Spatial resource allocation for emerging epidemics:

A comparison of greedy, myopic, and dynamic policies

Supplementary Appendix

S1 Proofs for R_0 Computation

Proof of Lemma 1. The proof follows from the standard Perron-Frobenius theorem for non-negative matrices (e.g., see Theorem 1 in Seneta (2006)).

Proof of Theorem 1. Given a nonlinear vector-valued function ($\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$), a linear approximation near $\mathbf{x_0}$ according to Taylor's theorem is:

$$\mathbf{f}(\mathbf{x}) \approx \mathbf{f}(\mathbf{x_0}) + \mathbf{J}(\mathbf{x_0})(\mathbf{x} - \mathbf{x_0}). \tag{1}$$

The matrix $\mathbf{J}(\mathbf{x_0})$ is the Jacobian of $\mathbf{f}$ evaluated at $\mathbf{x_0}$:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & & & \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\mathbf{x}=\mathbf{x_0}} \tag{2}$$

We apply this to the system of nonlinear equations:

$$\dot{\mathbf{s}} = -diag(\mathbf{s})\mathbf{A}\mathbf{i}, \tag{3}$$

$$\dot{\mathbf{i}} = diag(\mathbf{s})\mathbf{A}\mathbf{i} - diag(\boldsymbol{\delta})\mathbf{i}, \tag{4}$$

$$\dot{\mathbf{r}} = diag(\boldsymbol{\delta})\mathbf{i} \tag{5}$$

to obtain a linear approximation near the disease-free equilibrium (DFE), $\mathbf{i} = \mathbf{0}$. The vector of nonlinear differential equations for epidemic growth can be approximated by the linear expression:

$$\dot{\mathbf{i}} \;\approx\; diag(\mathbf{s})\mathbf{A} - diag(\boldsymbol{\delta})\mathbf{i} \;\approx\; diag(\mathbf{n})\mathbf{A} - diag(\boldsymbol{\delta})\mathbf{i}. \tag{6}$$

The second approximation follows from $\mathbf{r} << \mathbf{s}$ (Assumption 2), allowing for $\mathbf{s} \approx \mathbf{n}$ near the DFE.

The matrix $\mathbf{A}$ is time-dependent with $A_{i,j,t} = m_{i,j}\beta_j\psi_{j,t}/N_j$, where the total population within each region, N_j, is constant over time, assuming there is no migration or new births. The behavior dampening function $\psi_{j,t} = 1 - \frac{\overline{\psi}_j}{1+e^{-\alpha_j t}}$ is monotonically decreasing over $t, \forall j$, implying that:

$$\lim_{t\to\infty} \psi_{j,t} = 1 - \overline{\psi}_j. \tag{7}$$

If the DFE is locally asymptotically stable, the epidemic will theoretically die out as as $t \to \infty$. We

therefore consider the modified system:

$$\dot{\mathbf{i}} \approx diag(\mathbf{n})\tilde{\mathbf{A}} - diag(\boldsymbol{\delta})\mathbf{i}. \tag{8}$$

$\tilde{\mathbf{A}}$ represents the asymptotic matrix with $\tilde{A}_{i,j} = m_{i,j}\beta_j(1-\overline{\psi}_j)/N_j$. The solution of this linear differential equation system is:

$$I_i(t) = \sum_{j=1}^{K} c_{i,j}e^{-\lambda_j t} \tag{9}$$

with constants $c_{i,j}$ and eigenvalues λ_j. The DFE is locally asymptotically stable if all the eigenvalues of $diag(\mathbf{n})\tilde{\mathbf{A}} - diag(\boldsymbol{\delta})$ have negative real parts:

$$\lambda_{max}\left(diag(\mathbf{n})\tilde{\mathbf{A}} - diag(\boldsymbol{\delta})\right) < 0 \tag{10}$$

where $\lambda_{\max}(\cdot)$ is the dominant eigenvalue of a matrix.

Similar to Theorem 2 of Van Den Driessche and Watmough (2002), we note that $\mathbf{V} = diag(\boldsymbol{\delta})$ is a non-singular matrix with positive eigenvalues (by Assumption 1) and $\mathbf{F} = diag(\mathbf{n})\tilde{\mathbf{A}}$ is a non-negative matrix, so the following two conditions are equivalent:

$$\lambda_{max}\left(\mathbf{F}-\mathbf{V}\right) < 0 \quad \Longleftrightarrow \quad \lambda_{max}\left(\mathbf{F}\mathbf{V}^{-1}\right) < 1. \tag{11}$$

Equivalently,

$$\lambda_{max}\left(diag(\mathbf{n})\tilde{\mathbf{A}} - diag(\boldsymbol{\delta})\right) < 0 \quad \Longleftrightarrow \quad \lambda_{max}\left(diag(\mathbf{n})\tilde{\mathbf{A}}\ diag(\boldsymbol{\delta}^{-1})\right) < 1. \tag{12}$$

Let $R_0 = \lambda_{max}(diag(\mathbf{n})\tilde{\mathbf{A}}\ diag(\boldsymbol{\delta}^{-1})$. Therefore, the epidemic will approach the DFE if $R_0 < 1$.

Proof of Corollary 1. Let ρ_j be the maximum of the column and row sum of the matrix $\mathbf{F}\mathbf{V}^{-1}$:

$$\rho_j = \max\left\{\sum_{i=1}^{K}\frac{m_{i,j}\beta_j(1-\overline{\psi}_j)}{\delta_j}, \sum_{j=1}^{K}\frac{m_{i,j}\beta_j(1-\overline{\psi}_j)}{\delta_j}\right\}. \tag{13}$$

By the standard Perron-Frobenius theorem for non-negative matrices, the dominant eigenvalue λ_{max} of the matrix $\mathbf{F}\mathbf{V}^{-1}$ has the following bounds:

$$\min_j \sum_{i=1}^{K}\frac{m_{i,j}\beta_j(1-\overline{\psi}_j)}{\delta_j} \le \lambda_{max} \le \max_j \sum_{i=1}^{K}\frac{m_{i,j}\beta_j(1-\overline{\psi}_j)}{\delta_j}, \tag{14}$$

$$\min_i \sum_{j=1}^{K}\frac{m_{i,j}\beta_j(1-\overline{\psi}_j)}{\delta_j} \le \lambda_{max} \le \max_i \sum_{j=1}^{K}\frac{m_{i,j}\beta_j(1-\overline{\psi}_j)}{\delta_j}. \tag{15}$$

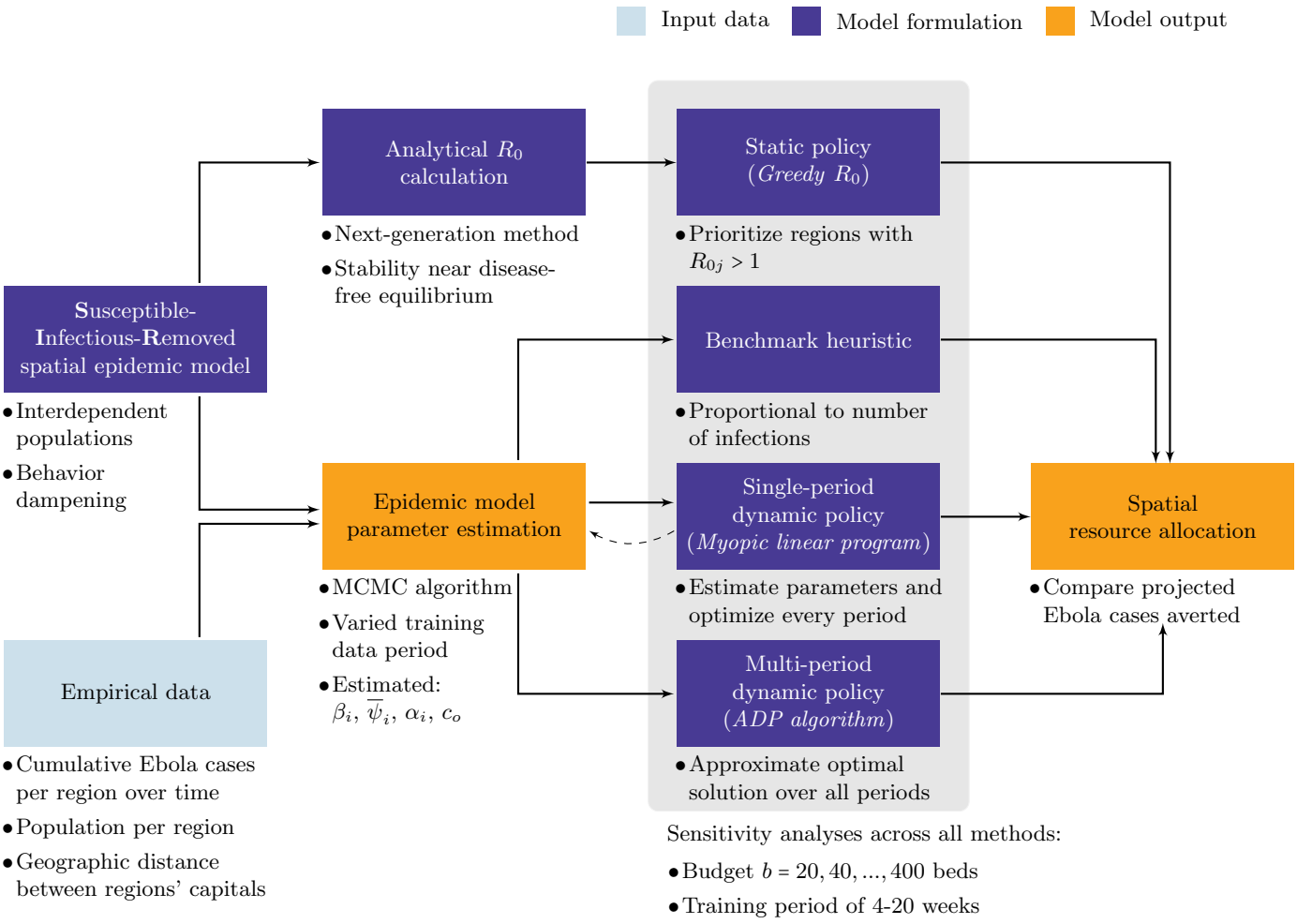
Using the upper bounds of Equations (14)–(15), we verify that:

$$\lambda_{max} \le \max_j\left\{\rho_j\right\}, \tag{16}$$

and by Lemma 1:

$$R_0 \le \max_j\left\{\rho_j\right\}. \tag{17}$$

Figure S1: Overview of the relationships between input data, model elements, and model results.



S2 Epidemic Model Calibration

The epidemic model was calibrated with confirmed Ebola case data from 21 regions in Guinea, Liberia, and Sierra Leone, which collectively accounted for 60% of the total infections observed (Humanitarian Data Exchange, 2015). Regions with no change in cases, or with fewer than 50 cases or five data points, were omitted. We also excluded 61 observations (amounting to less than 2.5% of the total dataset) that could clearly be identified as outliers as well as two regions with inconsistent case numbers. These particular observations were likely the result of misreporting. We used 20 weeks of available data; of these, the first 16 weeks (beginning on 2 September 2014) were used for fitting purposes and the next 4 weeks were used to test the model's prediction against the outbreak's actual trajectory. As a robustness check, we evaluated the model's performance using daily case count data over 4, 8, 12, 16, and 20 16 weeks.

Model calibration was performed using a Markov chain Monte Carlo approach with a Metropolis–Hastings algorithm, a method for fitting epidemic models to empirical data (Gibson, 1998; Currie, 2007). The model was simultaneously calibrated for all K regions in our dataset assuming that the epidemic dynamics described in equations (3)–(5) applied to each region. The model time horizon was divided into discrete intervals: $t \in (t_0, \ldots, t_f)$, corresponding to the daily collected data points.

The vector $\mathbf{w}_n$ contains the parameter c_0 and the parameters β_j, $\overline{\psi}_j$, and α_j, for $i = 1, 2, \ldots, K$, at iteration n of the MCMC algorithm. Since the value of each parameter is unknown, we assume Uniform *prior* distributions over the following intervals: $\beta_j \in (0, 1)$, $\overline{\psi}_j \in (0.5, 1)$, $\alpha_j \in (0, 1)$, and $c_0 \in (0.99, 1)$. We compute the sum of squared errors (SSE) between observed cumulative cases ($o_{i,t}$) and expected cumulative cases ($e_{i,t}(\mathbf{w}_n)$) in region i at time t based on model projections, where $e_{i,t} = I_{i,t} + R_{i,t}$.

$$\text{SSE} = \sum_{i=1}^{K} \sum_{t=t_0}^{t_f} \big(o_{i,t} - e_{i,t}(\mathbf{w}_n)\big)^2. \tag{18}$$

The Metropolis–Hastings algorithm recursively calculates a new parameter candidate set $\mathbf{w}_n$ at iteration n via maximum likelihood with the following likelihood ratio:

$$\text{LR} = \frac{P(\mathbf{o}|\mathbf{w}_n)}{P(\mathbf{o}|\mathbf{w}_{n-1})} = \exp\left(- \sum_{i=1}^{K} \sum_{t=t_0}^{t_f} \big(o_{i,t} - e_{i,t}(\mathbf{w}_n)\big)^2 + \sum_{i=1}^{K} \sum_{t=t_0}^{t_f} \big(o_{i,t} - e_{i,t}(\mathbf{w}_{n-1})\big)\right). \tag{19}$$

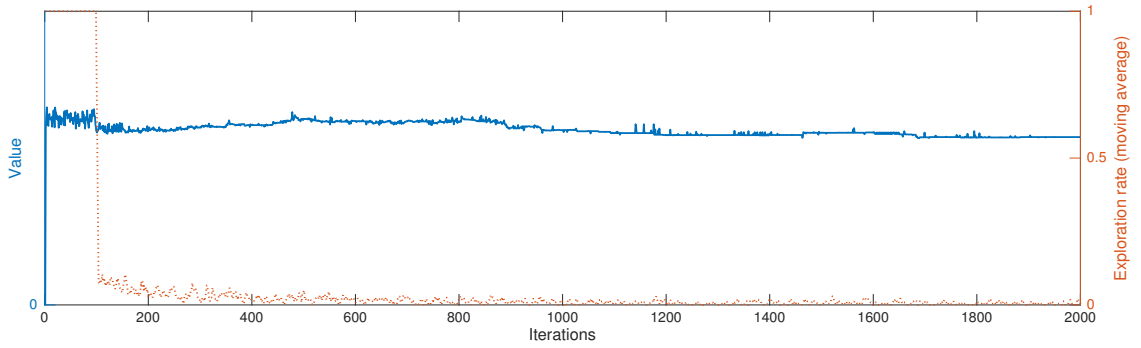
where the vector $\mathbf{o} = [o_{i,t}]$, for $i = 1, \ldots, K, t = t_0, \ldots, t_f$ represents accumulated observed cases, by region. At iteration n, the candidate parameter set $\mathbf{w}_n$ replaces the previous set $\mathbf{w}_{n-1}$ if LR $> \nu$, with $\nu \sim \text{Uniform}(0, 1)$. The algorithm continues until convergence is obtained for all parameter values (at most 150,000 iterations in our experiments).

S3 ADP Algorithm

Our approximate dynamic programming (ADP) solution approach consists of four components (pseudocode is given in Figure S3). An exploration / exploitation rule determines if the algorithm explores the value of a random state or exploits the available information on state value approximation (S3.1). The value of a state is approximated using linear basis functions with features extracted from a state (S3.2). The policy iteration algorithm (line 13) is reformulated into multiple Knapsack sub-problems within each period (S3.3). At the end of each iteration, the basis functions are updated using a recursive-least-squares approach (S3.4).

The ADP algorithm consists of one outer loop and two nested loops. The first nested loop computes an optimal allocation decision, based on the current approximate value function using a

Figure S2: ADP algorithm convergence (solid blue) and exploration rate (dotted red) with a budget of $B = 150$ beds, training period of 8 weeks, and maximum deployment rate of $r = 5$ beds per week per region.



nonlinear Knapsack approach. The second nested loop updates the basis function parameter vector ($\boldsymbol{\theta}^n$) with the observed state value. The outer loop repeats the two inner loops for N iterations, where N is sufficiently large to reach convergence of $\boldsymbol{\theta}$ in all tested cases. The optimal policy can then be computed using $\boldsymbol{\theta}^N$ within the value function approximation.

S3.1 Exploration versus Exploitation

One limitation of ADP is that an allocation decision (in the first loop of the algorithm) might visit only the states with a high approximated value. Previously unvisited, low-value states would be ignored even though their value approximation might change once visited. To address this problem, we explore the state space also by making random allocation decisions. Although exploration increases the likelihood of a near-optimal solution, it also reduces the speed of convergence—a trade-off well known as the "exploration–exploitation dilemma".

Because the exploration of our problem's state space is relatively inexpensive, we choose a combination of pure and ϵ-greedy exploration, which yielded good results in our experiments. During the first 100 iterations, the algorithm explores through random allocation decisions. Thereafter, during the nth iteration, if $e < 1/0.1n$ for $e \sim \text{Uniform}(0, 1)$, a random decision is chosen; otherwise, the algorithm exploits the current value function approximation (Figure S2). This procedure assures a initially high degree of exploration which decreases as the algorithm progresses.

S3.2 Feature Extraction

To efficiently approximate the future value of a state x_t, without sweeping over the entire state space, we use a basis function approach (Powell, 2011). A basis function $\phi_f(x_t)$ summarizes just a few state features $f \in \mathcal{F}$, thereby reducing the large set of state variables from the original dynamic

Table S1: Basis functions and associated R^2 and Akaike information criterion (AIC) averages for all populations.

Basis function ($\forall i \in 1, \ldots, K$)	# of variables	R^2	AIC ($\times 10^6$)
$S_i I_i$	K	.93	-5.89
$I_i + S_i \sum_{j=1}^{K} I_j$	$2K$	.95	-5.69
$S_i \sum_{j=1}^{K} I_j + B$	$K+1$	.14	71.78
$\frac{I_i}{S_i} + B$	$K+1$	.98	-7.18
ρ_i	K	.07	20.61
$\rho_i S_i I_i + B$	$K+1$	.98	-7.16

program. For our model, we examine the following set of features:

$$\mathcal{F} = \left\{ S_i,\ I_i,\ R_0,\ S_i I_i,\ \frac{I_i}{S_i},\ \sum_{j=1}^{K} I_j,\ S_i \sum_{j=1}^{K} I_j,\ \sum_{j=1}^{K} \frac{I_j}{S_i},\ \rho_i S_i I_i,\ B \quad \forall i \right\}. \tag{20}$$

Selecting appropriate basis functions is essential for a good approximation of the value function because doing so extracts those elements from the state space that have high explanatory power (e.g., the number of infected individuals in a population affects the future epidemic trajectory in that population). We rank potential basis functions in terms of their explanatory power for a state's future value (Hulshof et al., 2015). We simulate epidemic trajectory paths under different resource allocation policies, and then treat these simulated values as observations in a linear regression to determine which set of features best approximates the value function (Table S1).

Let $\boldsymbol{\phi}(x_t)$ denote the column vector of features, and $\boldsymbol{\theta}_t$ denote the parameter vector of estimated regression coefficients. The value function approximation at iteration n of the algorithm is:

$$\bar{V}_t^n(x_t|\boldsymbol{\theta}_t^n) = \boldsymbol{\phi}(x_t)^T \boldsymbol{\theta}_t^n. \tag{21}$$

To ensure that the algorithm is not only visiting states with an already high approximated value, we use ϵ-greedy exploration throughout our experiments.

S3.3 Knapsack Formulation

The resource allocation step of the ADP algorithm can be performed only for problems with small action spaces, since sweeps over the whole action space are computationally expensive. Although similar to a Knapsack problem, most solution methods cannot be applied because the objective function—minimizing Ebola cases in all future periods—is neither convex nor concave (Kellerer et al., 2004). We therefore divide each allocation problem into r subproblems, where r is the

maximum number of additional beds that can be allocated per region per period. In each of the resulting subperiods, indexed by τ, either 0 or 1 beds are allocated to each region. This 0/1 nonlinear Knapsack problem can be solved with a method introduced by Hochbaum (1995), which we briefly explain as follows.

For each subperiod $\tau \in (0, 1, \ldots, r)$, the benefit $c_{k,\tau}$ of allocating $a_{k,\tau} \in \{0, 1\}$ beds to region k, assuming no other regions receive beds ($a_i = 0, \ \forall i \neq k$), is approximately:

$$c_{k,\tau} = \sum_{i=1}^{K} S_{i,\tau} m_{i,k} \beta_k \psi_{k,\tau} \frac{\xi a_{k,\tau}}{N_k}. \tag{22}$$

The 0/1 Knapsack subproblem in period τ is thus:

$$\text{maximize} \quad \sum_{k=1}^{K} c_{k,\tau} a_{k,\tau} \tag{23}$$

$$\text{subject to} \quad \sum_{k=1}^{K} a_{k,\tau} \leq 1, \tag{24}$$

$$a_{k,\tau} \in \{0, 1\} \quad \forall k. \tag{25}$$

After transforming the original problem into a nonlinear Knapsack problem, we can solve large-scale resource allocation problems in a reasonable amount of time even though the underlying model is characterized by dynamics that are nonlinear and nonconvex.

S3.4 Recursive least squares updating

In line 20 of the ADP algorithm, the parameter vector $\boldsymbol{\theta}_t^n$ is updated using a recursive least-squares approach (Powell, 2011) that is based on the previous iteration's estimated coefficients $\boldsymbol{\theta}_t^{n-1}$:

$$\boldsymbol{\theta}_t^n = \boldsymbol{\theta}_t^{n-1} - \frac{1}{\gamma^n} G^{n-1} \boldsymbol{\phi}^n \Delta^n. \tag{26}$$

The $2K \times 2K$ matrix G^n is updated recursively using the expression

$$G^n = \frac{1}{\lambda^n} \left(G^{n-1} - \frac{1}{\gamma^n} (G^{n-1} \boldsymbol{\phi}^n (\boldsymbol{\phi}^n)^T G^{n-1}) \right), \tag{27}$$

and γ^n is a scalar computed using

$$\gamma^n = \lambda^n + (\boldsymbol{\phi}^n)^T G^{n-1} \boldsymbol{\phi}^n. \tag{28}$$

The difference Δ^n between the previous iteration's value approximation for state x_t and its corresponding "observed" (i.e., simulated) value at iteration n is:

$$\Delta^n = \bar{V}_t^{n-1}(x_t) - \hat{V}_t^n(x_t). \tag{29}$$

Figure S3: Approximate dynamic programming (AD) algorithm.

begin
  Set basis function ϕ_f;
  Set $\boldsymbol{\theta}_t^0$ for all t;
  Set starting state $x_0 = (\mathbf{S}_{t_0},\ \mathbf{I}_{t_0})$;
for $n = 1, 2, \ldots, N$ **do**
  Draw randomly $e \sim \text{Uniform}(0, 1)$;
  for $t = t_0, t_0 + 1, \ldots, t_f$ **do**
    if $\sum_{i=1}^{K} \sum_{t_0}^{t} a_{i,t}^n \geq B$ **then**
      $a_{i,t}^n = 0\ \forall i = 1, \ldots, K$;
    else if $(n < 200) \vee (e < 1/0.1n)$ **then**
      Randomly select $a_{i,t}^n \in \mathcal{A}_t$;
    else
      Compute
      $$a_{i,t}^n = \underset{a_{i,t} \in \mathcal{A}_t}{\text{argmin}} \Big(\sum_{i=1}^{K} S_{i,t} \sum_{j=1}^{K} m_{i,j} \beta_j \psi_{j,t} \frac{(I_{j,t} - \xi a_{j,t}^n)}{N_j} + \gamma \boldsymbol{\theta}_t^{n-1} \boldsymbol{\phi}(x_{t+1}^n \mid x_t^n, a_{i,t})^T \Big);$$
    end
    Compute state transition $x_t(a_{i,t}^n) \rightarrow x_{t+1}$
  end
  Initialize $\hat{V}_{t_f}^n = 0$;
  for $t = t_f, t_f - 1, \ldots, t_0$ **do**
    $$\hat{V}_t^n = \left(\sum_{i=1}^{K} S_{i,t} \sum_{j=1}^{K} m_{i,j} \beta_j \psi_{j,t} \frac{(I_{j,t} - \xi a_{j,t}^n)}{N_j} + \gamma \hat{V}_{t+1}^n \right);$$
    $\boldsymbol{\theta}_t^n = \boldsymbol{\theta}_t^{n-1} - \frac{1}{\gamma^n} G^{n-1} \boldsymbol{\phi}^n \Delta^n$;
  end
end
Return basis function parameters $\boldsymbol{\theta}_t^N$ for all t.

References

Currie, CSM. 2007. Bayesian methodology for dynamic modelling. *Journal of Simulation* **1**(2) 97–107.

Gibson, G. 1998. Estimating parameters in stochastic compartmental models using Markov chain methods. *Mathematical Medicine and Biology* **15**(1) 19–40.

Hochbaum, DS. 1995. A nonlinear knapsack problem. *Operations Research Letters* **17**(3) 103–110.

Hulshof, PJH, MRK Mes, RJ Boucherie, EW Hans. 2015. Patient admission planning using approximate dynamic programming. *Flexible Services and Manufacturing Journal* 1–32.

Humanitarian Data Exchange. 2015. Sub-national time series data on Ebola cases and deaths in Guinea, Liberia, Sierra Leone, Nigeria, Senegal and Mali since March 2014. URL `https://data.humdata.org/dataset/rowca-ebola-cases`.

Kellerer, H, U Pferschy, D Pisinger. 2004. *Knapsack Problems*. Springer, Berlin and New York.

Powell, WB. 2011. *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. 2nd ed. John Wiley & Sons.

Seneta, E. 2006. *Non-negative Matrices and Markov Chains*. Springer Series in Statistics.

Van Den Driessche, P, J Watmough. 2002. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Mathematical Biosciences* **180** 29–48.