d) * MICeff(b, d)
    hi <- maxEffluxInduction(d) * MICeff(b, d)
    record(b, d)[t mod W] <- (lo <= c and c <= hi)
    s <- number of true entries in record(b, d)
    flag(b, d) <- (s >= N)
```

The code sets the flag on when it is off and $s \geq N$, and off when it is on and s < N, which is equivalent to the final assignment: there is no hysteresis. With N = 0 the flag is set at the first update of any drug for which efflux is enabled, whatever the concentration.

The record is indexed by absolute simulation time modulo W, not by the number of updates the bacterium has received. Entries for timesteps in which the bacterium was not updated keep values written at earlier timesteps with the same index, so the record represents the last W timesteps exactly only for a bacterium updated at every one of them.

Before dosing begins no update occurs, so every record and flag is false at the dosing start time unless inherited otherwise.

When the update occurs

Location	Updated	Order relative to killing
Extracellular, in a compartment holding live bacteria when visited	Every timestep under dosing	After killing
Intracellular, in an infected macrophage that is not suppressed	Every timestep under dosing	After killing
Intracellular, in a chronically infected, activated or suppressed macrophage	Never	—
Intracellular, in a resting macrophage that has just taken up bacteria	Not on the timestep of uptake	—

The update follows killing at both call sites, so the flag used for killing at timestep t reflects exposure recorded up to timestep $t-1$. A bacterium killed by drug is not updated.

The record and flag travel with the bacterium (M6.2). A bacterium whose host becomes chronically infected or activated keeps the efflux state it had on the last timestep its host was infected, until it is released or its host reverts. A daughter inherits its parent's record and flag and is updated with the rest of its container in the same timestep.

The flag affects only the concentration used for killing. It does not affect consumption, partitioning, growth or mutation. Under the lookup-table methods, the flag of intracellular bacteria is not read (M6.10).

Values	Effect
W = 0	The record has no entries, and t mod W, a division by zero, terminates the run
N = 0	Flag set at the first update for any enabled drug
N > W	Flag never set
h = 0	MIC_eff infinite; the band is never entered, except at zero concentration when m_min = 0, where the product is not a number and the test is false; flag never set unless N = 0
$C_{50} = 0$	MIC_eff = 0; band [0, 0]; entered whenever the tested concentration is exactly zero

M6.13 Resistance: mutation and seeding

Bacteria acquire genetic resistance to a drug, and compensation of its fitness cost, by mutation at division, and some may be made resistant at treatment start. Both resistance and compensation are recorded per drug in the bacterium's concentration objects (M6.2), are inherited by every descendant, and are never lost.

Where mutation occurs

Mutation trials are performed whenever drugs are named, not only under the dosing gate. They therefore run from the first timestep, consume random draws, and may produce resistant bacteria before dosing begins (M6.1).

Mutation is applied only at division, only to the daughter, and only immediately after the daughter is created. Division occurs during growth (S4.9), at both growth sites: extracellular growth in the extracellular pass, and intracellular growth in the macrophage handlers. The parent is never subjected to a mutation trial, and a bacterium that never divides never mutates.

```
divide(parent, t, location, sigma):

    draw which product takes the larger biomass share   // S4.9
    daughter <- deep copy of parent                     // inherits all resistance state
    parent: biomass scaled; generation time <- t - birth time; birth time <- t
    daughter: new identifier; biomass scaled; generation time <- parent's new generation
time

    g <- parent's generation time
    pMut <- 1 - (1 - mu)^g
    repeat floor(sigma) times:
        resistance trial (daughter, pMut)
        compensation trial (daughter, pMut)
    update cumulative resistance statistics for the daughter at location   // M6.15
```

Here μ = Mtb_Agents_perBasePairMutationRate, per base pair per timestep; g is the parent's generation time in timesteps, the time since its previous division or since its creation if it has not divided before; and σ is the Mtb extent scaling factor (S9).

Mutation probability

The probability of a mutation at a given site over the parent's generation is $p_{\mathrm{mut}} = 1 - (1 - \mu)^g$, approximately μg for small μg. If g = 0, p_mut = 0 and the trials are performed but cannot succeed, still consuming draws.

Number of trials

Each daughter receives $\lfloor \sigma \rfloor$ trials, each a resistance trial followed by a compensation trial. The rationale is that in a two-dimensional simulation each daughter stands for σ daughters of a three-dimensional lesion, and so receives σ chances to mutate. S9.5 defines σ; it is at least 1, so at least one trial is always made, and it grows with the square root of the number of compartments in the Mtb extent (for an extent of 1000 compartments, 23 trials).

The value used is the one computed in the statistics block of the previous timestep; on the first timestep it is the value computed at initialisation. Under the three-dimensional configuration of M5, σ = 1 and one trial is made.

Resistance trial

For each drug in list order to which the daughter is not yet resistant:

1. the probability of resistance is $p_{\mathrm{res}} = p_{\mathrm{mut}} \times$ `numberResistMutations`, the number of point mutations known to confer resistance;
2. one uniform draw is consumed;
3. if the draw is below p_res, the daughter is made resistant to that drug.

Drugs to which the daughter is already resistant are skipped and consume no draw. p_res is not capped, so a value of one or more makes resistance certain.

Making a bacterium resistant to a drug, by mutation or by seeding, has these effects: the resistance flag for that drug is set; both of the bacterium's intrinsic growth rates, intracellular and extracellular (S4.5), are multiplied by `relativeFitnessResist`; and its three Mtb-specific half-saturations for that drug are multiplied by `foldIncreaseC50FromResistance`. A bacterium already resistant to the drug is left unchanged, so the cost and the fold increase are applied at most once per drug. Costs for different drugs multiply.

Compensation trial

At the start of each compensation trial, the daughter's two intrinsic growth rates are recorded. Then, for each drug in list order to which the daughter is resistant and for which it is not yet compensated:

1. the probability of compensation is $p_{\mathrm{comp}} = p_{\mathrm{mut}} \times$ `numberCompensatoryMutations`;
2. one uniform draw is consumed;
3. if the draw is below p_comp, the compensation flag is set, and each growth rate is set to its recorded value divided by `relativeFitnessResist`.

Compensation does not change the half-saturations, so a compensated bacterium remains resistant. Resistance acquired in a resistance trial is eligible for compensation in the compensation trial that immediately follows it.

If two or more drugs are compensated in the same trial, each assignment starts from the rates recorded at the start of that trial, so only the last takes effect: the growth rates reflect removal of the last drug's cost alone, although every compensated drug's flag is set.

The fitness cost acts through the intrinsic growth rates, which are the starting point of the effective growth rate computed at each growth step (S4.5), so it slows both growth and division. Under growth-rate randomisation the intrinsic rates are not read, so resistance carries no fitness cost and compensation has no effect (M8.7). Mutation itself, and the change in drug susceptibility it confers, are unaffected. The draws consumed by mutation at each division are made whether or not randomisation is enabled.

Values	Effect
`numberResistMutations` = 0	No resistance by mutation; draws still consumed
$p_{\mathrm{res}} \geq 1$	Every daughter becomes resistant at its first trial
`relativeFitnessResist` = 1	No fitness cost; compensation has no effect on growth

Values	Effect
`relativeFitnessResist` > 1	Resistance increases growth rates
`relativeFitnessResist` = 0	Resistant bacteria do not grow; compensation sets growth rates to 0/0, not a number
`foldIncreaseC50FromResistance` = 0	Resistant bacteria have half-saturation zero and are killed at the maximum rate at any positive concentration (M6.9)
`foldIncreaseC50FromResistance` < 1	Resistance increases susceptibility

The declared ranges admit all of these values (M6.17).

Seeded resistance at treatment start

At the first dosing timestep a specified number of existing bacteria may be made resistant, so that treatment begins against a pre-existing resistant subpopulation. Seeding runs when the treatment-start gate is true, at the start of the timestep, after the next-grid copy and before the statistics reset and the diffusion block. Seeded bacteria are therefore exposed to drug in the same timestep.

Seeding occurs at most once per run, and only if the dosing start time is reached as a simulated timestep. Timesteps are numbered from 1 (spine Element 3), so seeding never occurs with Drugs_dosageStartTime = 0, although dosing then begins at the first timestep, nor when a saved state is loaded at or after the dosing start time.

The candidate pool is built afresh each time it is needed. It consists of every live bacterium resistant to no drug, in this order:

1. extracellular bacteria, by compartment in grid order, and within each compartment in live-list order;
2. intracellular bacteria, by macrophage in macrophage-list order, and within each macrophage in live-list order.

Each candidate is recorded with the location designation of its container when the pool is built. Dead bacteria and bacteria already resistant to any drug are excluded.

```
seed resistant bacteria:

    // multi-drug seeding
    n <- Drugs_numberOfRandomMultiResistantBugsToAdd
    if n != 0:
        pool <- candidate pool
        if pool not empty:
            if n >= size(pool):
                convert every candidate, to every drug
            else:
                shuffle pool
                convert the first n candidates, to every drug
```

```
// mono-resistance seeding
for each drug d, in list order:
    n <- numberOfRandomMonoResistantBugsToAdd(d)
    pool <- candidate pool                    // rebuilt for each drug
    if pool empty:
        return                                // remaining drugs skipped
    if n == 0:
        continue
    if n >= size(pool):
        convert every candidate, to d only
    else:
        shuffle pool
        convert the first n candidates, to d only
```

Converting a candidate makes it resistant to the drug or drugs concerned, with the effects of the resistance trial. Its cumulative resistance statistics (M6.15) are then updated at its recorded location. Because the pool is rebuilt for each drug, every mono-resistant seed is resistant to exactly one drug. If any stage converts every candidate, the mono-resistance stage returns at its next drug without seeding the remaining drugs.

Values	Effect
Both counts zero for every drug	Pools built, nothing converted, no draws consumed
Multi-drug count negative	Pool built and shuffled, consuming draws; nothing converted
Mono count negative for a drug	Pool built and shuffled, consuming draws; nothing converted for that drug
A count at least the pool size	Every candidate converted; later drugs receive none
No live bacteria at the dosing start time	Nothing converted; the mono stage returns at its first drug

Negative counts are accepted, since one-sided ranges are not enforced (M6.17).

M6.14 Suppression of caseum killing

Bacteria in a caseated compartment are normally subject to a density-dependent death rule (S4.11). Selecting certain drugs disables that rule for the remainder of the run, from the dosing start time.

During the extracellular pass, caseum killing at a caseated compartment is applied only if the fluoroquinolone suppression predicate is false. The predicate is true when both of these hold:

- at least one named drug has the abbreviation MXF, LVX or GFX; and
- $t \geq$ Drugs_dosageStartTime.

The test is on the drug's name alone; no parameter of the drug is consulted, and no other drug triggers it. Because it requires a named drug, it is true only when the dosing gate is open. In a run naming any of the three drugs it is therefore true at every timestep from the dosing start time onward, everywhere on the grid, irrespective of dose, delivery or local drug concentration.

While the predicate is true:

- the caseum killing routine is not called at any caseated compartment, so no bacterium dies by that route and `TotCaseatKill` does not grow, since dispersal on caseation contributes nothing at fraction 1;
- the random draws that routine would consume are not consumed, so the draw sequence diverges from that of an otherwise identical run without the suppression;
- killing of agents when a compartment becomes caseated or entrapped (S5) is unaffected, as are the bacteria those events kill or disperse.

Bacteria in a caseated compartment have an effective growth rate of zero (S4.6), so they neither divide nor starve. With caseum killing suppressed, such a bacterium can die only by drug killing (M6.11), with the non-replicating constants; by an activated macrophage in an adjacent compartment (S2.8); or after the compartment heals (S5.8), when the ordinary rules apply. Bacteria dispersed or released into caseum therefore accumulate, and under the no-interaction method the non-replicating activity of the selected drugs decides whether caseous populations decline at all.

Selecting a fluoroquinolone therefore changes the caseum death rule of S4.11 as well as adding drug killing.

M6.15 Observation

When drugs are named, the module adds per-drug statistics, per-compartment exposure grids and an optional grid dump. Seven statistics of the main statistics table also belong to the module: `TotDrugKill`, the three pre-drug counts and the three interaction-factor averages. None of these quantities is read by any rule of the model, so observation has no effect on dynamics and consumes no random draws.

The per-drug statistics are members of the per-drug object (M6.2), not entries of the main statistics table (S9). They therefore exist only when drugs are named, and are serialised with the per-drug object.

Statistic lifetimes

Throughout M6.15, W_c = Drugs_drugConcentrationStatInterval, in timesteps.

Lifetime	**Statistics**	**Where reset or replaced**
Reset at the start of every timestep	Regional drug totals; average concentrations by bacterial class; counts above MIC; resistance-risk counts; current resistance, MDR, compensation and efflux counts; the distribution accumulator	Start of the timestep, after seeding and before the diffusion block
Recomputed every timestep	Intracellular and total concentration grids	Statistics block
Reset per compartment at the start of each statistics interval	AUC and Cmax grids, intracellular, extracellular and total	Statistics block, under dosing
Rolling window of one statistics interval	Regional concentration windows; regional AUC and Cmax	Statistics block, from the dosing start time

Lifetime	Statistics	Where reset or replaced
Cumulative over the run	Cumulative resistance, MDR and compensation counts; single-drug kill counts	Never reset

Per-drug statistics start at zero when the per-drug object is created. In a run started at time zero they are first evaluated by the statistics pass at the end of initialisation, before the first output (spine Element 5).

Intracellular and total concentration grids

At the start of the statistics block, after the granuloma extents are defined, each drug's intracellular and total grids are recomputed whenever drugs are named. For every compartment p, the intracellular grid is set to zero and the total grid to T(p) = F(p) + B_c(p) + E(p). Then, for each macrophage in macrophage-list order, the entries at its compartment are overwritten: the intracellular grid takes the macrophage's C_i, and the total grid takes the volume-weighted mean

$$T(p) = \frac{(F(p)+B_c(p)+E(p))V+C_i\,V_{\mathrm{mac}}}{V+V_{\mathrm{mac}}},$$

which is the blended value used for killing (M6.9). The macrophage list is read after dead agents have been removed, and a compartment holds at most one macrophage, so no entry is overwritten twice. Before dosing begins, the total grid reflects Init-section drug only, and the intracellular grid is zero.

These two grids feed the exposure grids, the regional totals and windows, and the grid dump.

Grid dump

The option `--dump-antibiotics`, together with `--grid-dump-timespec`, writes each drug's total grid at the timesteps of the timespec, and at the final timestep or the timestep of an early stop. The dump is written after the timestep has completed, so it reflects that timestep's statistics block. Requesting the dump without a timespec, or without any drug named, is fatal at startup. Drug fields are not valid names for `--grids-to-dump`, so the free and bound fields cannot be dumped separately.

Per-compartment exposure grids

Under dosing, the per-compartment sweep of the statistics block maintains six exposure grids per drug: AUC and Cmax, each for intracellular, extracellular and total drug. Compartments are visited in grid order, and at each compartment, for each drug:

1. if (t − Drugs_dosageStartTime) mod W_c = 0, all six grids are reset to zero at p, so the first reset falls at the dosing start time;
2. each AUC grid is incremented by its source concentration divided by the number of timesteps per hour, $\mathrm{AUC}(p) \leftarrow \mathrm{AUC}(p) + C(p)/6$, in mg h/L, the intracellular AUC using the intracellular grid, the extracellular AUC free drug, and the total AUC the total grid;
3. each Cmax grid is set to the larger of its current value and the same source concentration.

The AUC is a rectangle-rule sum of concentrations sampled once per timestep, at the end of the timestep, so interval m covers the timesteps from $t_0 + mW_c$ to $t_0 + (m+1)W_c - 1$, with t_0 = Drugs_dosageStartTime. For a compartment without a macrophage the intracellular grid is zero, so a compartment's intracellular AUC records the drug of whichever macrophages occupied it during the interval.

Counts above MIC

The counts above MIC compare concentrations with the drug's `MIC` parameter, with different concentrations, operators and gates.

Statistics	Concentration compared	Test	Gate	Evaluated at
Compartments above MIC; extracellular bacteria in those compartments	Free drug	$F(p)$ $\geq$ MIC	Dosing	Statistics block, per compartment
Macrophages above MIC; intracellular bacteria in those macrophages	Intracellular concentration	C_i > MIC	Named	State commitment, per living macrophage

The bacterial counts are live counts. Every compartment and every living macrophage is tested, with or without bacteria. The macrophage counts are made after each macrophage's state is committed, so a macrophage that died in the timestep is excluded. Before dosing begins they are zero unless `MIC` is negative, which the declared range does not prevent.

Resistance-risk counts

On the last timestep of each statistics interval, the timestep for which ($t + 1 -$ Drugs_dosageStartTime) mod W_c = 0, the program counts the bacteria exposed to drug in a range associated with selection for resistance. The count is made under dosing, in the per-compartment sweep, after that timestep's exposure grids have been updated. For each compartment, an exposure measure and a pair of bounds are selected for each drug in list order by `resistRiskIndicatorFlag`:

Flag	Intracellular exposure X_i	Extracellular exposure X_e	Lower bound L	Upper bound U
0	Intracellular AUC	Extracellular AUC	`aucRiskLB` $\times$ W_c/144	`aucRiskUP` $\times$ W_c/144
1	Intracellular Cmax	Extracellular Cmax	`cmaxRiskLB`	`cmaxRiskUP`
Any other value	As set for the previous drug at this compartment, or zero for the first drug	As for X_i	As for X_i	As for X_i

The AUC bounds are thus treated as values per day, scaled to the length of the statistics interval; the Cmax bounds are used unscaled. An exposure X is at high risk relative to a half-saturation C_{50} if $L \leq X/C_{50} \leq U$, where C_{50} is the drug's parameter, not the bacterium's Mtb-specific value, so resistant and susceptible bacteria are assessed alike.

Condition on the compartment	Exposure and half-saturation	Counts incremented
Holds a macrophage	X_i and `c50Intra`	Intracellular bacteria at risk, by the macrophage's live count; macrophages at risk, by one
Not caseated	X_e and `c50Extra`	Extracellular bacteria at risk, by the live extracellular count; compartments at risk, by one
Caseated	X_e and `c50Nonrep`	Non-replicating bacteria at risk, by the live extracellular count; compartments at risk, by one

The extracellular tests are split by caseation status, which includes caseation by entrapment, not by the container's designation. Every compartment is tested, with or without bacteria, and a macrophage is counted if its compartment meets the intracellular test, whether or not it holds bacteria.

The risk counts are reset at the start of every timestep, so they are non-zero only in the statistics of the last timestep of each interval. An output stream whose cadence misses those timesteps records them as zero throughout. A W_c of zero makes both modulus tests divide by zero, which terminates the run.

Regional drug totals

Under dosing, the per-compartment sweep accumulates five regional sums of the total grid for each drug, and one distribution accumulator, after the total grid has been recomputed.

Statistic	Compartments contributing
Whole grid	Every compartment
Granuloma	Compartments with cell density at least the area threshold, or caseated
Normal tissue	All other compartments
Caseum	Caseated compartments, by killing count or by entrapment
Partial caseum	Compartments whose killing count is at least Core_partialCaseationFraction $\times$ Core_nrKillingsCaseation

Each sum adds T(p), so it is a sum of concentrations over compartments, not a drug mass.

The granuloma and normal-tissue sums partition the grid. The test is the threshold definition of the granuloma (S9), with the area threshold of 0.5, evaluated directly from cell density and caseation, not read from the stored membership flag. The two coincide unless M4 freezes the flag (M4.8). The partial-caseum sum is not a subset of the caseum sum: compartments caseated by entrapment with a low killing count are in the caseum sum only. The distribution accumulator supplies the minimum and maximum of the total grid.

Regional concentration windows

From the dosing start time, after the per-compartment sweep, each drug's regional concentrations are recorded in rolling windows used for regional AUC and Cmax. This step tests $t \geq$ Drugs_dosageStartTime directly and loops over the drug list, so it does nothing when no drug is named.

Write N_grid = 90000 for the number of compartments, n_g for the granuloma compartment count of the cell-density extent (S9), which is the number of compartments contributing to the granuloma sum, and n_c for the number of caseated compartments counted in the same sweep. At timestep t, entry t mod W_c of each window is overwritten as follows.

Window	Value entered	Units
Whole grid	Whole-grid sum / N_grid	mg/L
Normal tissue	Normal-tissue sum / $(N_{\text{grid}} - n_g)$	mg/L
Granuloma	Granuloma sum / n_g	mg/L
Caseum	Caseum sum / n_c	mg/L
Blood	Blood concentration (M6.3)	mol/L
Lung	Lung compartment / `volDistLung`, or zero if that is not positive	mg/L; always zero (M6.3)
Peripheral	Peripheral compartment (M6.3)	mg/kg

The regional windows hold means of the blended total grid, including bound drug and the intracellular drug of resident macrophages. No window is kept for partial caseum.

The window index is absolute simulation time modulo W_c, not time since the dosing start. Window entries are zero at creation and are not written before the dosing start time, so for the first $W_c - 1$ timesteps of dosing the unwritten entries hold zeros.

Regional AUC and Cmax

After the windows are updated, each drug's regional AUC is the sum of its window divided by the number of timesteps per hour, $\sum_j w_j/6$, in mg h/L for the tissue regions. The blood window is multiplied by `MW`, giving mg h/L; the peripheral window is used as recorded, giving mg h/kg. Cmax is the largest window entry, computed for the whole grid, normal tissue, granuloma, caseum, and blood after multiplication by `MW`; none is computed for the lung or peripheral windows.

The granuloma AUC and Cmax are set to zero if n_g = 0 at the current timestep, and the caseum AUC and Cmax likewise if n_c = 0. No such guard applies to the other regions. These values cover the last W_c timesteps exactly only once W_c timesteps of dosing have elapsed.

When no compartment is caseated, the caseum entry is 0/0, not a number; when there is no granuloma, the granuloma entry is likewise not a number. A not-a-number entry stays in its window until it is overwritten W_c timesteps later. The guards test only the current timestep, so after a region first appears, or reappears, within the window, the regional AUC is not a number for up to $W_c - 1$ timesteps, and the regional Cmax depends on where the non-numbers lie, because the maximum is taken by a comparison that is always false for a non-number. A granuloma covering the whole grid would make the normal-tissue entry 0/0 in the same way.

Average concentrations by bacterial class

The per-drug averages by bacterial class are evaluated whenever drugs are named.

Statistic	Numerator	Denominator	Evaluated
Average in infected macrophages	Sum of intracellular concentration over the macrophages processed by the infected and chronically infected branches (M6.11)	Number of infected and chronically infected macrophages after state commitment	Numerator in the next-state pass; division in the statistics block, from the dosing start time
Average in compartments with replicating extracellular bacteria	Sum of free drug over compartments whose container is designated replicating	Number of such compartments, incremented once per drug	Extracellular pass
Average in compartments with non-replicating extracellular bacteria	Sum of free drug over compartments whose container is designated non-replicating	Number of such compartments, incremented once per drug	Extracellular pass

A zero denominator gives an average of zero. All of them are reset at the start of every timestep.

For the two extracellular averages, a compartment contributes if its container holds live bacteria when it is reached in the pass. It is classified by the designation recorded before that visit's reclassification (M6.11), and the concentration added is free drug at that moment, after any dispersal earlier in the pass. These are averages over compartments, not over bacteria. With n drugs named, each drug's extracellular average is 1/n of the mean free drug over the contributing compartments.

The numerator of the infected-macrophage average is summed over macrophages whose state before commitment was infected or chronically infected and which reached the corresponding branch; its denominator counts macrophages whose committed state is infected or chronically infected. The two populations differ: macrophages dying by senescence or apoptosis, or suppressed, contribute to neither; macrophages newly infected by uptake in the resting branch contribute to the denominator only; and macrophages that burst, or revert to resting, contribute to the numerator only. Before the dosing start time the division is not performed, and the reported value is the raw sum, which is zero.

Resistance and efflux counts

Each drug carries current resistance counts, reset at the start of every timestep, cumulative resistance counts, never reset, and current efflux counts. Every count is kept in four categories by the location of the bacterium counted: intracellular; extracellular, which includes non-replicating; non-replicating; and total, the sum of intracellular and extracellular.

For a live bacterium, the program counts the named drugs to which it is resistant, and classifies it thus:

1. if it is resistant to every named drug, it is counted as MDR under every drug; in addition, if it is compensated for every named drug, it is counted as MDR double-compensated under every drug, and otherwise as MDR compensated under each drug for which it is compensated;
2. otherwise, it is counted as resistant under each drug to which it is resistant, and as compensated under each of those drugs for which it is also compensated.

A bacterium is counted in at most one of the two branches. MDR therefore means resistance to every named drug, not to a particular pair. With one drug named, every resistant bacterium is MDR and the resistant and compensated counts remain zero. With three or more drugs named, a bacterium resistant to two of them is counted as resistant, not MDR. Efflux counts follow the same pattern: a bacterium with the efflux flag set for every named drug is counted as double-efflux under every drug, and otherwise as efflux-induced under each drug whose flag is set. With one drug named, every bacterium with efflux is counted as double-efflux, and the efflux-induced counts remain zero.

Current counts are evaluated whenever drugs are named, at these points:

- in the extracellular pass, at each compartment whose container holds live bacteria when it is reached, after drug killing and the efflux update, with the container's designation after reclassification;
- at state commitment, for each living macrophage, over its live intracellular bacteria;
- when a macrophage is created, over its intracellular bacteria; of the macrophages created during a run, only a reinfecting macrophage holds bacteria (M8);
- at initialisation, for each extracellular bacterium placed from the Init section.

Bacteria dispersed during the extracellular pass into compartments already visited are not counted in that timestep, and no bacterium is counted twice.

Cumulative counts are incremented for one bacterium at each of these events:

- at every division, for the daughter, after its mutation trials (M6.13), with the designation of the dividing container, which for extracellular growth is the designation before that timestep's reclassification;
- at seeding, for each bacterium converted, at the location recorded in the candidate pool;
- at initialisation, for each extracellular bacterium placed from the Init section;
- when a macrophage is created, for each of its intracellular bacteria.

Every daughter is counted, whether its resistance arose by mutation in that division or was inherited. The cumulative counts are therefore counts of resistant bacteria produced, by division or seeding, not counts of mutation events or of distinct resistant lineages.

Single-drug kill counts. Each drug declares three cumulative counts of bacteria killed by that drug alone: intracellular, extracellular and non-replicating. No operation increments them, and they are never reset, so they are zero throughout every run.

Global statistics

Seven entries of the main statistics table belong to the module and exist in every configuration.

Statistic	Lifetime	Content
`TotDrugKill`	Cumulative	Bacteria killed by drug, all drugs and locations (M6.11)
`NumExtPreDrug`, `NumNonPreDrug`, `NumIntPreDrug`	Reset every timestep	Live bacteria in each container immediately before drug killing, summed by location
`AvgExtInteractFactor`, `AvgNonInteractFactor`, `AvgIntInteractFactor`	Reset every timestep	Interaction factor summed over bacteria evaluated, divided by the matching pre-drug count

The pre-drug counts and interaction-factor sums are updated only under the INDIGO and lookup-table methods, the extracellular and non-replicating entries being distinguished by the container's designation at the time of killing. The division is performed in the statistics block at every timestep, a zero denominator giving zero. Each average is therefore the mean, over bacteria evaluated in the timestep, of k(f)/k(1) under INDIGO, or 1 for a bacterium evaluated by the fallback (M6.10); it is zero under the lookup methods, which accumulate no factor, and under the no-interaction method, which updates neither sum nor count. In the default configuration all seven are zero throughout.

Per-drug output columns

Each per-drug object declares 86 output columns. The lung values, the intracellular macrophage plasma concentration (M6.2) and the single-drug kill counts are identically zero in every run. The per-drug columns are absent from output when no drug is named.

M6.16 Saved state and reload

A saved state records the complete state of the module, so a run with drugs can be resumed exactly. Every per-drug object is saved in full, with its parameter values as read when it was created or last reloaded, its transport data, both drug buffers, its plasma state and every statistic of M6.15. Every macrophage and every bacterium, live or dead, is saved with its concentration objects. The drug-kill object is saved with its method and, for INDIGO and the lookup methods, its tables; the interaction file or directory is not read again on load.

A state can be loaded by a regular load (`--load`) or by the automatic restart mechanism. After a state is loaded, the timestep synchronisation checks run first, followed by the module's reload processing. The loaded per-drug objects already exist when the checks run, so the drug diffusivity and permeability checks of M6.4 and M6.5 apply on every load of a state saved with drugs, using the parameter values held in the saved objects.

Load	Drugs in saved state	Drugs named on command line	Result
Regular	No	No	No drugs; nothing done
Regular	No	Yes	Warning; drugs added
Regular	Yes	Same names, same order	Parameters reloaded
Regular	Yes	Different names or order, or none	Warning if none; fatal
Restart	No	Any	Warning if any named; run continues without drugs
Restart	Yes	Any	Warning if none named; run continues with the saved drugs and parameters

Wherever drugs are present in the saved state, the saved drug-kill object is retained. If the command-line interaction method differs from the saved one, a warning is issued and the saved method continues to be used; the command-line interaction file is not consulted.

Regular load of a state saved with drugs

The command-line drug list, with white space removed, is compared with the saved list joined by commas. The comparison is order-sensitive, since order fixes the bit values (M6.1). If the lists match, these steps are performed for each per-drug object:

1. every per-drug attribute is read again from the current parameter file, and the permeability test of M6.4 is applied with the current diffusion timestep;
2. the transport data are rebuilt for the selected diffusion module: the spectral multiplier is recomputed from the reloaded diffusivity and degradation rate, or the Crank-Nicolson solver object is rebuilt;
3. every live bacterium, extracellular or intracellular, and every macrophage has its three half-saturations reset to the reloaded parameter values, and each resistant bacterium then has the reloaded fold increase applied.

Fields, plasma state, statistics and every bacterium's resistance, compensation and efflux state are retained from the saved state, and Init-section initialisation of free drug is not repeated. These quantities are not refreshed:

- the intrinsic growth rates of resistant and compensated bacteria, which keep the fitness factors in force when the cost was applied (M6.13);
- the half-saturations of dead bacteria;
- the lengths of the efflux windows and the regional concentration windows, fixed at creation from Drugs_effluxInductionStatInterval and Drugs_drugConcentrationStatInterval;
- any global `Drugs_*` parameter, which is read from the current parameter file like every other global parameter.

A change between save and load in either window-length parameter therefore leaves the arrays at their old length while the index and loop bounds follow the new value (M6.12, M6.15).

Regular load of a state saved without drugs

If drugs are named, the module is initialised at load time:

1. the per-drug objects and the drug-kill object are created as at time zero (M6.1), including the permeability test and Init-section initialisation of free drug;
2. one concentration object per drug is added to every macrophage, every live intracellular bacterium and every live extracellular bacterium;
3. the next grid is overwritten with a copy of the current grid.

Dead bacteria receive no concentration objects. No rule reads the concentration objects of a dead bacterium, so this has no effect on dynamics.

Bacteria present at the load have never undergone mutation trials under the named drugs; mutation begins at the first division after the load. If the load occurs at or after Drugs_dosageStartTime, seeding never occurs (M6.13), and dosing begins at the first timestep after the load. The first dose then falls at a timestep for which t — Drugs_dosageStartTime is a multiple of `dosageInterval`, not necessarily the first timestep after the load (M6.3).

Restart load

A restart performs none of the reload processing. The saved per-drug objects are used with their saved parameters, transport data and drug-kill object, whatever the current parameter file and command line contain. A restart therefore continues the original run exactly, provided the executable and the global parameters are unchanged.

Reload processing consumes no random draws.

M6.17 Parameters

Every parameter the module reads is listed here. Global parameters have the form `Drugs_name`. Per-drug attributes have the form `Drugs_XXX_name`, where XXX is the drug abbreviation, and are written by attribute name alone. No values are given: no drug is named in the default configuration, so no parameter of this section is consulted there, and values belong to the parameter file of a study that selects the module. Declared ranges are given in the Meaning column with their enforcement.

Declaration and validation

All declared drugs share one set of 50 attributes. Per-drug attributes are read by two independent readers, which apply different rules.

Property	General parameter loader	Per-drug reader
Drugs read	Every declared drug, whether named or not	Every named drug, declared or not

Property	General parameter loader	Per-drug reader
When	Startup, and every regular or restart load	Creation of each per-drug object; regular reload (M6.16)
Omitted attribute	Takes its declared default; an attribute with no declared default fails startup with a missing-parameter error (spine 7.0, "Resolution")	Fatal
Range check	Declared range, where enforced	None, except the permeability test of M6.4
Values used by the module	None	All

The per-drug reader requires every attribute under the exact names it reads, including `aucRiskUP`, `cmaxRiskUP`, `consumptionHalfsat` and `MIC`, and treats a missing attribute or drug node as fatal. The module reads every per-drug value it uses from the per-drug object; the general loader's values for per-drug attributes are validated but never consulted. Consequently:

- a named drug must have every attribute present in its parameter-file group, so declared per-drug defaults never take effect, including the zero defaults of `epitheliumBindingKa`, `epitheliumBindingRateConst` and `cellularKout`;
- because the general loader reads every declared drug whether or not it is named, the parameter file must supply, for every declared drug, each per-drug attribute declared without a default;
- the general loader's range check is the only range check on per-drug attributes; it applies to the file value of each declared drug, so it protects named declared drugs.

Global `Drugs_*` parameters and Mtb_Agents_perBasePairMutationRate are read only by the general loader, so their declared defaults apply.

A declared range is enforced only when its minimum and maximum differ (spine 7.0, "Validity ranges"). Of the per-drug attributes, only `PKModel`, `vesselActivity` and `partitionUsingSteadyState` carry enforced ranges, all [0, 1]; of the global parameters, only Drugs_interactThreshold does, also [0, 1]. Every other parameter of this section is declared with a lower bound of zero and no upper bound; that range is not enforced, so negative values are accepted. Because `PKModel` is range-checked by the general loader, a value outside {0, 1} for a declared drug is fatal at startup; the error at the PK step (M6.3) is reached only for an undeclared drug.

A drug absent from the declarations may be named if its parameter-file group holds every attribute. The per-drug reader then loads it and the module treats it as a declared drug, but no attribute is range-checked except permeability, so the enforced ranges of `PKModel`, `vesselActivity` and `partitionUsingSteadyState` are unprotected. Drugs_interactThreshold, a global parameter, is checked whatever drugs are named.

The per-drug reader parses each attribute as its declared type. An integer attribute, such as `dosageInterval`, whose file value is not an integer is treated as missing, which is fatal.

Parameter table

Identifier	Value or range	Units	Meaning
Drugs_dosageStartTime	—	timesteps	Dosing start: gates (M6.1), dose timing (M6.3), seeding (M6.13), caseum switch (M6.14), exposure intervals and windows (M6.15). [0, ·], not enforced[1]
Drugs_drugConcentrationStatInterval	—	timesteps	W_c: regional window length, exposure intervals, AUC risk-bound scaling (M6.15). Fixed at creation of each per-drug object. [0, ·], not enforced[2]
Drugs_effluxInductionStatInterval	—	timesteps	W: efflux window length and index (M6.12). Fixed at creation of each concentration object. [0, ·], not enforced[2]
Drugs_numberOfRandomMultiResistantBugsToAdd	—	bacteria	Multi-drug seeding (M6.13). [0, ·], not enforced; negative consumes draws without effect
Drugs_interactThreshold	—	fraction of MIC	INDIGO qualification (M6.10). [0, 1], enforced; declared default 0.1
Drugs_bloodVolume	—	L	Plasma pharmacokinetics are specified in M6.3. [0, ·], not enforced
Mtb_Agents_perBasePairMutationRate	—	per base pair per timestep	μ (M6.13). Read only when drugs are named. [0, ·], not enforced[3]
`PKModel`	—	none	0: one absorption compartment; 1: two transit compartments (M6.3). [0, 1], enforced
`kAbs`	—	per hour	Absorption from A_1, and from A_2 in model 1. [0, ·], not enforced
`q`	—	L/(h kg)	Central–peripheral exchange. [0, ·], not enforced
`volDistCentral`	—	L/kg	V_c; conversion between blood and central amount. [0, ·], not enforced[4]
`volDistPeriph`	—	L/kg	V_p. [0, ·], not enforced[4]
`clearance`	—	L/(h kg)	Elimination from the central compartment. [0, ·], not enforced
`dose`	—	mg/kg	Dose added to A_1, half per call (M6.3). [0, ·], not enforced[5]
`dosageInterval`	—	timesteps, integer	Dose timing (M6.3). [0, ·], not enforced[5]

Identifier	Value or range	Units	Meaning
`MW`	—	mg/mol	Blood conversion (M6.3), delivery (M6.4), consumption (M6.6), blood AUC and Cmax (M6.15). Must be in mg/mol for the units to close. [0, ·], not enforced[4]
`vascularPermeability`	—	cm/s	P (M6.4); bounded above by the permeability tests, about 1.667×10^{-5} cm/s under the selected module; the one value the per-drug reader checks. [0, ·], not enforced[6]
`permCoeff`	—	none	ϕ (M6.4). Zero: tissue drug only drains at sources. [0, ·], not enforced[6]
`vesselActivity`	—	none	Scaling of delivery at sources with cell density at least 0.8 (M6.4). Zero: no delivery there. [0, 1], enforced
`diffusivity`	—	cm^2/s	D: spectral multiplier, explicit and Crank-Nicolson coefficients, accuracy check on load (M6.5). [0, ·], not enforced[7]
`minDiffusivity`	—	cm^2/s	Read and stored, never used (M6.5). [0, ·], not enforced
`degRateConst`	—	none declared; applied as s^{-1}	k: degradation of free drug, folded into the multiplier, and of both bound pools (M6.5). [0, ·], not enforced[7]
`partitionUsingSteadyState`	—	none	Non-zero: steady-state mode; zero: dynamic mode (M6.7). [0, 1], enforced
`caseumUnboundFraction`	—	none	u_c (M6.7). Must lie in (0, 1] in steady-state mode and in [0, 1] in dynamic mode. [0, ·], not enforced[8]
`caseumBindingRateConst`	—	none declared; applied as s^{-1}	k_fc; dynamic mode only, but required for every named drug. [0, ·], not enforced[8]
`epitheliumBindingKa`	—	none	K_a (M6.7). Must be positive in dynamic mode. [0, ·], not enforced[8]
`epitheliumBindingRateConst`	—	none declared; applied as s^{-1}	k_fe; dynamic mode only, but required for every named drug. Zero: no epithelium binding in dynamic mode. [0, ·], not enforced[8]
`cellUptake`	—	none	a: intracellular-to-free concentration ratio (M6.7); also governs whether a macrophage raises or lowers the concentration seen (M6.9). [0, ·], not enforced[8]

Identifier	Value or range	Units	Meaning
`cellularKout`	—	none declared; applied as s^{-1}	k_out; dynamic mode only, but required for every named drug. Zero: no exchange with macrophages in dynamic mode. [0, ·], not enforced[8]
`consumptionRate`	—	M/s per bacterium	k_c (M6.6); a molar rate converted by `MW`. [0, ·], not enforced[9]
`consumptionHalfsat`	—	mg/L	H (M6.6). [0, ·], not enforced[9]
`degRateConstInt`	—	none declared; applied as s^{-1}	k_i: intracellular degradation in every macrophage (M6.6). [0, ·], not enforced[9]
`intraActivity`	—	none declared; applied per cellular timestep	E_max, intracellular (M6.9); INDIGO reference choice (M6.10). [0, ·], not enforced[10]
`extraActivity`	—	none declared; applied per cellular timestep	E_max, replicating extracellular. [0, ·], not enforced[10]
`nonrepActivity`	—	none declared; applied per cellular timestep	E_max, non-replicating extracellular. [0, ·], not enforced[10]
`c50Intra`	—	mg/L	Initial intracellular half-saturation of each bacterium (M6.2); resistance-risk test (M6.15). [0, ·], not enforced[10]
`c50Extra`	—	mg/L	Initial extracellular half-saturation; resistance-risk test. [0, ·], not enforced[10]
`c50Nonrep`	—	mg/L	Initial non-replicating half-saturation; resistance-risk test. [0, ·], not enforced[10]
`hillIntra`	—	none	h, intracellular (M6.9); efflux band (M6.12). [0, ·], not enforced[10]
`hillExtra`	—	none	h, replicating extracellular; efflux band. [0, ·], not enforced[10]
`hillNonrep`	—	none	h, non-replicating extracellular; efflux band. [0, ·], not enforced[10]
`MIC`	—	mg/L	INDIGO qualifying threshold (M6.10); counts above MIC (M6.15). Not used for the efflux band. [0, ·], not enforced[10]
`numTimeStepsEffluxInduction`	—	timesteps	N (M6.12). [0, ·], not enforced

Identifier	Value or range	Units	Meaning
`minEffluxInduction`	—	declared mg/L, fraction for RIF; applied as a dimensionless multiple	m_min (M6.12). Negative: efflux disabled. [0, ·], not enforced[11]
`maxEffluxInduction`	—	declared mg/L, fraction for RIF; applied as a dimensionless multiple	m_max (M6.12). Below m_min: band empty. [0, ·], not enforced
`concentrationChangeByEfflux`	—	fraction	ε (M6.9). [0, ·], not enforced[12]
`numberResistMutations`	—	count, real	Multiplier of p_mut for resistance (M6.13). [0, ·], not enforced
`numberCompensatoryMutations`	—	count, real	Multiplier of p_mut for compensation (M6.13). [0, ·], not enforced
`relativeFitnessResist`	—	fraction	Multiplier of both intrinsic growth rates on resistance, divisor on compensation, with the value then in force (M6.13). [0, ·], not enforced
`foldIncreaseC5OFromResistance`	—	none	Multiplier of the three half-saturations on resistance (M6.13); reapplied on reload (M6.16). [0, ·], not enforced
`numberOfRandomMonoResistantBugsToAdd`	—	bacteria	Mono-resistance seeding (M6.13). [0, ·], not enforced; negative consumes draws without effect
`resistRiskIndicatorFlag`	—	none	0: AUC; 1: Cmax; any other value: values carried from the previous drug (M6.15). [0, ·], not enforced
`aucRiskLB`	—	none declared; applied per day	Lower AUC bound, scaled by W_c/144, compared with AUC divided by the drug's half-saturation. [0, ·], not enforced
`aucRiskUP`	—	none declared; applied per day	Upper AUC bound, likewise. [0, ·], not enforced
`cmaxRiskLB`	—	none	Lower Cmax bound, unscaled, compared with Cmax divided by the drug's half-saturation. [0, ·], not enforced
`cmaxRiskUP`	—	none	Upper Cmax bound, likewise. [0, ·], not enforced
`volDistLung`	—	L/kg	Conversion for the lung window, which is always zero (M6.15). [0, ·], not enforced

The rate constants of the PK model are applied with the step $(\Delta t_d/2)/3600$ hours, so they must be per hour, as declared. The four risk bounds are compared with a ratio of exposure to half-saturation, so they are dimensionless for the Cmax measure and in hours for the AUC measure after scaling.

[1] Zero: dosing from the first timestep, and no seeding (M6.13). Negative: as for zero, with the exposure intervals offset accordingly.

[2] Zero makes the window modulus divide by zero, which terminates the run. A change between save and load leaves the arrays at their old length (M6.16).

[3] μ = 0: trials still consume draws, never succeeding. $0 < \mu < 1$: p_mut lies in [0, 1) for every g. μ = 1: certain mutation for every daughter of a parent with $g \geq 1$. $1 < \mu < 2$: p_mut alternates above and below one with the parity of g. Negative: p_mut negative; no mutation.

[4] Zero `volDistCentral`, `volDistPeriph` or `MW` divides by zero in the PK model or the recovery of blood, giving non-finite plasma values.

[5] A zero `dosageInterval` makes the dose modulus divide by zero, which terminates the run. A negative interval doses as its absolute value would. The 10^{-12} cutoff also catches negative values, so plasma state cannot stay negative between steps, and a negative `dose` is cancelled by the cutoff whenever it leaves A_1 negative.

[6] A negative `vascularPermeability` reverses the sign of the flux and passes every permeability test, since the tests bound only its magnitude from above or compare magnitudes; the clamp of M6.4 still prevents free drug at the source going negative. A negative `permCoeff` makes the flux outward at every source.

[7] Zero diffusivity gives non-numbers under the spectral module (M6.5). A negative diffusivity or degradation rate makes some or all modes of the spectral multiplier exceed one, so they grow without bound; under the explicit module a negative diffusivity reverses the sign of the diffusive update, with the same result. The accuracy check does not detect negative diffusivities, and in a run started at time zero does not see drugs at all.

[8] Steady-state mode: `caseumUnboundFraction` = 0 gives a non-number in the caseum-bound pool of every damaged compartment, undetected; above 1 gives a negative bound-to-free ratio, whose negative pool is reset to zero after the mass check, adding mass, and a denominator of zero or less gives infinite or negative results; `epitheliumBindingKa` = -1 divides by zero in unoccupied, undamaged compartments, and other negative values give a negative epithelium-bound pool reset to zero after the mass check, adding mass; a negative `cellUptake` gives a negative intracellular pool, and the denominator may vanish. Dynamic mode: `epitheliumBindingKa` zero or negative, `caseumUnboundFraction` above 1, a negative `cellUptake` or any negative binding or exit rate drives pools negative, which is fatal (M6.7).

[9] A negative `consumptionRate` makes bacteria produce drug. A zero `consumptionHalfsat` gives non-numbers in infected macrophages without drug (M6.6); a negative value lets C + H vanish. A negative `degRateConstInt` makes intracellular drug grow.

[10] Under the no-interaction and INDIGO methods: a zero activity means no killing by that drug at that location, and under INDIGO such a drug contributes nothing; a negative activity gives a negative rate and no kill, and under no interaction the combined rate is floored at zero; a zero half-saturation gives the maximal rate at any positive concentration and a non-number, so no kill, at zero concentration; a zero Hill coefficient gives E_max/2 at every concentration, including zero (M6.9); a negative Hill coefficient gives a rate decreasing with concentration and a non-number at zero; a zero or negative `MIC` makes every drug qualify at every concentration under INDIGO, and counts compartments and macrophages without drug as above MIC.

[11] The program disables efflux only for a negative value. The declared lower bound of zero would exclude that value if it were enforced. A value of 1.0 sets the lower edge of the band at the derived inhibitory concentration; it does not disable efflux.

[12] Above one, efflux increases exposure. Negative: a negative effective concentration, a rate generally not a number, and no kill.

Fixed structural constants: the delivery density threshold 0.8 (M6.4); the concentration cutoff 10^{-12} (M6.3, M6.4, M6.5, M6.7); the partitioning mass tolerance 10^{-6} mg (M6.7); the 90 per cent definition of the derived MIC, giving 9^{1/h} (M6.12); the dispersal divisor 9, one share per Moore compartment (M6.8); 600 s per cellular timestep, 6 timesteps per hour and 144 per day (M6.3, M6.15); the 1000 mol/L to mol/cm^3 conversion (M6.4); the fluoroquinolone names MXF, LVX and GFX (M6.14).

Read but without effect: `minDiffusivity`; `volDistLung`; the declared defaults of every per-drug attribute; the general loader's values of every per-drug attribute; and, for dynamics, the observation attributes `resistRiskIndicatorFlag`, `aucRiskLB`, `aucRiskUP`, `cmaxRiskLB` and `cmaxRiskUP`. Appendix B indexes them.

M6.18 Interfaces

These couplings are stated for a study that selects the module. In the default configuration none carries any value (M6.1).

With	Provides or receives	See
Spine module-slot table	Provides the M6 summary; default: not selected	M6.1
Spine Element 3	Receives the cellular timestep schedule, its order and the split half-step; the module's operations occupy fixed points of it	M6.1
Spine 2.1, 2.2, 2.3	Receives the compartment constants, the Moore ordinal convention, grid order and the one-macrophage-per-compartment rule	M6.4, M6.7, M6.8
Spine Element 5	Drug objects are created after the synchronisation checks and before any agent or extracellular bacterium is placed	M6.1
Spine 7.0 ("Resolution"; "Validity ranges")	Receives the omitted-parameter and range-enforcement rules	M6.17
Spine (random stream)	Kill draws, mutation trials and seeding shuffles draw from the single global generator, interleaved with growth and behavioural draws	M6.9, M6.13
S1	Intracellular drug moves with the macrophage	M6.8

With	Provides or receives	See
S2	Reads macrophage presence and state, the next-state branch and the live intracellular count; writes intracellular drug and intracellular bacterial deaths; dispersal of intracellular drug on death	M6.6, M6.8, M6.11
S3	The Fas-mediated and dispersing cytotoxic kill routes disperse intracellular drug through the macrophage routines they call; the clean kill destroys it	M6.8
S4	Reads container designation, live counts, generation time at division and container order after growth; writes growth rates (fitness cost and compensation), resistance and efflux state, and drug deaths added to the dead list and decayed by S4	M6.11, M6.13
S5	Reads the killing count, through the caseum level, caseation status and entrapment kills; suppresses caseum killing for named fluoroquinolones; adds nothing to the killing count	M6.7, M6.11, M6.14
S6	A separate drug sink: drug consumption acts on the drug fields only and shares no code, field or parameter with the cytokine sink	M6.6
S7	Reads the vascular source list and the 0.8 density threshold shared with oxygen delivery; drug delivery runs in the same vascular routine as oxygen, before it	M6.4
S8	Reads the source order produced by the shuffle before recruitment, used by the next timestep's delivery	M6.4
S9	Reads cell density (S9.3), the Mtb extent scaling factor σ of the previous timestep, and the cell-density extent size; provides the per-drug statistics and the seven global statistics	M6.4, M6.13, M6.15
M1	Fixes the diffusion and molecular timesteps that set the half-step, the PK step and the dynamic-partitioning substeps; performs drug diffusion with a no-flux boundary and folds degradation into the spectral multiplier, sharing the M1.3 construction; supplies the smoother and the accuracy factor	M6.5
M2	None	—
M3	The ODE stepper selected by `--odesolver` is also the stepper for the PK models and dynamic partitioning	M6.3, M6.7
M4	Independent, except that the regional granuloma split uses the threshold test, which coincides with the stored membership flag unless M4 freezes it	M6.15
M5	In three dimensions $\sigma = 1$, so each daughter receives one mutation trial; the three-dimensional delivery neighbourhood is given in M5.3; cloned three-dimensional solvers build drug plans as in M5.10	M6.4, M6.5, M6.13

With	Provides or receives	See
M7	Shares the vascular routine, running after drug delivery; neither module reads the other's state. Plasma pharmacokinetics are specified in M6.3	M6.3
M8	Under growth-rate randomisation, resistance carries no fitness cost and compensation has no effect (M8.7); a reinfecting macrophage carries concentration objects and its bacteria are counted at creation	M6.13, M6.15
M9	Neutrophils hold no drug, apply no drug killing and update no efflux state, and neutrophil-held bacteria are not seeding candidates (M9.6)	—
Appendix B	Indexes `--antibiotics-to-include`, `--drug-interaction-method`, `--drug-interaction-file` and `--dump-antibiotics`, the module's documented status, its gate rows and its inert parameters	M6.1, M6.17

Dendritic cells are a subclass of macrophage. When present, they would carry concentration objects and contribute to the intracellular and total grids and to the infected-macrophage average (M6.15). They are absent in the default configuration.