Implementation and acceleration of optimal control for systems biology

Jesse A Sharp[1,2] *, Kevin Burrage[1,2,3], Matthew J Simpson[1].

[1] *School of Mathematical Sciences, Queensland University of Technology (QUT), Australia.*

[2] *ARC Centre of Excellence for Mathematical and Statistical Frontiers, QUT, Australia.*

[3] *Department of Computer Science, University of Oxford, UK (Visiting Professor).*

Abstract

Optimal control theory provides insight into complex resource allocation decisions. The forward-backward sweep method (FBSM) is an iterative technique commonly implemented to solve two-point boundary value problems (TPBVPs) arising from the application of Pontryagin's Maximum Principle (PMP) in optimal control. In this review we discuss the PMP approach to optimal control and the implementation of the FBSM. By conceptualising the FBSM as a fixed point iteration process, we leverage and adapt existing acceleration techniques to improve its rate of convergence. We show that convergence improvement is attainable without prohibitively costly tuning of the acceleration techniques. Further, we demonstrate that these methods can induce convergence where the underlying FBSM fails to converge. All code used in this work to implement the FBSM and acceleration techniques is available on GitHub at https://github.com/Jesse-Sharp/Sharp2021.

* Corresponding author
Email address: `jesse.sharp@hdr.qut.edu.au`, *Telephone* `+ 61 7 31385241` *Fax* `+ 61 7 3138 2310` (Jesse A Sharp[1,2]).

1 Introduction

Across the life sciences, we encounter systems over which we wish to exert control. Whether we consider outbreak control in epidemiology [1,51], chemotherapy in oncology [8,18,70], muscle contraction and gait regulation in biomechanics [29,42,58], engineering cellular processes in synthetic biology [25,36], or biodiversity and invasive species management in ecology [9,19,21], we face decisions around how a particular intervention should be applied to best achieve desired outcomes. Using mathematical models of such systems, optimal control theory provides insight into these resource allocation decisions.

Optimal control is a science of trade-offs; between competing objectives, or in weighing up the benefits of control measures against their costs. We illustrate some key concepts of optimal control in Figure 1. Suppose that without intervention, a crop yield will double, from x_0 to $2x_0$, between now and harvest time. We might consider applying a control, such as fertiliser, to increase the growth rate of the crop; thereby increasing the yield at harvest to $3x_0$. Of course, applying fertiliser comes at a cost, and this must be considered against the increase in crop yield. As such, it is not immediately apparent how much fertiliser should be applied, and over what time period. This depends entirely on our characterisation of optimality; the *pay-off*. Depending on the pay-off, the optimal control may be continuous; whereby the strength can be readily and continuously adjusted throughout time, or bang-bang (discontinuous); whereby the control is applied at either a lower or upper bound with finitely many discrete switches between the two. The pay-off determines the objective(s) of control; which in our stylised example may be to maximise profits after cost of fertilising is considered, or achieve a specific yield, for example $3x_0$, using the minimum amount of fertiliser.

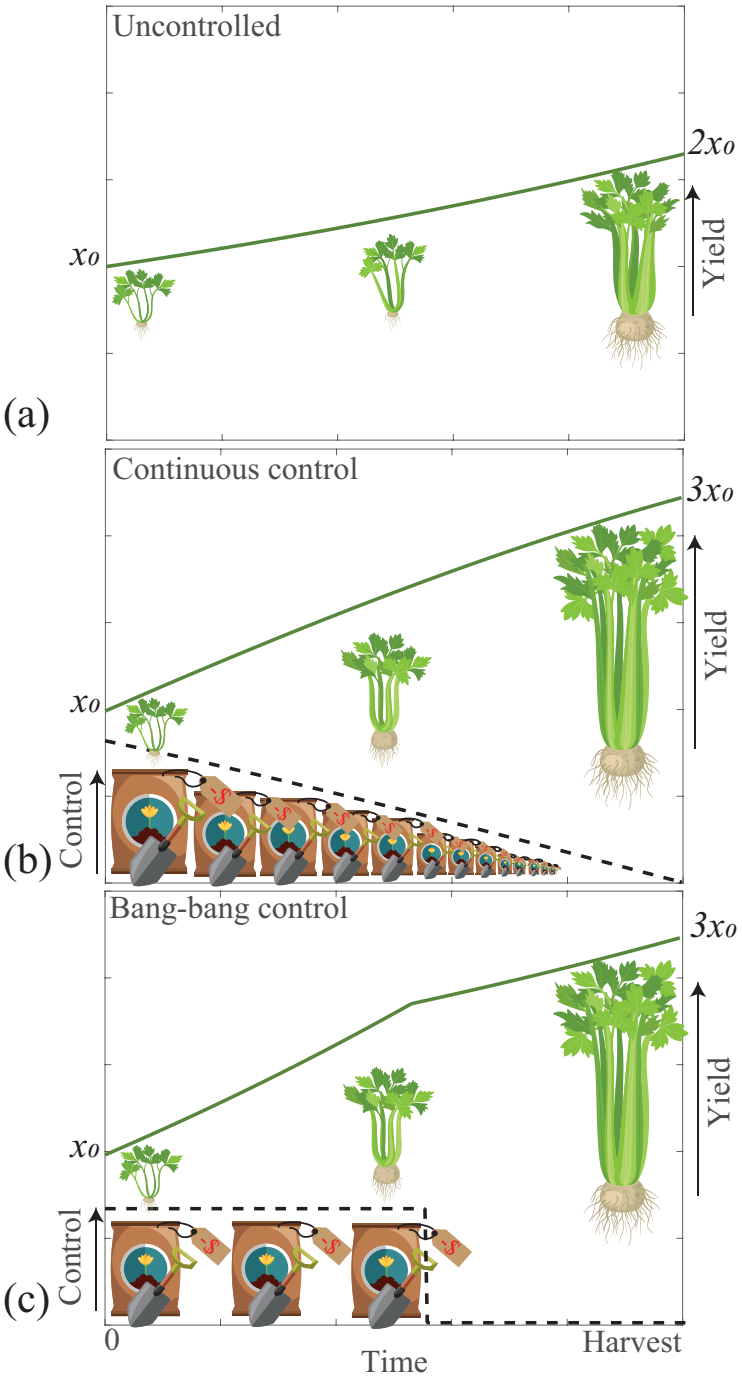


Fig. 1. A pictorial example of optimal control for a growing crop. Suppose that initially, the crop yield is x_0. We want to grow this crop to increase the yield, represented by the green line, come harvest time. Actions taken to increase the growth rate of the crop; such as applying fertiliser, are the controls, represented in black dash. Scenarios are presented for (a) no control, (b) continuous control, and (c) bang-bang control. Optimal control theory helps us determine how best to apply these controls. Illustrations adapted from ilyakalinin/iStock/Getty Images, johavel/iStock/Getty Images.

Much of modern day optimal control theory stems from the seminal works of Pontryagin; through the Pontryagin Maximum Principle (PMP) [61], and Bellman; through the advent of dynamic programming and the Hamilton-Jacobi-Bellman equation [11], in the 1950s and 1960s. These foundations of optimal control are built upon centuries of development in the calculus of variations [31]. For brief but broad expositions of the theoretical roots of optimal control and developments following these seminal works, we direct the reader to articles such as [16,68].

Often we are unable to solve optimal control problems analytically, so we pursue computational approaches. Broadly, the numerical methods for optimal control can be classed as either indirect or direct methods; for indirect methods optimality conditions are derived in the calculus of variations fashion via the PMP, leading to a two-point boundary value problem (TPBVP), while for direct methods the control problem is discretised and reformulated as a nonlinear programming problem [64]. For an early history of numerical methods in optimal control, including gradient and conjugate gradient methods, Newton-Raphson methods, quasilinearisation, feasible direction algorithms and feedback solutions we suggest [60]. Surveys [64,68] give an excellent overview of more recent developments in relevant numerical methods, including the forward-backward sweep method (FBSM), multiple-shooting methods, control parameterisation, collocation and pseudospectral methods and complete discretisation into finite-dimensional nonlinear programming problems.

The FBSM is an iterative method for solving the TPBVPs that arise from the indirect PMP approach to optimal control. FBSM has achieved popularity in optimal control, owing particularly to its straightforward scalability to large systems, and to its moderate computational cost and mathematical complexity [43]. In this work we review the implementation of the FBSM to solve optimal control problems, and investigate means of accelerating the convergence. To contextualise our discussion of the FBSM, we first consider the more familiar

technique of successive over-relaxation (SOR). SOR is a generalisation of the Gauss–Seidel method, and is widely applied in numerical linear algebra to accelerate convergence when solving linear systems iteratively [76]. Essentially, the process of SOR involves specifying an acceleration or relaxation parameter, $\beta \in (0, 2)$; a weighting factor that serves to reduce the spectral radius of the iterative matrix operator [66]. The error and rate of convergence of SOR is sensitive to this (problem dependent) choice of β, prompting investigation into theoretical convergence results and methods of determining β [22,41,66]. Despite challenges in identifying the optimal β, the SOR has historically been widely applied and studied in the literature due to the ease with which it can be implemented, and the rapid convergence it can deliver; even without identifying the optimal β [35,79].

This narrative closely parallels that of the FBSM in optimal control, where a weighting factor ω can be applied when updating the control between iterations to aid convergence. The optimal choice of ω is problem dependent, and significantly impacts the rate of convergence, or whether the FBSM converges at all. Nonetheless, the FBSM is frequently used in applied optimal control work as it is relatively straight-forward to implement, and can still converge in absence of the optimal ω. Theoretical convergence results of the FBSM are available in the literature [34,49], although the focus is on the FBSM without weighted updating, with no consideration for choosing ω. Using regularisation techniques, the FBSM is modified in [44] to improve convergence properties for large systems in a continuous setting, with a view to training deep neural networks in machine learning. These convergence results have recently been extended to the numerically discretised setting through symplectic Runge-Kutta discretisation; taking advantage of the variational structure of optimal control problems [45]. The authors also demonstrate that the rate of convergence of the regularised FBSM with symplectic discretisation can be improved with Anderson acceleration, an iterative acceleration technique. Although promis-

ing, this regularisation introduces a regularisation parameter, ρ. Similar to ω, the choice of ρ impacts convergence, and its choice is problem dependent. Understanding and implementing the regularisation and symplectic techniques is not trivial, and introduces conceptual complexity beyond what is necessary for many applied optimal control problems. As such, the standard FBSM remains an attractive choice for practitioners.

To this end, we aim to review acceleration techniques that can be paired with the standard FBSM. We implement such techniques alongside the FBSM with the goals of: (1) increasing the rate and frequency of convergence, and (2) reducing the importance of, and challenges associated with, selecting ω. A graphical overview of the optimal control process we employ in this work, including the incorporation of acceleration methods, is presented in Figure 2. We note that all code used to implement the algorithms presented in this review; the FBSM and the Wegstein, Aitken-Steffensen and Anderson acceleration methods, is available on GitHub.

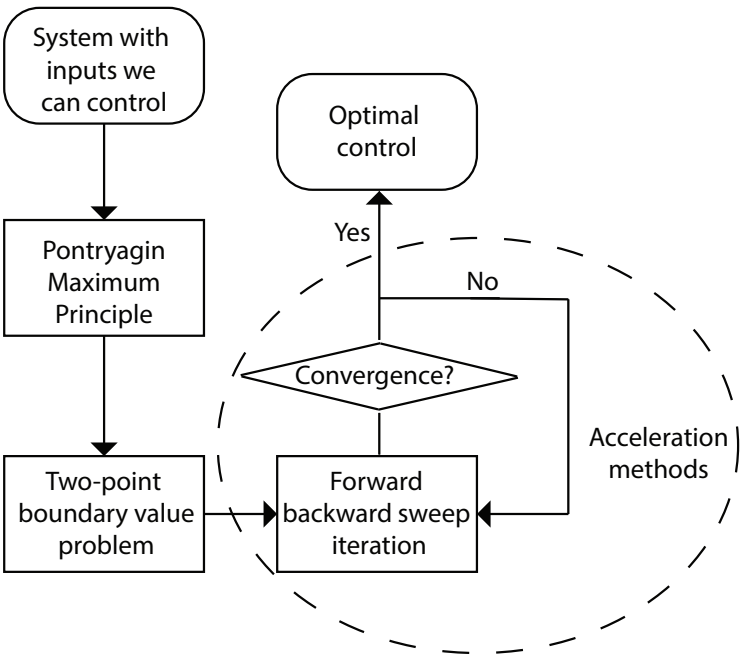


Fig. 2. The process of optimal control via the Pontryagin Maximum Principle approach, with the incorporation of acceleration methods.

Throughout this work we consider optimal control in the systems biology context. However, we note that optimal control is relevant to a wide variety of fields including chemical engineering [46], aeronautics and astronautics [64], management science and economics [69]. The FBSM, and by extension, the acceleration techniques we consider in this work, can be readily applied in any of these areas.

In §2 we review the PMP approach to optimal control, and the implementation of the FBSM. We provide a single-variable linear model, and a multi-variable nonlinear model in §3; and pose and solve example continuous, bang-bang (discontinuous), and fixed endpoint control problems. We review potential iterative acceleration methods in §4, and present the results of selected techniques in §5. We discuss the performance of these techniques in §6, and identify opportunities for application and further investigation.

2 Forward-backward sweep method

In an optimal control problem with one state variable, $x(t)$, one control, $u(t)$, over a fixed time interval, $t \in [t_0, t_N]$; such as the crop growth example presented in Figure 1, we seek the optimal control $u^*(t)$ that minimises or maximises a specified pay-off function, J, subject to the dynamics of the state. In this section we briefly review the Pontryagin Maximum Principle approach to such an optimal control problem, and the standard implementation of the FBSM for solving the resulting two-point boundary value problem. The FBSM is readily extended to problems with multiple state variables, multiple controls, state constraints and free end-times [43,49,70,71], however for this overview we restrict ourselves to the single variable, single control, fixed end-time case for clarity.

The pay-off typically comprises a cost function $\mathcal{L}(t, x(t), u(t))$ integrated over

the time interval, and/or a function of the final state, $x(t_N)$. As such, we seek to minimise or maximise J, subject to

$$J = \boldsymbol{\phi}(x(t_N)) + \int_{t_0}^{t_N} \mathcal{L}(t, x(t), u(t))\ \mathrm{d}t. \tag{1}$$

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(x(t), u(t), t), \qquad x(t_0) = x_0. \tag{2}$$

Applying the PMP, we construct the Hamiltonian; $H(t, x(t), u(t), \lambda(t)) = \mathcal{L}(t, x(t), u(t)) + \lambda(t) f(x(t), u(t), t)$, where $\lambda(t)$ is the co-state variable linking our state to our pay-off. The necessary conditions for optimal control are obtained from the Hamiltonian:

(1) The optimal control, $u^*(t)$, is obtained by minimising the Hamiltonian,
$$\frac{\partial H}{\partial u} = 0. \tag{3}$$

(2) The co-state is found by setting,
$$\frac{\mathrm{d}\lambda}{\mathrm{d}t} = -\frac{\partial H}{\partial x},$$

(3) satisfying the transversality condition,
$$\lambda(t_N) = \lambda_N = \left.\frac{\partial \phi}{\partial x}\right|_{t=t_N}. \tag{4}$$

Following these steps yields a TPBVP to solve for $x(t)$, $\lambda(t)$, subject to $x(t_0) = x_0$, and $\lambda(t_N) = \lambda_N$. To solve this numerically, we discretise t into $N+1$ time points separated by a step-size $\mathrm{d}t = (t_N - t_0)/N$; $\mathbf{t} = [t_0, t_0 + \mathrm{d}t, ..., t_0 + N\mathrm{d}t] = [t_0, t_1, ..., t_N]$. Here, we consider a uniform discretisation in time; although this is not strictly necessary, as discussed in §3. Using superscripts to denote the iteration number, provide an initial guess of the control

at each t; $\mathbf{u}^{(0)} = [u_0^{(0)}, u_1^{(0)}, ..., u_N^{(0)}]$. From $\mathbf{u}^{(0)}$, solve Equation (2) numerically from t_0 to t_N to obtain $\mathbf{x}^{(0)} = [x_0^{(0)}, x_1^{(0)}, ..., x_N^{(0)}]$. Now, using $\mathbf{x}^{(0)}$, solve for $\boldsymbol{\lambda}^{(0)} = [\lambda_0^{(0)}, \lambda_1^{(0)}, ..., \lambda_N^{(0)}]$ backwards in time from t_N to t_0, starting from λ_N. With the optimality condition from Equation (3), generate a temporary update for the control, $\hat{\mathbf{u}}^{(1)}$. The next iteration begins with an updated guess for the control, $\mathbf{u}^{(1)}$. These steps are repeated until a convergence condition is satisfied. The algorithm for the FBSM is summarised in §1 of the supplementary material.

In some instances, directly updating the control, such that

$$\mathbf{u}^{(k)} = \hat{\mathbf{u}}^{(k)}, \quad k = 1, 2, ... \tag{5}$$

is sufficient, however more commonly a weighted update is performed [43,70], such that in the $(k+1)$th iteration,

$$\mathbf{u}^{(k+1)} = \omega \mathbf{u}^{(k)} + (1-\omega)\hat{\mathbf{u}}^{(k+1)}, \quad k = 1, 2, ..., \quad \omega \in [0, 1). \tag{6}$$

This weighted updating is also referred to as applying a relaxation factor, similar to SOR as discussed in §1. An appropriate choice of ω in Equation (6) can accelerate convergence relative to Equation (5), or in some cases induce convergence where Equation (5) leads to divergence. The weighting parameter, ω, can be held constant between iterations, although faster convergence may by achieved by updating ω. For example, by reducing ω as the system approaches convergence, a greater portion of the updated control is maintained relative to the control from the previous iteration [43], possibly accelerating convergence. A challenge commonly faced in implementing this control up-

dating scheme is that the best choice for ω is problem dependent, and often is determined heuristically in practice. We address the extent to which the proposed acceleration algorithms address this issue in §4.

To facilitate the following discussion regarding acceleration, we note that the FBSM can be thought of as a generalised fixed point iteration [49], where each iteration comprises a forward and backward sweep and a control update. As such, for a control problem with one control, discretised into $N+1$ time points, each iteration of the FBSM can be thought of as the application of a nonlinear operator, $\mathcal{F}$, of dimension $N+1$, such that $\mathbf{u}^{(k+1)} = \mathcal{F}(\mathbf{u}^{(k)})$, or:

$$\begin{bmatrix} u_0^{(k+1)} \\ u_1^{(k+1)} \\ \vdots \\ u_N^{(k+1)} \end{bmatrix} = \begin{bmatrix} f_0(u_0^{(k)}, u_1^{(k)}, ..., u_N^{(k)}) \\ f_1(u_0^{(k)}, u_1^{(k)}, ..., u_N^{(k)}) \\ \vdots \\ f_N(u_0^{(k)}, u_1^{(k)}, ..., u_N^{(k)}) \end{bmatrix},$$

where $\mathcal{F} = (f_0, f_1 ... f_N)^{\mathrm{T}}$. However, in general we are not able to write down an explicit expression for $\mathcal{F}$. Conceptualising the FBSM as a fixed point iteration process informs the choice of acceleration methods discussed in §4.

Importantly, we use the term *function evaluation* in this work to refer to the process of solving the system of ODEs for the state forward in time and the system of ODEs for the co-state backwards in time, once. This aligns with a single iteration of the standard FBSM. The function evaluation nomenclature becomes convenient when discussing the FBSM in the context of acceleration algorithms, that typically focus on reducing the number of times expensive functions are evaluated. Producing numerical solutions to the ODE systems

is by far the most computationally expensive component of the FBSM. This computational expense increases with the size and complexity of the systems; reducing the number of times these systems must be solved becomes more advantageous as the size and complexity of the systems increases. The function evaluation description also facilitates comparison between acceleration methods that require solving the ODE systems a different number of times per iteration. Throughout this work, we use $\mathcal{N}$ to denote the total number of function evaluations a given method takes to achieve convergence.

2.1 Adapted forward-backward sweep method

The FBSM can be extended to handle problems where we aim not only to minimise or maximise a given quantity over time, but also ensure that a specific state is reached at final time. This aligns with the crop growth example from Figure 1 if the objective is to achieve a specific yield of $3x_0$ at harvest, rather than to maximise yield. In this case we may have an integral term in the pay-off as described in Equation (1), however the function of the final state, $\boldsymbol{\phi}(x(t_N))$, is redundant in a control problem with a prescribed final state. Equation (2) is also modified to incorporate the additional constraint:

$$J = \int_{t_0}^{t_N} \mathcal{L}(t, x(t), u(t))\ \mathrm{d}t, \quad \text{subject to,}$$

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(x(t), u(t), t), \qquad x(t_0) = x_0, \qquad x(t_N) = x_N. \tag{7}$$

Here, x_N is the specified state that must be reached at final time. Since we have introduced an additional boundary value to the system, we no longer obtain the transversality condition from Equation (4). Instead, we seek the final time

condition on the co-state, λ_N, and associated optimal control that satisfies Equation (7). We proceed by considering an adapted FBSM that takes as an input a guess for this final time condition, $\hat{\lambda}_N$, and solves the corresponding control problem. If we denote this application of the FBSM as the function $V(\hat{\lambda}_N)$, and the corresponding final value of the state, $\hat{x}_N$, then the adapted FBSM is an iterative process that solves for the root of $V(\hat{\lambda}_N)$; the value of $\hat{\lambda_N}$ for which $x_N - \hat{x}_N = 0$. This outer iterative process can be solved using standard techniques such as the bisection method or secant method; the former converging more reliably provided that the initial guesses for $\hat{\lambda}_N$ form an interval that brackets the root, the latter converging in fewer iterations [43]. Each of these outer iterations necessitates solving a boundary value problem to convergence, often involving numerous iterations of the FBSM. In this work we apply the secant algorithm as presented in [43] without modification, for the adapted FBSM. The acceleration techniques described in §4 are applied only to the inner FBSM processes, reducing $\mathcal{N}$ for each internal FBSM problem, leaving the outer secant iterations unchanged. Using $\mathcal{N}^{(k)}$ to denote the number of function evaluations in the kth internal FBSM problem, we can express the cumulative function evaluations required for convergence of the adapted FBSM as Σ, such that $\Sigma = \mathcal{N}^{(1)} + \mathcal{N}^{(2)} + ...$

3 Control problems

To investigate the robustness and effectiveness of the iterative acceleration techniques that we will discuss in §4, we consider two distinct systems, and for each system we study three example control problems. The first system is a single species linear differential equation subject to a control. We later demonstrate that under certain conditions we are able to obtain exact solutions for control problems applied to this model. The second system is a three species model for acute myeloid leukaemia (AML) governed by a coupled nonlinear

system of differential equations, subject to a control. We construct the linear model to examine the behaviour of the acceleration techniques as applied to a simple idealised set of control problems. We include the AML model, variations upon which have been considered in the literature [24,70,71], to examine how the acceleration techniques perform when applied to problems more reflective of those considered in applied optimal control. For each model, we consider three distinct control problems; continuous control, bang-bang control and continuous control with fixed endpoint.

For all control results presented in this work, convergence is deemed to be achieved when the error, ε, measured as the Euclidean norm of the difference between subsequent controls, falls below a tolerance of 1×10^{-10}. Numerical solutions to ODEs are obtained using a fourth-order Runge-Kutta method [62] with constant time-stepping. A uniform time discretisation is sufficient for all control problems considered in this work. However, the FBSM and acceleration methods readily generalise to a non-uniform discretisation. If the desired discretisation for the state equations differs from that of the co-state equations, it is necessary to perform interpolation within each iteration of the FBSM to obtain values at corresponding time points. This can be computationally expensive and introduce an additional source of error. Where the desired discretisations for the state and co-state differ, numerical schemes with internal interpolation such as Continuous Runge-Kutta methods may be appropriate [10,80].

3.1 Single-variable linear model

The linear model is a single species model for the growth of $x(t)$, subject to control $u(t)$ that increases the growth rate. This model could represent our stylised crop growth example presented in §1. We suppress the explicit time dependence of the state and co-state variables and the control in the following

equations for notational convenience. For numerical results, we solve the linear problems on the domain $0 \leq t \leq 1$, with time-step $\mathrm{d}t = 3.91 \times 10^{-3}$, giving $N = 257$ time points.

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = \gamma x(t) + u(t), \quad x(0) = x_0, \quad \gamma > 0, \quad 0 \leq t \leq 1. \tag{8}$$

In absence of control, $u(t) \equiv 0$, this model admits the solution $x(t) = x_0 \mathrm{e}^{\gamma t}$, describing exponential growth.

3.1.1 Continuous control

We seek to maximise a quadratic cost function J, subject to

$$J = \int_0^1 (ax^2 - bu^2)\mathrm{d}t, \quad a > 0, b > 0. \tag{9}$$

Following the standard Pontryagin Maximum Principle approach for solving optimal control problems, we form the Hamiltonian and derive the co-state equation, transversality condition and optimality condition. The Hamiltonian is given by

$$H = ax^2 - bu^2 + \lambda(\gamma x + u). \tag{10}$$

The co-state equation is

$$\frac{\mathrm{d}\lambda}{\mathrm{d}t} = -\frac{\partial H}{\partial x} = -2ax - \lambda\gamma, \tag{11}$$

with transversality condition $\lambda(1) = 0$. In this case the optimality condition is

$$\frac{\partial H}{\partial u} = \lambda - 2bu = 0,$$

such that the optimal control is given by

$$u^*(t) = \frac{\lambda(t)}{2b}. \tag{12}$$

For model parameter $\gamma = 0.5$ and pay-off weightings $a = b = 1$, with initial condition $x_0 = 1$, we are able to solve the control problem analytically using standard techniques for linear systems with complex eigenvalues [39]. The process is laborious so we present the approach and analytical solution in §2 of the supplementary material. In the supplementary material we also plot the analytical results against the numerical results to demonstrate the excellent agreement. The numerical solution to the linear continuous control problem is presented in Figure 3. Convergence via the FBSM requires $\mathcal{N} = 57$ iterations.

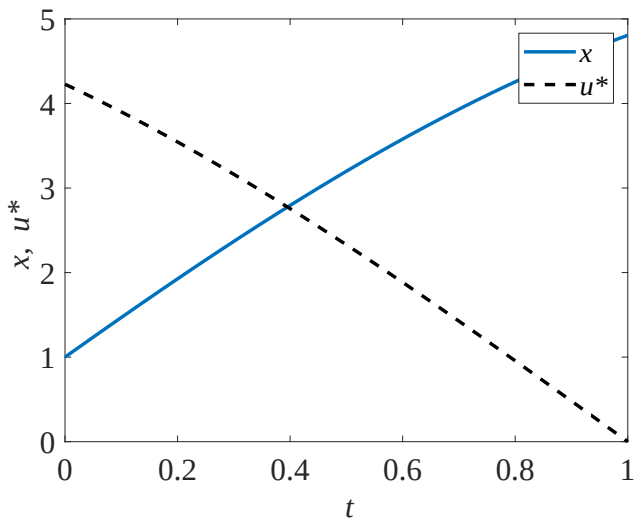


Fig. 3. Solution to the linear continuous control problem. The optimal control, $u^*(t)$, is shown in black dash and the corresponding state, $x(t)$, in blue. This solution is produced with model parameter $\gamma = 0.5$, time-step $\mathrm{d}t = 3.91 \times 10^{-3}$, over the interval $0 \leq t \leq 1$. The contributions of the state and the control to the pay-off are equally weighted, with $a = b = 1$.

3.1.2 Bang-bang control

For the bang-bang control we consider the same state equation as in Equation (8), and incorporate bounds on the control.

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = \gamma x(t) + u(t), \quad x(0) = x_0, \quad \gamma > 0, \quad 0 \leq t \leq 1, \quad 0 \leq u(t) \leq 2.$$

We seek to maximise a cost function J that is linear in u,

$$J = \int_0^1 (ax^2 - bu)\mathrm{d}t, \quad a > 0, b > 0.$$

We form the Hamiltonian and derive the co-state equation and transversality condition:

$$H = ax^2 - bu + \lambda(\gamma x + u).$$

The co-state equation is

$$\frac{\mathrm{d}\lambda}{\mathrm{d}t} = -\frac{\partial H}{\partial x} = -2ax - \lambda\gamma,$$

with transversality condition $\lambda(1) = 0$.

In seeking the optimality condition we find

$$\frac{\partial H}{\partial u} = \lambda - b. \tag{13}$$

As Equation (13) does not depend on u, we define a switching function

$$\psi(t) = \lambda - b,$$

and produce an expression for the control, based on the bounds on u and the

sign of the switching function:

$$u^*(t) = \begin{cases} 0, & \psi(t) < 0, \\ 2, & \psi(t) > 0. \end{cases} \tag{14}$$

If $\psi(t)$ is zero over any finite interval excluding isolated points, the optimal control is singular rather than bang-bang. Over such intervals, minimisation of the Hamiltonian does not provide sufficient information to determine the optimal control, and further conditions must be considered. [15,43]. We restrict our focus in this work to non-singular bang-bang optimal control problems. The numerical solution to the linear bang-bang control problem is presented in Figure 4. Convergence to this solution via the FBSM required $\mathcal{N} = 8$ iterations.

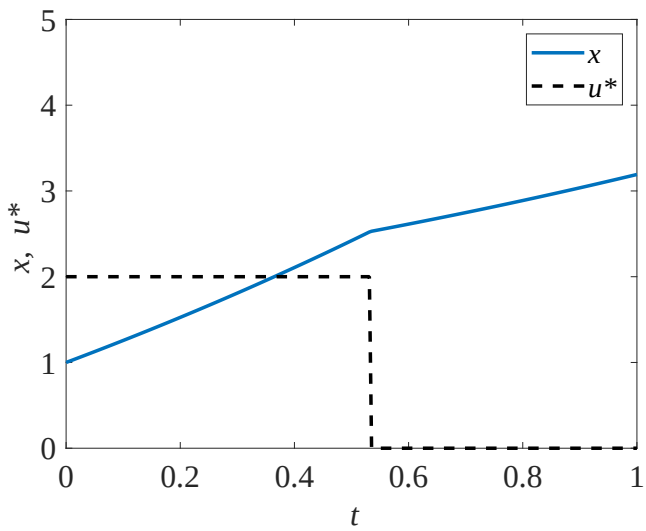


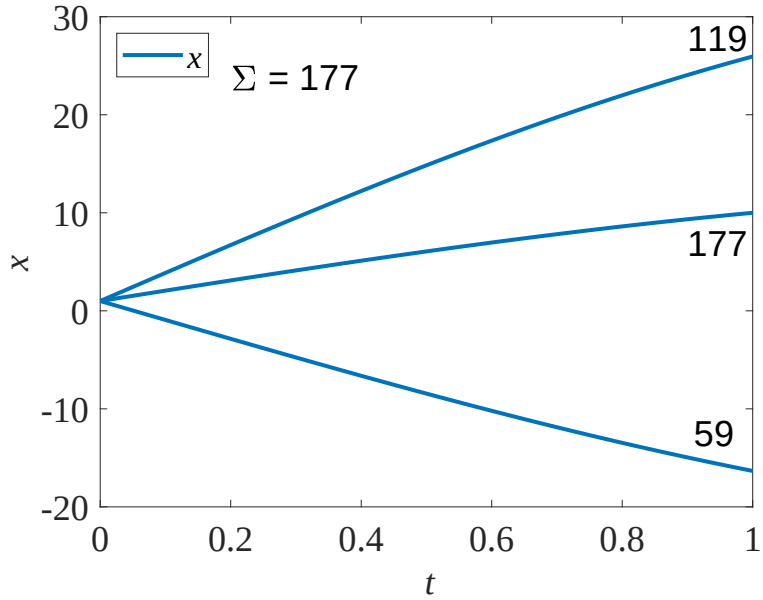
Fig. 4. Solution to the linear bang-bang control problem. The optimal control, $u^*(t)$, is shown in black dash and the corresponding state, $x(t)$, in blue. This solution is produced with model parameter $\gamma = 0.5$, time-step $\mathrm{d}t = 3.91 \times 10^{-3}$, over the interval $0 \leq t \leq 1$, with pay-off weightings of $a = 1$ for the state, and $b = 3$ for the control. The bang-bang control has prescribed bounds of $0 \leq u^*(t) \leq 2$.

3.1.3 Continuous control with fixed endpoint

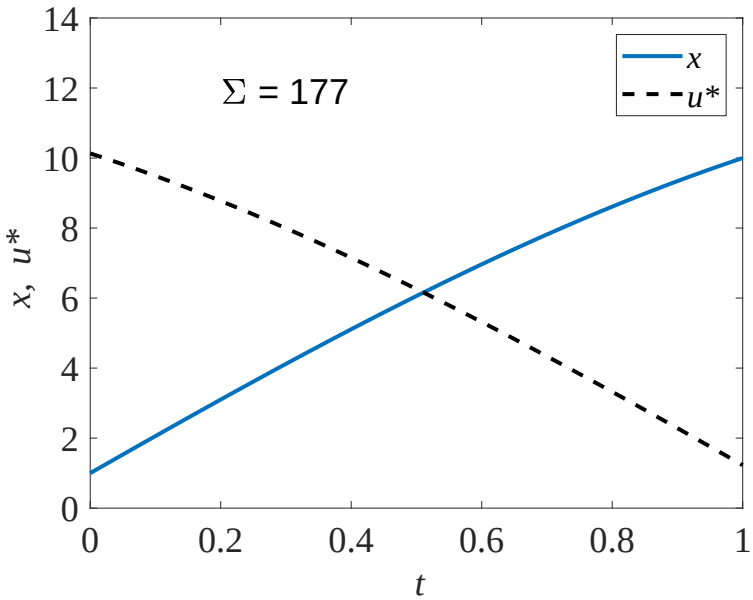
For the fixed endpoint problem, we proceed with the same state equation, however we now impose a terminal condition on x.

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = \gamma x(t) + u(t), \quad x(0) = x_0, \quad x(1) = 10, \quad \gamma > 0, \quad 0 \leq t \leq 1.$$

We seek to maximise the same quadratic cost function J, as considered in Equation (9). As such, we form the same Hamiltonian given in Equation (10) and derive the same co-state, Equation (11), and expression for the control, Equation (12). Note however that we do not prescribe a final time condition on the co-state equation via the transversality condition; as the system already has two boundary conditions, doing so would cause it to be overdetermined. Instead, we make two guesses for $\lambda(1)$; for example, $\lambda^{(0)}(1) = -10$ and $\lambda^{(1)}(1) = 10$. We proceed by applying the adapted FBSM outlined in §2, using these guesses to initialise the secant method. Numerical results for the linear fixed endpoint control problem are presented in Figure 5. Convergence of the adapted FBSM is achieved after $\Sigma = 177$ iterations.



(a) Iterates



(b) Converged result

Fig. 5. Results are presented for the linear problem with specified terminal state value, $x(t_N) = 10$, solved using the adapted FBSM. Underlying FBSM problems are solved with time-step $\mathrm{d}t = 3.91 \times 10^{-3}$, over the interval $0 \leq t \leq 1$, with pay-off weightings of $a = b = 1$. In (a) the $x(t)$ iterates of the adapted FBSM are presented. We annotate the cumulative function evaluations after the first ($\mathcal{N}^{(1)} = 59$) and second ($\mathcal{N}^{(1)} + \mathcal{N}^{(2)} = 119$) iterations of the adapted FBSM, based on initial guesses for $\lambda(t_N)$ of $\lambda(t_N) = -10$ and $\lambda(t_N) = 10$. The total cumulative function evaluations required for convergence of the adapted FBSM, $\Sigma = \mathcal{N}^{(1)} + \mathcal{N}^{(2)} + \mathcal{N}^{(3)} = 177$, is indicated. The converged result for $x(t)$, satisfying $|x(t_N) - 10| \leq 1 \times 10^{-10}$ is presented in (b); this figure also includes the optimal control, $u^*(t)$.

3.2 Multiple-variable nonlinear model

The AML model is a nonlinear coupled multi-species model describing the interactions between progenitor blood cells, $A(t)$, and leukaemic stem cells, $L(t)$, that occupy the same niche in the bone marrow; thereby competing for space and resources. Haematopoietic stem cells, $S(t)$, act as upstream production of $A(t)$. These dynamics have been explored in the literature both experimentally [38,72], and through mathematical modelling [24,47]. We subject the model to a chemotherapy-like control, $u(t)$, that acts as an additional death term for $L(t)$. The state can be expressed as $\boldsymbol{x}(t) = [S(t), A(t), L(t)]^T$. As there are now three state equations, we require three co-state equations; $\boldsymbol{\lambda}(t) = [\lambda_1(t), \lambda_2(t), \lambda_3(t)]^T$. We suppress the explicit time dependence of the state and co-state variables and the control in the following equations for notational convenience:

$$
\begin{aligned}
\frac{\mathrm{d}S}{\mathrm{d}t} &= \underbrace{\rho_S S(1-S)}_{\text{logistic growth}} - \underbrace{\delta_S S}_{\text{differentiation}} \quad , \\
\frac{\mathrm{d}A}{\mathrm{d}t} &= \underbrace{\delta_S S}_{\substack{\text{upstream}\\ \text{production}}} + \underbrace{\rho_A A(1-A-L)}_{\substack{\text{logistic growth}\\ \text{with competition}}} - \underbrace{\delta_A A}_{\text{differentiation}} \quad , \\
\frac{\mathrm{d}L}{\mathrm{d}t} &= \underbrace{\rho_L L(1-A-L)}_{\substack{\text{logistic growth}\\ \text{with competition}}} - \underbrace{\delta_L L}_{\text{differentiation}} - \underbrace{\frac{\alpha L}{\gamma + L}}_{\substack{\text{immune}\\ \text{response}}} - \underbrace{uL}_{\substack{\text{chemotherapy}\\ \text{control}}} \quad .
\end{aligned}
\tag{15}
$$

For each control problem associated with the AML model, we use initial conditions that yield a coexisting steady state in absence of control (all three species non-zero); $S(0) = 1 - \delta_S/\rho_S$, $A(0) = 0.3255$, and $L(0) = 0.3715$. We solve the AML problems numerically on the domain $0 \leq t \leq 10$, with time-step $\mathrm{d}t = 4.88 \times 10^{-4}$, giving $N = 20481$ time points. Model parameters are specified in Table 1.

Table 1: AML model parameters

Description	Variable	Value	Dimension
Proliferation of S	ρ_S	0.5	$[\mathrm{T}^{-1}]$
Proliferation of A	ρ_A	0.43	$[\mathrm{T}^{-1}]$
Proliferation of L	ρ_L	0.27	$[\mathrm{T}^{-1}]$
Differentiation of S into A	δ_S	0.14	$[\mathrm{T}^{-1}]$
Differentiation of A	δ_A	0.44	$[\mathrm{T}^{-1}]$
Differentiation of L	δ_L	0.05	$[\mathrm{T}^{-1}]$
Characteristic rate of the immune response	α	0.015	$[\mathrm{T}^{-1}]$
Half saturation constant of the immune response	γ	0.01	[-]

Parameters correspond to those presented with the original model [24], with immune response parameters introduced in subsequent work [70].

3.2.1 Continuous control

For the AML continuous control problem we seek to minimise a quadratic cost function J, that accounts for both the cost of applying the control and the cost of the leukaemic burden, subject to

$$J = \int_0^{10} (a_1 u^2 + a_2 L^2)\mathrm{d}t, \quad a_1 > 0, a_2 > 0. \tag{16}$$

We form the Hamiltonian and derive the co-state equation, transversality condition and optimality condition. The Hamiltonian is given by

$$\begin{aligned} H = a_1 u^2 + a_2 L^2 &+ (\rho_S S(1-S) - \delta_S S)\lambda_1 \\ &+ (\delta_S S + \rho_A A(1-A-L) - \delta_A A)\lambda_2 \\ &+ (\rho_L L(1-A-L) - \delta_L L - \frac{\alpha L}{\gamma + L} - uL)\lambda_3. \end{aligned} \tag{17}$$

The co-state equations are

$$\begin{aligned}
\frac{\mathrm{d}\lambda_1}{\mathrm{d}t} &= -\frac{\partial H}{\partial S} = -\rho_S\lambda_1 + 2\rho_S\lambda_1 S + \delta_S\lambda_1 - \delta_S\lambda_2, \\
\frac{\mathrm{d}\lambda_2}{\mathrm{d}t} &= -\frac{\partial H}{\partial A} = -\rho_A\lambda_2 + 2\rho_A\lambda_2 A + \rho_A\lambda_2 L + \delta_A\lambda_2 + \rho_L\lambda_3 L, \\
\frac{\mathrm{d}\lambda_3}{\mathrm{d}t} &= -\frac{\partial H}{\partial L} = -2a_2 L + \rho_A\lambda_2 A - \rho_L\lambda_3 + \rho_L\lambda_3 A + 2\rho_L\lambda_3 L \\
&\quad + \delta_L\lambda_3 + \frac{\alpha\gamma\lambda_3}{(\gamma+L)^2} + \lambda_3 u,
\end{aligned} \tag{18}$$

with transversality conditions $\lambda_1(10) = \lambda_2(10) = \lambda_3(10) = 0$, obtained in the usual way. In this case the optimality condition is

$$\frac{\partial H}{\partial u} = 2a_1 u - \lambda_3 L = 0, \tag{19}$$

such that the optimal control is given by

$$u^*(t) = \frac{\lambda_3 L}{2a_1}. \tag{20}$$

Numerical solutions for the AML continuous control problem are presented in Figure 6. These solutions are obtained via the FBSM, requiring $\mathcal{N} = 38$ iterations with $\omega = 0.55$. This choice of ω minimises $\mathcal{N}$ for the AML continuous control problem solved with the FBSM without acceleration techniques. We discuss the choice of ω further in §5.

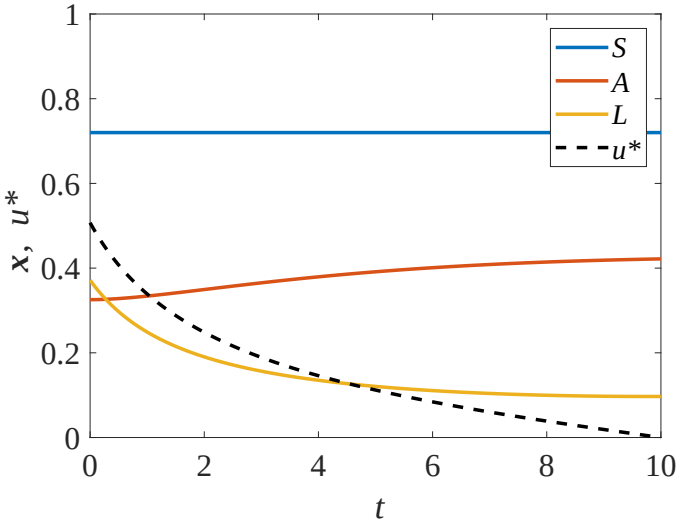


Fig. 6. Solution to the AML continuous control problem. The optimal control, $u^*(t)$, is shown in black dash and the corresponding state equations for $S(t)$, $A(t)$ and $L(t)$ are shown in blue, red and yellow, respectively. This solution is produced with model parameters given in Table 1, time-step $\mathrm{d}t = 4.88 \times 10^{-4}$, over the interval $0 \leq t \leq 10$, with pay-off weightings of $a_1 = 1$ for the control, and $a_2 = 2$ for the state variable $L(t)$.

3.2.2 Bang-bang control

For the bang-bang AML problem we consider the same states as in Equation (15), and incorporate bounds, $0 \leq u \leq 0.3$, on the control. We seek to minimise a cost function J that is linear in the control and the state variable L:

$$J = \int_0^{10} (a_1 u + a_2 L)\mathrm{d}t, \quad a_1 > 0, a_2 > 0. \tag{21}$$

We form the Hamiltonian and derive the co-state equations, transversality conditions and optimality condition. The Hamiltonian is given by

$$\begin{aligned} H = a_1 u + a_2 L &+ (\rho_S S(1-S) - \delta_S S)\lambda_1 \\ &+ (\delta_S S + \rho_A A(1-A-L) - \delta_A A)\lambda_2 \\ &+ (\rho_L L(1-A-L) - \delta_L L - \frac{\alpha L}{\gamma + L} - uL)\lambda_3. \end{aligned} \tag{22}$$

The co-state equations are

$$
\begin{aligned}
\frac{\mathrm{d}\lambda_1}{\mathrm{d}t} &= -\rho_S\lambda_1 + 2\rho_S\lambda_1 S + \delta_S\lambda_1 - \delta_S\lambda_2, \\
\frac{\mathrm{d}\lambda_2}{\mathrm{d}t} &= -\rho_A\lambda_2 + 2\rho_A\lambda_2 A + \rho_A\lambda_2 L + \delta_A\lambda_2 + \rho_L\lambda_3 L, \\
\frac{\mathrm{d}\lambda_3}{\mathrm{d}t} &= -a_2 + \rho_A\lambda_2 A - \rho_L\lambda_3 + \rho_L\lambda_3 A + 2\rho_L\lambda_3 L \\
&\quad + \delta_L\lambda_3 + \frac{\alpha\gamma\lambda_3}{(\gamma+L)^2} + \lambda_3 u,
\end{aligned}
\tag{23}
$$

with transversality conditions $\lambda_1(10) = \lambda_2(10) = \lambda_3(10) = 0$. In this case the switching function is

$$
\psi(t) = \frac{\partial H}{\partial u} = a_1 - \lambda_3 L, \tag{24}
$$

such that the optimal control is given by

$$
u^*(t) = \begin{cases} 0, & \psi(t) > 0, \\ 0.3, & \psi(t) < 0. \end{cases} \tag{25}
$$

Note that the correspondence between the sign of $\psi(t)$ and the chosen bound is reversed in Equation (25) relative to Equation (14) as we are now performing minimisation rather than maximisation. Numerical solutions for the AML bang-bang control problem are presented in Figure 7. These solutions are obtained via the FBSM, requiring $\mathcal{N} = 34$ iterations with $\omega = 0.4$. This choice of ω minimises $\mathcal{N}$ for the AML bang-bang control problem solved with the FBSM without acceleration techniques. We discuss the choice of ω further in §5.

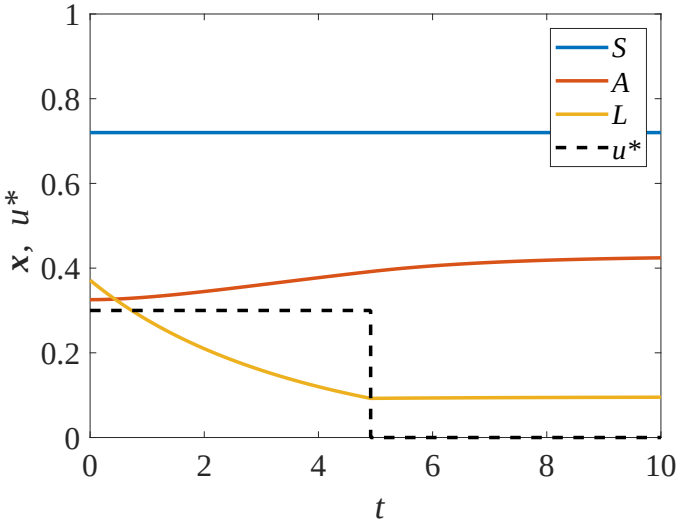


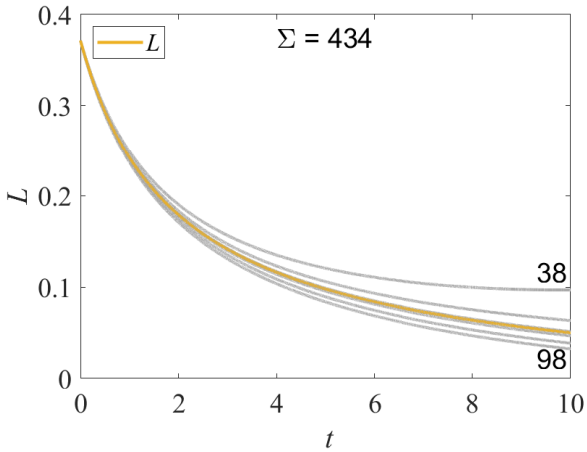
Fig. 7. Solution to the AML bang-bang control problem. The optimal control, $u^*(t)$, is shown in black dash and the corresponding state equations for $S(t)$, $A(t)$ and $L(t)$ are shown in blue, red and yellow, respectively. This solution is produced with model parameters given in Table 1, time-step $\mathrm{d}t = 4.88 \times 10^{-4}$, over the interval $0 \leq t \leq 10$, with pay-off weightings of $a_1 = 1$ for control, and $a_2 = 2$ for the state variable $L(t)$.

3.2.3 Continuous control with fixed endpoint

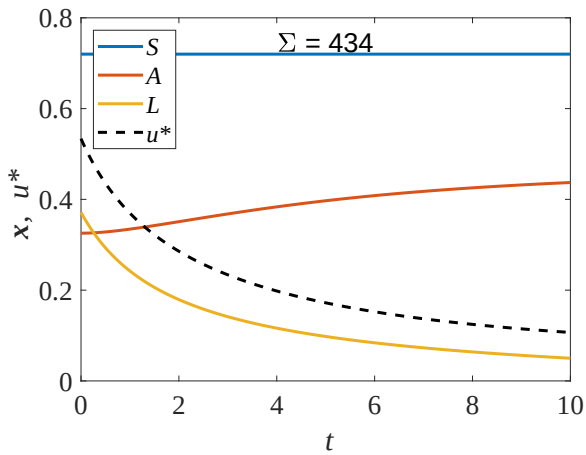
For the fixed endpoint problem, we proceed with the same state equations as for the AML continuous control problem given in Equation (15), however we now impose a terminal condition on the leukaemic population; $L(10) = 0.05$. We seek to minimise the same quadratic cost function J, as considered in Equation (16). We form the same Hamiltonian given in Equation (17) and derive the same co-state, Equation (18), and expression for the control, Equation (20).

We obtain final time conditions, $\lambda_1(10) = \lambda_2(10) = 0$, via the transversality conditions as usual, however we do not prescribe $\lambda_3(10)$. Instead, we make two guesses for $\lambda_3(10)$; for instance, $\lambda_3^{(0)}(10) = 0$ and $\lambda_3^{(1)}(10) = 10$. We then proceed by applying the adapted FBSM outlined in §2, using these guesses to initialise the secant method. Numerical results for the AML fixed endpoint control problem are presented in Figure 8. These results are produced using

the adapted FBSM with $\omega = 0.55$ in $\Sigma = 434$ iterations. This choice of ω minimises Σ for the AML fixed endpoint control problem solved with the FBSM without acceleration techniques. We discuss this further in §5.



(a) Iterates



(b) Converged result

Fig. 8. Results are presented for the AML problem with specified terminal state, $L(t_N) = 0.05$, solved using the adapted FBSM. Each underlying FBSM problem is solved with model parameters given in Table 1, time-step $\mathrm{d}t = 4.88 \times 10^{-4}$, over the interval $0 \leq t \leq 10$, with pay-off weightings of $a_1 = 1$ for the control, and $a_2 = 2$ for the state variable $L(t)$. In (a) the $L(t)$ iterates of the adapted FBSM are presented in grey; the converged solution satisfying $L(t_N) = 0.05$ is plotted in yellow. We annotate $\mathcal{N}$ for the first ($\mathcal{N}^{(1)} = 38$) and second ($\mathcal{N}^{(1)} + \mathcal{N}^{(2)} = 98$) iterations of the adapted FBSM, based on initial guesses for $\lambda_3(t_N)$ of $\lambda_3(t_N) = 0$ and $\lambda_3(t_N) = 10$. Due to the close proximity, subsequent iterations are not annotated. The cumulative function evaluations required for convergence of the adapted FBSM ($\Sigma = 434$) is indicated. The converged result for $L(t)$, satisfying $|L(t_N) - 0.05| \leq 1 \times 10^{-10}$, is presented in (b); this figure also includes the optimal control, $u^*(t)$ and trajectories for $S(t)$ and $A(t)$.

4 Iterative accelerators

In this section we outline several techniques for acceleration of iterative schemes. Where appropriate, we first present the univariate/scalar version of the method for familiarity, then provide the multivariate/vector analogue of the method for use with accelerating the FBSM. We attempt to use notation that aligns most closely with commonly used notation in the literature, while maintaining internal consistency in this work. In the scalar case, we consider the iterative process $x^{(k+1)} = f(x^{(k)})$, where $x^{(k)}$ is the kth iterate and f is the iterating function. In the vector case we consider $X^{(k+1)} = F(X^{(k)})$, where $X^{(k)} = [x_0^{(k)}, x_1^{(k)}, ...x_N^{(k)}]^{\mathrm{T}}$ is the kth iterate, consisting of $N+1$ values, and $F = [f_0, f_1, ...f_N]^{\mathrm{T}}$ is the $N+1$ dimensional operator of the iterative process. For clarity, we stress that in the context of the acceleration algorithms applied to the FBSM, $X^{(k)}$ is the discretised control in the kth iteration.

The acceleration methods considered in this work apply either to problems stated as fixed point iterations (as above), or as root-finding problems. For acceleration via root-finding algorithms, we can consider the complementary problems in the scalar and vector setting, respectively; $g(x) := x - f(x) = 0$ and $G(X) := X - F(X) = \mathbf{0}$, where $\mathbf{0}$ is the zero column vector of length $N+1$.

We note that many of the methods presented here can be written in several different forms. While some forms better facilitate analysis of aspects such as convergence speed and numerical stability, others emphasise ease of understanding and implementation. In this work we prioritise usability and present methods and algorithms in forms reflective of their implementation where possible. For the purpose of this work, we feel it is sufficient to present the methods and discuss their implementation without delving into their derivation or rigorous theoretical convergence results. For readers interested in these aspects, we suggest these articles [13,63], and numerical analysis texts [17,37].

4.1 Newton and Quasi-Newton methods

Newton's method is one of the most prevalent root-finding algorithms, due to its relatively straightforward implementation and potential for quadratic convergence [37]. For a univariate function Newton's method is given by

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}. \tag{26}$$

We arrive at the scalar secant method by replacing the derivative term, $f'(x^{(k)})$, in Equation (26) with a finite difference approximation:

$$x^{(k+1)} = x^{(k)} - f(x^{(k)})\frac{x^{(k)} - x^{(k-1)}}{f(x^{(k)}) - f(x^{(k-1)})}.$$

Newton's method for multivariate systems is

$$\begin{aligned} X^{(k+1)} &= X^{(k)} + \Delta X^{(k)}, \\ &\quad \text{where } \Delta X^{(k)} \text{ is obtained by solving} \\ J_k \Delta X^{(k)} &= -F(X^{(k)}). \end{aligned}$$

Here, J_k is the Jacobian matrix of F evaluated at $X^{(k)}$ [37]. Setting aside the interpretation of the Jacobian in the context of the FBSM; numerically approximating an $N \times N$ Jacobian matrix using finite differences requires $\mathcal{O}(N^2)$ FBSM iterations at each Newton step. A range of Quasi-Newton methods have been developed to minimise the computational expense associated with computing the Jacobian at each Newton step. It is not immediately apparent

how the secant method should be extended to multivariate systems, but one such interpretation is the Quasi-Newton Broyden's method. Broyden's method reduces the number of function evaluations required at each Newton step by forming the full Jacobian only initially, then updating the Jacobian matrix via a rank-one update based on the secant method [14,17]. We later discuss the Wegstein method [78], which is another interpretation of the secant method in multivariate settings.

In the context of accelerating the FBSM, techniques that require forming or approximating a full Jacobian, even once, are not appropriate. We have an $N + 1$ dimensional system, where $N + 1$ is the number of time points in the discretisation of the ODEs, so we expect N to be large, relative to the number of iterations required for the FBSM to converge without acceleration techniques, via Equation (6). As such, we restrict our focus to Jacobian-free methods in the remainder of this section; in particular, we discuss and implement the Wegstein and Aitken-Steffensen methods and Anderson acceleration. We provide a broad overview alongside the key equations here, and provide complete algorithms alongside notes for implementation in §4 of the supplementary material.

4.2 Wegstein method

Wegstein's method can be thought of as an element-wise extension of the secant method to multivariate systems [33]. Although Wegstein's method appears less popular than other methods considered in this work, it has found practical utility, particularly in chemical and process engineering software [48,73]. We include it here due to the striking similarity it bares to the control update with relaxation presented in Equation (6). It is also one of the more straightforward techniques, both in conception and implementation:

$$\hat{x}^{(k+1)} = f(x^{(k)}),$$

$$x^{(k+1)} = q^{(k)}x^{(k)} + (1 - q^{(k)})\hat{x}^{(k+1)}, \tag{27}$$

$$\text{where } q^{(k)} = \frac{a^{(k)}}{a^{(k)} - 1}, \text{ and } a^{(k)} = \frac{f(x^{(k)}) - f(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}. \tag{28}$$

In implementation, from an initial value x_0 it is necessary to perform two function evaluations; i.e. $x_1 = f(x_0)$, and $f(x_1)$, before it is possible to compute Equation (28) for the Wegstein method [78]. In subsequent iterations only one new function evaluation is required.

The extension of Wegstein's method to multivariate systems follows exactly the process outlined in Equations (27) and (28), as it is extended element-wise. While convergence is guaranteed when using Wegstein's method for a single nonlinear equation, the uncoupling implied by the element-wise extension can lead to divergence [57].

In Equations (27) and (28) $q^{(k)}$ denotes q in the kth iteration, however, we note that it is not necessarily most effective to update q every iteration. As such, in this work we explore various updating regimes. There is also the option of applying bounds on q. Bounds of $-5 < q_i < 0$, $\forall i$, where i denotes the ith element of the system, are frequently applied when implementing Wegstein's method [7,67]. This bounding appears to work reasonably well for the small nonlinear test systems we consider in §5 of the supplementary material, although we were not able to identify a theoretical result supporting this specific choice. For the control problems we consider, this bounding is not effective, so we apply different bounds, discussed further in §5. The univariate Wegstein method can be thought of as a modification of the Aitken method, which at the time the Wegstein method was developed, was only understood for the univariate case [23].

4.3 Aitken-Steffensen method

Aitken's Δ^2 method, also referred to as Aitken's delta-squared process or Aitken extrapolation, was originally posed by Aitken in 1927 as a means of extending Bernoulli's method of approximating the largest root of an algebraic equation. This extension facilitates numerically approximating not only the largest root, but all roots of the equation [3]. Aitken's method generates a new sequence, $\hat{x}$, in parallel to the fixed point iteration.

$$\begin{aligned} \hat{x}^{(k)} &= x^{(k)} - \frac{(x^{(k+1)} - x^{(k)})^2}{x^{(k+2)} - 2x^{(k+1)} + x^{(k)}}, \quad \text{or} \\ \hat{x}^{(k)} &= x^{(k)} - \frac{(\Delta x^{(k)})^2}{\Delta^2 x^{(k)}}, \end{aligned} \tag{29}$$

where Δ is the difference operator; $\Delta x^{(k)} = x^{(k+1)} - x^{(k)}$, and the higher order operator is applied recursively; $\Delta^2 x^{(k)} = \Delta(\Delta x^{(k)}) = \Delta x^{(k+1)} - \Delta x^{(k)}$ [37]. From an initial value, $x^{(0)}$, two function evaluations; iterations of the underlying fixed point process, must be performed to obtain $x^{(1)}$ and $x^{(2)}$, before Equation (29) can be computed.

The derivation of Aitken's method assumes an underlying linearly converging series of iterates. The order of convergence of the resulting Aitken accelerated series is still linear, however this series converges faster than the original series [17]. We discuss Aitken's Δ^2 method and Steffensen iteration together, as Steffensen iteration is a straightforward extension of Aitken's method, whereby the Aitken value, $\hat{x}^{(k)}$, is used to continue the fixed point iteration, i.e. $x^{(k+1)} = \hat{x}^{(k)}$. Despite the striking similarity, Steffensen's method was seemingly developed shortly after (1933) and without knowledge of Aitken's method [75]. Steffensen iteration can achieve quadratic convergence [2,37,52]. Further theoretical convergence results for the Steffensen method are estab-

lished by Nievergelt [53] and in a series of papers by Noda [54,55,56].

Aitken and Steffensen iteration can be extended to multivariate systems [37]. In the following statements we outline the method for an $N+1$ dimensional system; $X^{(k)} = [x_0^{(k)}, x_1^{(k)}, ..., x_N^{(k)}]^{\mathrm{T}} \in \mathbb{R}^{N+1}$, as appropriate for use with the FBSM:

$$\hat{X}^{(k)} = X^{(k)} - \Delta\mathcal{X}^{(k)}(\Delta^2\mathcal{X}^{(k)})^{-1}\Delta X^{(k)}, \tag{30}$$

where $\Delta X^{(k)} = X^{(k+1)} - X^{(k)}$, $\mathcal{X}^{(k)}$ is a matrix constructed with columns $(X^{(k)}, X^{(k+1)}, ..., X^{(k+N)})$, such that $\mathcal{X}^{(k)}$ is a square matrix of dimension $N+1$, with $\Delta\mathcal{X}^{(k)} = \mathcal{X}^{(k+1)} - \mathcal{X}^{(k)}$, and $\Delta^2\mathcal{X}^{(k)} = \Delta\mathcal{X}^{(k+1)} - \Delta\mathcal{X}^{(k)}$.

In the form given by Equation (30) there are glaring issues with using the Steffensen method to accelerate convergence of the FBSM. Setting aside the question of whether $\Delta^2\mathcal{X}^{(k)}$ is invertible, forming $\mathcal{X}^{(k)}$ would require $\mathcal{O}(N)$ iterations of the FBSM to be performed, and since N relates to the number of time points in the discretisation of the ODEs in the FBSM, we expect N to be large, relative to the number of iterations required for the FBSM to converge without acceleration.

We instead consider a modification of the Steffensen method, requiring fewer function evaluations per iteration. Introduce $m < N$, and define $\Delta X^{(k)} = X^{(k+1)} - X^{(k)}$ as before, $\mathcal{X}^{(k)}$ is now a rectangular matrix constructed with columns $(X^{(k)}, X^{(k+1)}, ..., X^{(k+m+1)})$, such that $\mathcal{X}^{(k)} \in \mathbb{R}^{N+1\times m+2}$, with $\Delta\mathcal{X}^{(k)} = \mathcal{X}^{(k+1)} - \mathcal{X}^{(k)}$, and $\Delta^2\mathcal{X}^{(k)} = \Delta\mathcal{X}^{(k+1)} - \Delta\mathcal{X}^{(k)}$, both of dimension $N+1\times m$. We now interpret the matrix inverse in Equation (30) as the Moore-Penrose pseudoinverse [59], a generalisation of the matrix inverse for singular and rectangular matrices; we discuss this further in §3 of the supplementary material. This partial implementation requires only $m+1$ function evaluations per it-

eration. For the remainder of this document when referring to the Steffensen method we are specifically referring to this partial Steffensen implementation. We present the derivation of the multivariate Aitken-Steffensen method and outline where the partial implementation differs, in §3 of the supplementary material.

4.4 Anderson Acceleration

Anderson Acceleration or Anderson Mixing, originally denoted as the extrapolation algorithm by Anderson in the 1960s [5], is a technique developed for accelerating convergence of fixed point iteration problems with slowly converging Picard iterations [6]. Anderson Acceleration is of particular interest in this work, as it has recently been implemented to accelerate the convergence of a regularised version of the FBSM [45]. In contrast to a standard fixed point iteration; whereby the next iterate depends only on the immediately preceding iterate, Anderson Acceleration has 'memory' through the inclusion of previous iterates [28]. Unlike other methods considered in this work, Anderson Acceleration explicitly utilises the differences between residuals of subsequent iterates alongside iterates and their differences in computing future iterates.

Anderson Acceleration involves solving a least-squares problem at each iteration. The problem can be expressed in both constrained and unconstrained forms, with the updating step dependent on the form [30,77]. We solve the following unconstrained least-squares problem in each iteration of Anderson Acceleration:

$$\gamma = \arg\min_{\gamma}(||G - \gamma dG||), \tag{31}$$

where $\arg\min(\cdot)$ returns the argument, γ, that minimises the expression in Equation (31). The corresponding updating step is

$$X^{(k+1)} = X^{(k)} + G^{(k)} - (dX^{(k-1)} + dG^{(k-1)})\gamma, \tag{32}$$

where $G^{(k)} = F(X^{(k)}) - X^{(k)}$ is the residual, $dX^{(k)}$ is a matrix with columns $(\Delta X^{(k-m)}, \Delta X^{(k-m+1)}, ..., \Delta X^{(k)})$, and $dG^{(k)}$ is a matrix with columns $(\Delta G^{(k-m)}, \Delta G^{(k-m+1)}, ..., \Delta G^{(k)})$, and m indicates the number of previous iterates that are incorporated.

4.5 Acceleration methods applied to typical fixed point problems

As a precursor to implementing these acceleration methods for control problems, we apply them to solve example nonlinear systems of dimension 2×2, 3×3, and 4×4. We provide these systems and the results of the acceleration methods compared to standard fixed point iteration in §5 of the supplementary material. We do not discuss these results in detail, although broad comparisons regarding the application of the acceleration methods to these systems and to control problems are made in §6. We provide code on GitHub for implementing the acceleration algorithms to solve systems of arbitrary size.

5 Acceleration results

In this section we discuss the results of applying the acceleration algorithms. When discussing results we are solely focused on reducing $\mathcal{N}$, the number of function evaluations required for the control problems to reach convergence; as in all convergent cases we arrive at the same optimal control results. We

first discuss the aspects of each method that can be tuned, then outline the results of the standard FBSM with the best choice of ω but without acceleration methods applied, to establish a baseline against which to compare the acceleration methods. A detailed suite of results for each control problem and each acceleration method, for various combinations of tuning parameters, is provided in §6 of the supplementary material.

5.1 Tuning

Each method we consider has parameters that can be tuned to improve performance for a given problem. For the FBSM without acceleration, we can select $\omega \in [0, 1)$; the parameter that weights the contribution of the control from the previous iteration, and the newly calculated control, to the control used in the next iteration, as stated in Equation (6). Control problems based on the linear model are able to converge via direct updating, as given in Equation (5), equivalent to $\omega = 0$. Increasing ω in this case only serves to increase $\mathcal{N}$, so we do not attempt to tune ω when considering the linear model. Using the standard FBSM without acceleration the continuous linear problem requires $\mathcal{N} = 57$ while the bang-bang linear problem requires only $\mathcal{N} = 8$.

In Figure 9 we plot $\mathcal{N}$ against $\omega \in [0, 1)$, for the continuous and bang-bang AML problems. As expected, for small ω we find that the problem does not converge, and for large ω, $\mathcal{N}$ increases. For the continuous AML problem we identify $\omega = 0.55$ as the best choice, with $\mathcal{N} = 38$. For the bang-bang AML problem we find that $\omega = 0.4$ is best, with $\mathcal{N} = 34$.

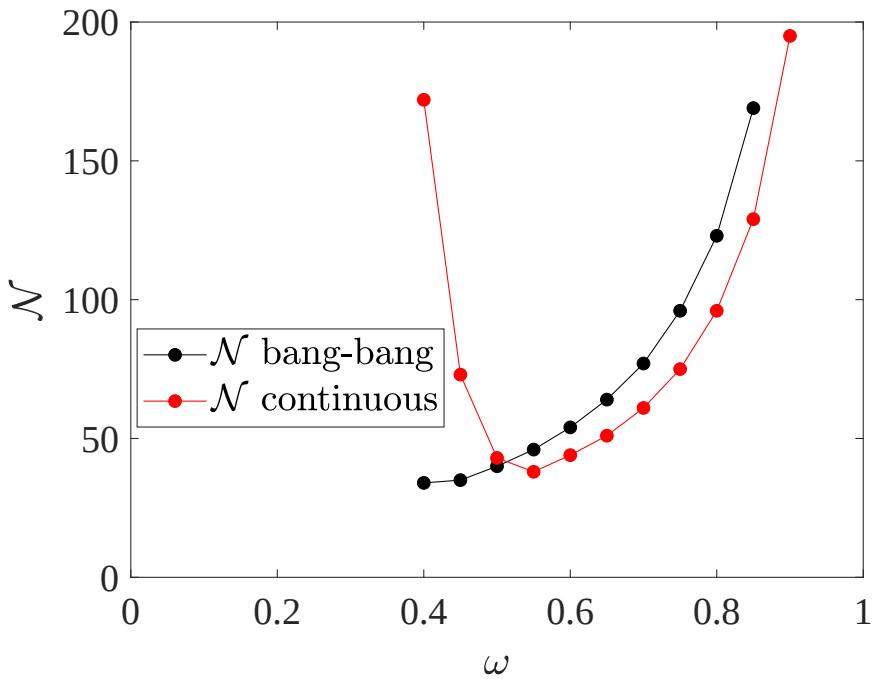


Fig. 9. Here we plot $\mathcal{N}$ against $\omega \in [0, 1)$ in increments of 0.05, for the AML continuous (in red) and bang-bang (in black) control problems, using FBSM with no acceleration. Results correspond to model parameters given in Table 1, time-step $\mathrm{d}t = 4.88 \times 10^{-4}$, over the interval $0 \leq t \leq 10$. The continuous problem is solved with pay-off weightings of $a_1 = 1$ for control, and $a_2 = 2$ for the state variable $L(t)$, while the bang-bang problem is solved with $a_1 = 1$ and $a_2 = 3$. Where an ω value does not have a corresponding marker, this indicates that the procedure fails to converge within $\mathcal{N} = 200$.

Recall that fixed endpoint problems are solved using the adapted FBSM; this entails solving several control problems to convergence with the FBSM. Each of these control problems can have a different optimal ω. In this instance, $\omega = 0.55$ also happens to be best for the AML fixed endpoint problem if holding ω constant, when considering $\omega \in [0, 1)$ at increments of 0.05. These ω values will not coincide in general. When applying the acceleration methods to the fixed endpoint problems, we employ the tuning parameters that perform best for the continuous problem. This does not imply that we are using the best tuning parameters for the acceleration methods in the context of the fixed endpoint problem. Importantly, this demonstrates whether or not the techniques can effectively reduce Σ, the cumulative function evaluations

required for convergence of the adapted FBSM for fixed endpoint problems, without requiring prohibitive tuning.

With the Wegstein method, we only select ω for the two FBSM iterations required for initialisation, and specify n, such that we update q every nth iteration. We generate results for $n \in \{1, 2, ..., 10\}$. We also bound q, however identifying suitable bounds is challenging. In this work we select bounds that perform reasonably, but acknowledge that we do not search for optimal bounds, nor do we think that attempting to do so is realistic. This drawback of Wegstein's method contributes to its inconsistent performance relative to other methods. For the partial Aitken-Steffensen methods, we choose ω, and the parameter m that specifies the dimension of the $N+1 \times m$ matrices in the updating step, requiring $m+1$ function evaluations per iteration. We generate results for $m \in \{1, 2, ..., 10\}$. Similarly, for Anderson Acceleration we select ω and M; where M determines the maximum number of previous iterations to retain when solving the least squares problem and performing the updating step. We produce results for $M \in \{1, 2, ..., 10\}$.

5.2 Wegstein method

For the continuous linear problem, we apply bounds of $-2 \leq q \leq 0$. For the bang-bang linear problem, we leave q unbounded. For both AML problems we apply bounds $-1 \leq q \leq 1$. We explore the effect of updating q every nth iteration, $n \in \{1, 2, ..., 10\}$. For the continuous linear problem, $n = 4$ minimises $\mathcal{N}$, although $n \in \{1, 2, ..., 5\}$ all perform well. For the linear bang-bang problem, the Wegstein method converges without bounding on q, and varying n does not affect convergence. The Wegstein method outperforms other acceleration methods for the linear bang-bang problem with $\mathcal{N} = 9$, but does not improve upon $\mathcal{N} = 8$ for the FBSM without acceleration.

For the continuous AML problem, the performance of Wegstein's method is inconsistent. With $n = 6$ and $\omega = 0.55$, the Wegstein method achieves convergence with $\mathcal{N} = 26$, outperforming the FBSM; however, almost every other combination of tuning parameters considered with $\omega \in [0, 1)$ and $n \in \{1, 2, ..., 10\}$ require larger $\mathcal{N}$ than the FBSM without acceleration. Generally, increasing n produces worse outcomes. We do, however, observe that the Wegstein method can induce convergence for $\omega < 0.4$, where the standard FBSM does not converge. For the bang-bang AML problem, the Wegstein method appears more robust; consistently outperforming the standard FBSM across most of the tuning parameter space. The best result requires only $\mathcal{N} = 9$, with $\omega = 0$ and $n = 7$, although several other combinations of tuning parameters are similarly successful. For $\omega \geq 0.4$, corresponding to values that the underlying FBSM converges, we find that moderate $n \in \{3, 4, ..., 7\}$ produces the best results; while for smaller ω, larger $n \in \{6, 7, ..., 10\}$ consistently performs best. Once again we observe that convergence is achieved for ω values where the underlying FBSM would not converge.

For the linear fixed endpoint problem the adapted FBSM with the Wegstein method consistently generates a moderate reduction in Σ, compared to the adapted FBSM without acceleration, for all $n \in \{1, 2, ..., 10\}$, $-2 \leq q \leq 0$. For the AML fixed endpoint problem we do not observe improvement. Using the tuning parameters that perform best for the continuous AML problem, we find that Σ for the adapted FBSM with Wegstein's method is more than double that of the adapted FBSM without acceleration. This results from the inconsistency of Wegstein's method with poor tuning. In §6 of the supplementary material it can be seen that some control problems within the adapted FBSM that require $\mathcal{N} \approx 50$ without Wegstein's method, require $\mathcal{N} \approx 200$ with the specified Wegstein tuning parameters.

5.3 Partial Aitken-Steffensen method

Both Aitken and Steffensen methods significantly and consistently outperform the FBSM without acceleration for the continuous linear problem. The Aitken method performs best for $m \in \{1, 2, 3\}$, requiring $\mathcal{N} = 12$. Steffensen's method performs best when $m = 6$, requiring only $\mathcal{N} = 8$. In the linear bang-bang case, both Aitken and Steffensen methods perform marginally worse than the FBSM without acceleration, which requires only $\mathcal{N} = 8$. In the best cases, with $m = 1$ the Aitken method requires $\mathcal{N} = 10$, and with $m = 7$ the Steffensen method requires $\mathcal{N} = 9$.

For the continuous AML problem, we observe a stark difference between the Aitken and Steffensen methods; while the Steffensen method is able to achieve convergence for values of ω where the underlying FBSM fails to converge; $\omega \leq 0.35$, particularly for $m \in \{1, 2, 3, 4\}$, the Aitken method only converges to the optimal control for ω values where the underlying FBSM converges. For $\omega \leq 0.35$ the Aitken method achieves apparent convergence; the iterative procedure terminates as the convergence criteria is met. However, explicitly calculating the pay-off associated with these controls via Equation (16), and comparing this result to the pay-off associated with the control obtained via the standard FBSM, indicates that the controls obtained via the Aitken method for $\omega \leq 0.35$ are not optimal, as they fail to minimise J. The best result for the Aitken method, with $\omega = 0.5$ and $m = 5$, requires $\mathcal{N} = 30$, marginally improving on the FBSM without acceleration, requiring $\mathcal{N} = 38$. Steffensen's method produces more significant improvements, requiring only $\mathcal{N} = 19$ with $\omega = 0.5$ and $m = 5$. In each case neighbouring combinations of tuning parameters also yield equivalent or comparable improvement over the standard FBSM. In the bang-bang AML problem, we observe similar behaviour; for ω values that the underlying FBSM fails to converge, the Steffensen method consistently converges. The Aitken method achieves apparent convergence for these values

of ω; the iterative procedure terminates as the convergence criteria is met, however the resulting controls contain intermediate values between the lower and upper bounds. As such the resulting controls are not bang-bang, so we treat these results as failing to converge. At best, Aitken's method requires $\mathcal{N} = 8$, with $\omega = 0.5$ and $m = 1$, while Steffensen's requires only $\mathcal{N} = 7$, with $\omega = 0.5$ and $m = 5$. The vast majority of tuning parameter combinations yield improvements over the $\mathcal{N} = 34$ of the standard FBSM.

Aitken and Steffensen methods consistently offer significant improvement over the standard adapted FBSM for the linear fixed endpoint problem for $m \in \{1, 2, ..., 10\}$, with the exception of $m = 1$ for the Steffensen method, which yields only marginal improvement. Using the best performing tuning parameters for the continuous AML problem, we find that both Aitken and Steffensen methods improve upon the standard adapted FBSM for the AML fixed endpoint problem. Relative to $\Sigma = 434$ required without acceleration, the $\Sigma = 360$ required with Aitken's method reflects a modest improvement, while the $\Sigma = 238$ required with the Steffensen method is a significant improvement.

5.4 Anderson Acceleration

Anderson Acceleration performs exceptionally well on the continuous linear problem, requiring only $\mathcal{N} = 7$ for $M \in \{4, 5, ..., 10\}$ compared to $\mathcal{N} = 57$ for the standard FBSM. For the linear bang-bang problem however, it is the worst performing acceleration method; achieving at best $\mathcal{N} = 11$, with $M = 1$.

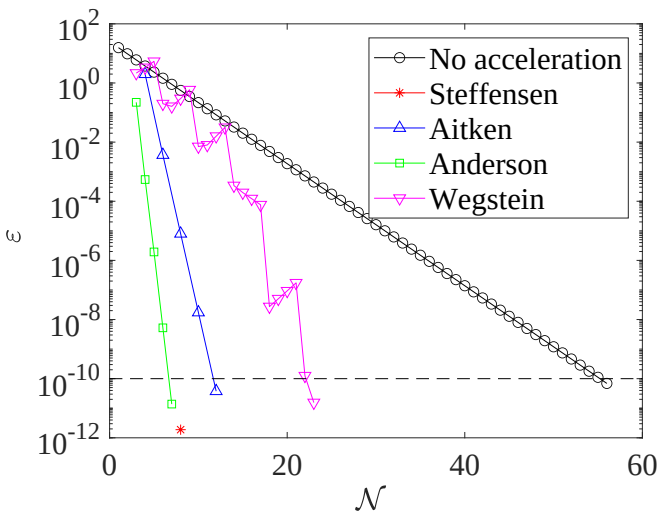
Similarly to the Wegstein and Steffensen methods, Anderson Acceleration achieves convergence in both the continuous and bang-bang AML problems for ω values where the underlying FBSM fails to converge. Anderson Acceleration achieves the best individual result for the continuous AML problem, requiring only $\mathcal{N} = 17$, with $\omega = 0.85$ and $M = 6$. Again, we observe compa-

rable improvement over a wide range of tuning parameters. For the bang-bang AML problem Anderson Acceleration consistently outperforms FBSM without acceleration, particularly for $\omega < 0.7$, at best requiring $\mathcal{N} = 17$, with $\omega = 0.35$ and $M \in \{7, 8, 9, 10\}$, with other non-neighbouring tuning parameter combinations also yielding $\mathcal{N} = 17$.

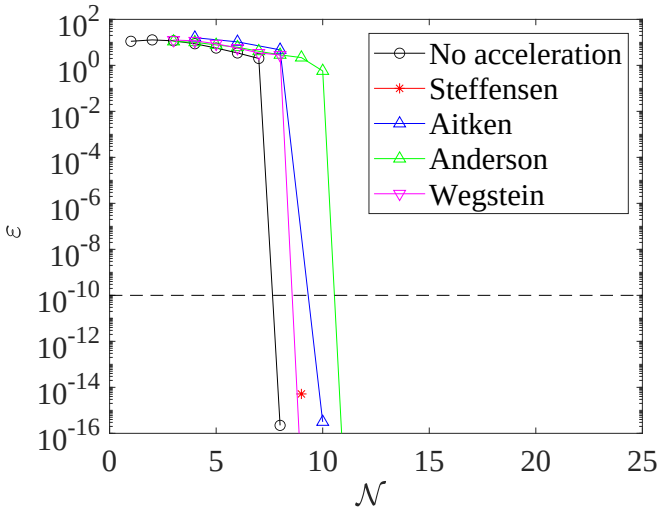
For both the linear and AML fixed endpoint problems Anderson Acceleration produces the most significant reduction in Σ, and improves upon the adapted FBSM over a wide range of tuning parameters. In the linear case, Anderson Acceleration requires only $\Sigma = 24$ for $M \in \{4, 5, ..., 10\}$. In the AML fixed endpoint problem, using the tuning parameters that perform best for the continuous AML problem, Anderson Acceleration converges in only $\Sigma = 204$; less than half as many as the standard adapted FBSM.

5.5 Method comparison with best tuning

Results presented in Figure 10 and Figure 11 provide comparison of the error, ε, as each method approaches convergence, for the linear and AML problems, respectively. Error is measured as the Euclidean norm of the difference between subsequent controls; $\varepsilon = ||F(X^{(k)}) - X^{(k)}||$, with the exception of Aitken's method, where error is measured as the difference between subsequent values in the Aitken series; $\varepsilon = ||\hat{X}^{(k)} - \hat{X}^{(k-1)}||$. Convergence is achieved when $\varepsilon \leq 1 \times 10^{-10}$, marked in black dash. In each case, we are plotting the result that minimises $\mathcal{N}$ for each method, over the space of tuning parameters considered, including the best tuning of ω for the FBSM without acceleration. Error is plotted on a logarithmic scale. For the linear bang-bang problem with the Wegstein and Anderson methods, and the AML bang-bang problem with the Wegstein method, the error after the final iteration is $\varepsilon = 0$, as two subsequent iterates for the control are identical. This is represented on the logarithmic scale as a line that intersects the horizontal axis.

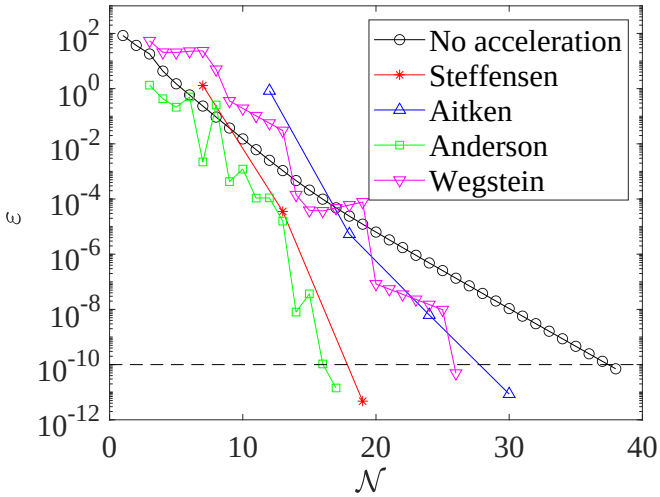


(a) Linear model continuous control

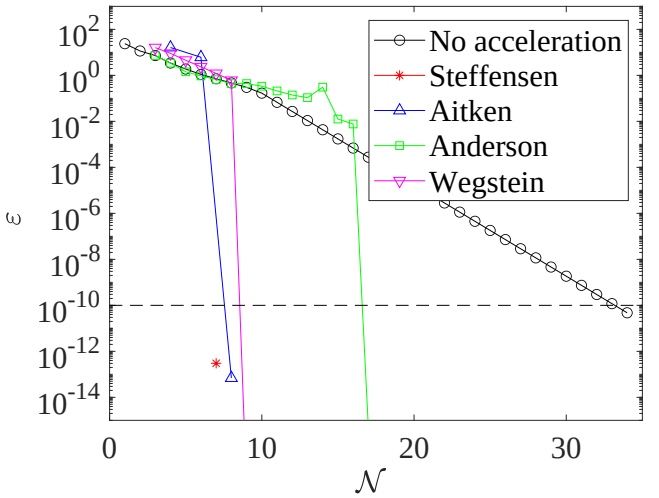


(b) Linear model bang-bang control

Fig. 10. Convergence rates for the result that minimises $\mathcal{N}$, for each acceleration method, when applied to the linear control problems. Results in this plot are produced with model parameter $\gamma = 0.5$, time-step $\mathrm{d}t = 3.91 \times 10^{-3}$, over the interval $0 \leq t \leq 1$, with pay-off weighting $a = b = 1$ for the continuous control (a), and $a = 1$, $b = 3$ for the bang-bang control (b). The tolerance of 1×10^{-10} required for convergence is marked in black dash. As the methods do not necessarily use the same number of function evaluations per iteration, markers indicate each time ε is computed. Continuous control results correspond to the FBSM with no acceleration, the partial Steffensen method with $m = 6$, partial Aitken method with $m = 1$, Anderson Acceleration with $M = 4$ and Wegstein with bounds $-2 \leq q \leq 0$, updating q every 4th iteration. Bang-bang results correspond to the FBSM with no acceleration, the partial Steffensen method with $m = 7$, partial Aitken method with $m = 1$, Anderson Acceleration with $M = 1$ and Wegstein without bounds on q, updating q every iteration. The standard FBSM outperformed all acceleration methods in solving the linear bang-bang control problem. We attribute this to how few iterations were required ($\mathcal{N} = 8$) for convergence without acceleration.



(a) AML model continuous control



(b) AML model bang-bang control

Fig. 11. Convergence rates for the converged result that minimises $\mathcal{N}$, for each acceleration method, when applied to the AML control problems. Results in this plot are produced with model parameter $\gamma = 0.5$, time-step $\mathrm{d}t = 4.88 \times 10^{-4}$, over the interval $0 \leq t \leq 1$, with pay-off weighting $a = b = 1$ for the continuous control (a), and $a = 1$, $b = 3$ for the bang-bang control (b). The tolerance of 1×10^{-10} required for convergence is marked in black dash. As the methods do not necessarily use the same number of function evaluations per iteration, markers indicate each time ε is computed. Continuous control results correspond to the FBSM with no acceleration, $\omega = 0.55$, the partial Steffensen and partial Aitken methods with $m = 5$ and $\omega = 0.5$, Anderson Acceleration with $M = 6$ and $\omega = 0.85$, and Wegstein method with $\omega = 0.55$, bounds $-1 < q < 1$, updating q every 6th iteration. Bang-bang results correspond to the FBSM with no acceleration, $\omega = 0.4$, the partial Steffensen method with $m = 5$ and $\omega = 0.5$, partial Aitken method with $m = 1$ and $\omega = 0.5$, Anderson Acceleration with $M = 7$ and $\omega = 0.35$, and Wegstein method with $\omega = 0$, bounds $-1 < q < 1$, updating q on the 7th iteration.

6 Discussion

Modelling processes in the life sciences is complex; frequently involving large state systems consisting of several ODEs [12,26,74]. The acceleration methods we implement act only on the control term; the number and form of state equations has no bearing on the mathematical and computational complexity of the acceleration methods. As such, the methods scale excellently with system complexity. In this section we discuss the results presented in §5, and draw insights into the convergence behaviour of the FBSM when augmented with acceleration techniques. We highlight opportunities for application of these methods, and outline several avenues for further investigation.

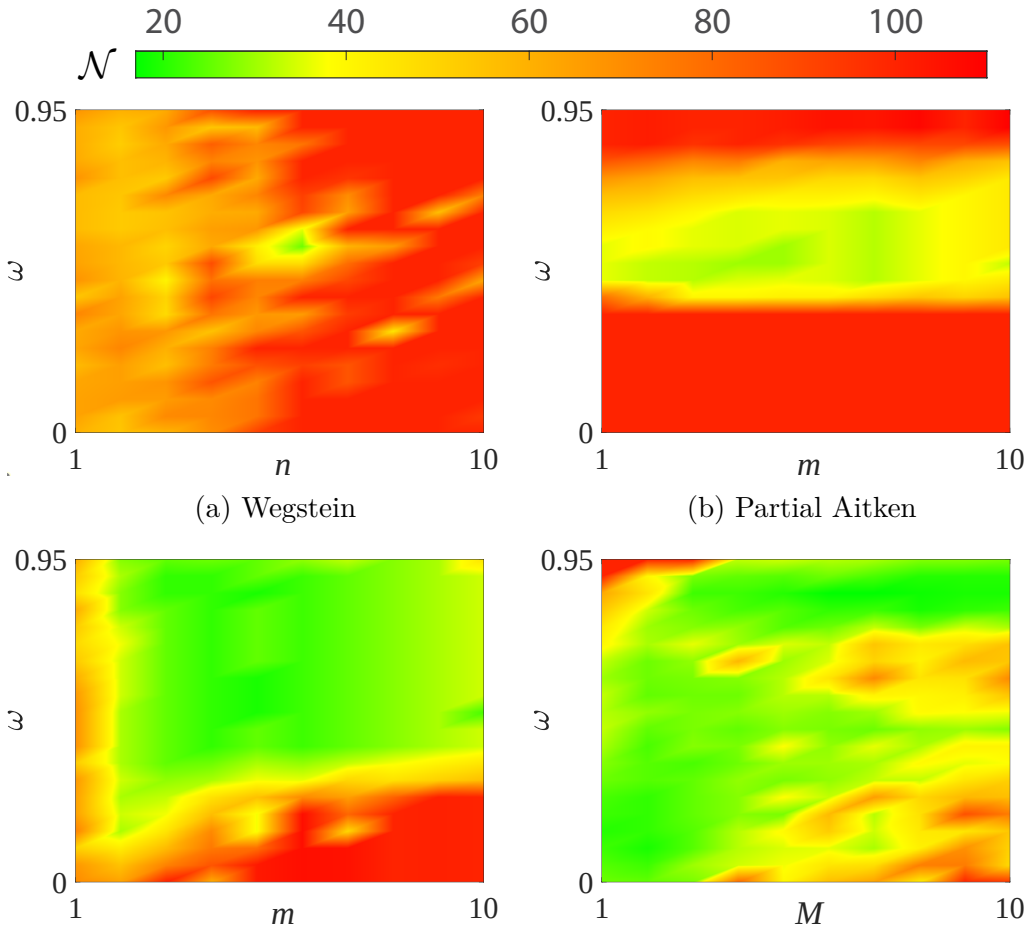
6.1 Acceleration outcomes

In evaluating the performance of each acceleration method, we are interested in: (1) how significantly they are able to reduce $\mathcal{N}$, (2) method robustness, and (3) method accessibility. In this context we use robustness to refer to how consistently the method outperforms the best tuned FBSM over the range of tuning parameters considered. We judge the accessibility of each method based on implementation and conceptual complexity. Overall, we find that the acceleration methods, particularly Anderson and Steffensen, significantly and robustly reduce $\mathcal{N}$. Anderson Acceleration appears most effective for continuous control, while the Steffensen method appears best for bang-bang control. The Aitken method occasionally outperforms Steffensen, but overwhelmingly the Steffensen method appears to be the better option of the two for the range of parameters we consider. Implementing the Anderson and Steffensen methods introduces challenge beyond that of the underlying FBSM, although it is not prohibitively difficult; particularly with reference to the code where we implement these methods, that we make available on GitHub. Both methods

introduce conceptual complexity, perhaps marginally less-so for the Steffensen method due to the similarities it shares with the familiar Newton's method.

We produce heatmaps to visualise the convergence behaviour of the acceleration methods across the range of tuning parameters considered. Figure 12 corresponds to the AML continuous control problem, while Figure 13 corresponds to the AML bang-bang control problem. Recall that with the tuning of ω that minimises $\mathcal{N}$, the FBSM with no acceleration requires $\mathcal{N} = 38$ for the AML continuous control problem, and $\mathcal{N} = 34$ for the AML bang-bang control problem. Tuning parameter combinations that reflect a reduction in $\mathcal{N}$ relative to the these FBSM results are shaded in the green spectrum, while worse performing combinations are shaded in the red spectrum. The midpoint of the colour spectra, yellow, corresponds to the FBSM result with the best tuning, without acceleration. Simulations are terminated when $\mathcal{N}$ exceeds 100; reflecting a combination of tuning parameters that do not yield convergence within this specified maximum. Data supporting these heatmaps, and similar results for the linear control problems are provided in §6 of the supplementary material.

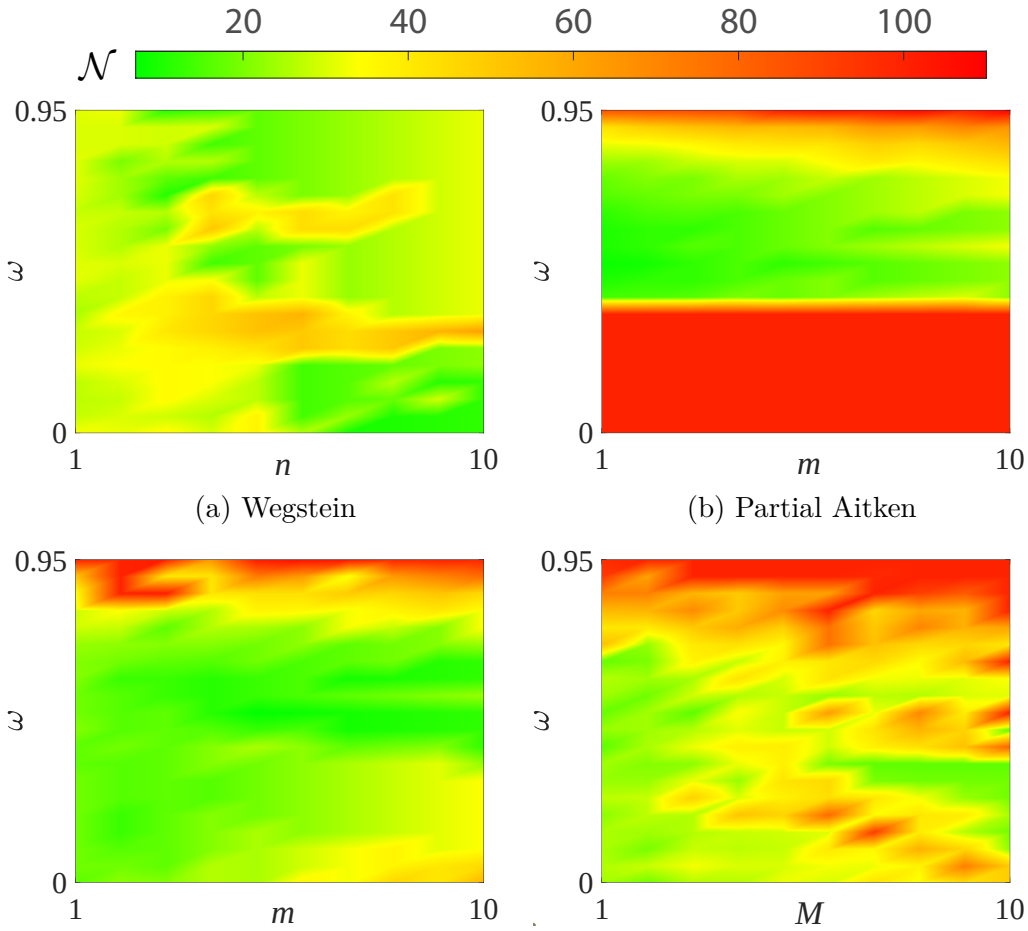
In identifying tuning parameter combinations that yield significant reductions in $\mathcal{N}$, we are looking for bright green areas in the heatmaps. We assess the robustness of each method by considering whether we observe large contiguous areas in the green spectrum, such as in Figure 12(c), indicating robustness, or patchy areas with both green spectrum and red spectrum, such as Figure 13(d), suggesting a lack of robustness.



(a) Wegstein (b) Partial Aitken

(c) Partial Steffensen (d) Anderson

Fig. 12. This heatmap provides insight into the convergence behaviour of the acceleration methods for the AML continuous control problem. Here we visualise $\mathcal{N}$ against ω and the method specific tuning parameter; n for Wegstein (a), m for partial Aitken (b) and partial Steffensen (c), and M for Anderson Acceleration (d). Tuning parameter combinations requiring $\mathcal{N} = 38$, equivalent to the best tuned FBSM without acceleration, are shaded yellow. Colours in the green-yellow spectrum represent a reduction in $\mathcal{N}$ relative to FBSM without acceleration, while colours in the yellow-red spectrum represent an increase in $\mathcal{N}$.



(a) Wegstein (b) Partial Aitken

(c) Partial Steffensen (d) Anderson

Fig. 13. This heatmap provides insight into the convergence behaviour of the acceleration methods for the AML bang-bang control problem. Here we visualise $\mathcal{N}$ against ω and the method specific tuning parameter; n for Wegstein (a), m for partial Aitken (b) and partial Steffensen (c), M for Anderson Acceleration (d). Tuning parameter combinations requiring $\mathcal{N} = 34$, equivalent to the best tuned FBSM without acceleration, are shaded yellow. Colours in the green-yellow spectrum represent a reduction in $\mathcal{N}$ relative to FBSM without acceleration, while colours in the yellow-red spectrum represent an increase in $\mathcal{N}$.

In Table 2 we provide our subjective but informed rating of the methods against the criteria of reduction in $\mathcal{N}$, robustness and accessibility. We consider the continuous and bang-bang control cases separately in terms of reduction in $\mathcal{N}$ and robustness.

Table 2: Method comparison

	Reduction in $\mathcal{N}$		Robustness		Accessibility	
Method	Cts	BB	Cts	BB	Imp	Complexity
FBSM	∼	∼	✓	✓	✓	✓
Wegstein	∼	✓✓	✗	✓	✓	∼
Aitken	✓	✓	∼	✓	∼	∼
Steffensen	✓✓	✓✓	✓	✓✓	∼	∼
Anderson	✓✓	✓	✓	✓	∼	✗

We rate the methods considered in this work against key factors such as the reduction in $\mathcal{N}$ that they deliver and how robustly they perform over the range of tuning parameters considered, for both continuous (Cts) and bang-bang (BB) control problems. We also consider how accessible the methods are from the standpoints of ease of implementation (Imp) and conceptual complexity. Methods are rated as being either strongly positive (✓✓), positive (✓), neutral (∼), negative (✗) against each aspect.

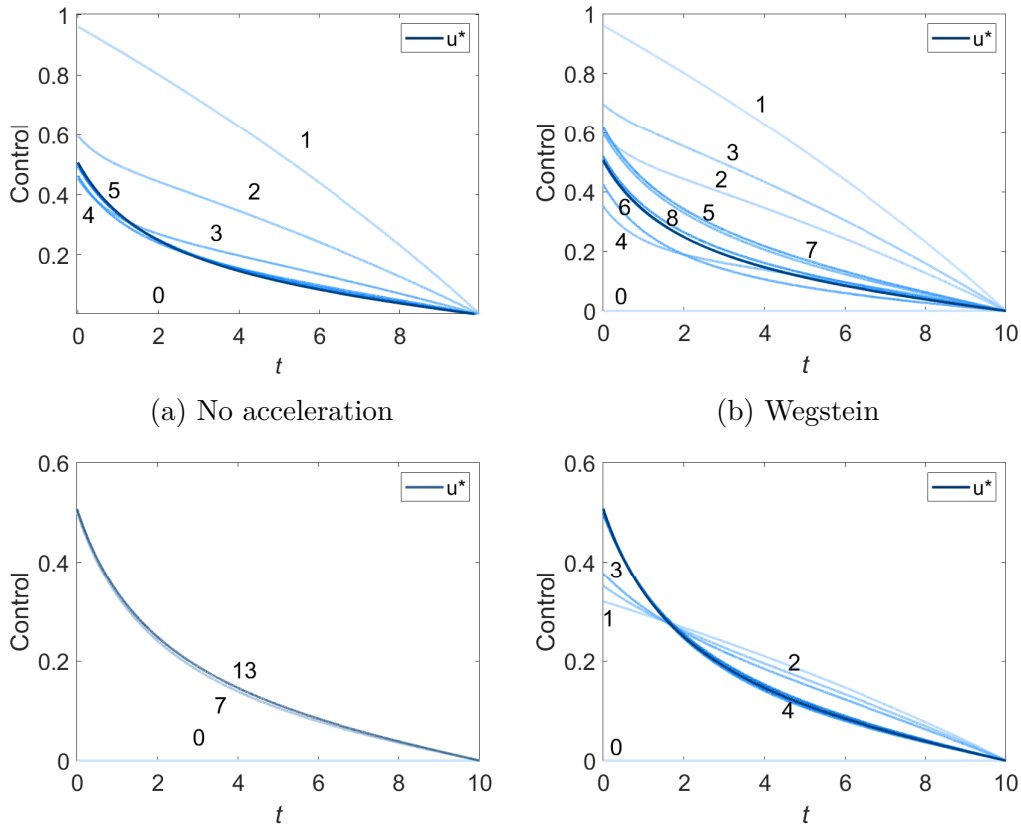
Despite its conceptual simplicity and straightforward implementation, Wegstein's method is significantly hampered by the difficulty in choosing bounds. If there were a more informed approach for identifying suitable bounds, Wegstein's method could be particularly useful for bang-bang control problems. Due to the effect of ω, intermediate control iterates of the FBSM do not appear bang-bang; as such the bulk of $\mathcal{N}$ are incurred in refining the control about the switching points. Wegstein's method can accelerate this refinement by adaptively setting $q_i = 0$ where appropriate.

6.2 Convergence insights

As outlined in §5, the linear model control problems converge with $\omega = 0$. It may at first seem counter-intuitive that Wegstein's method can improve upon this, given that the computed q in Wegstein's method acts as a stand-in for ω. There are two aspects of distinction that enable Wegstein's method to generate improvement in this case; first, while ω is held constant both

within the time discretisation and between iterations, the element-wise nature of Wegstein's method enables each element of the discretisation to have a different value, $q_i, i \in \{0, 1, ..., N\}$, and q can be updated between iterations; secondly, observing the values of q_i in Wegstein's method indicates that $q < 0$ can be appropriate. This suggests that $\omega < 0$ could also be used to accelerate the standard implementation of the FBSM. Preliminary investigation suggests that this is true for the linear model, however we do not pursue this further as we expect it to be of limited applicability beyond contrived problems.

We apply the acceleration methods to small nonlinear test systems in §5 of the supplementary material. We know these systems have multiple fixed points; all methods we consider aside from Aitken's method, in some of our examples, reach different fixed points to fixed point iteration. In contrast, when applied to accelerate control problems, we observe only the Aitken method converging to a result other than the optimal control obtained via the FBSM, as discussed in §5. This apparent convergence of the Aitken method to controls that are not optimal is a significant deterrent to using the Aitken method in situations where the optimal control is not known a priori. Outside of this issue with the Aitken method, each acceleration method produces the same optimal control for a given problem. However, they each approach the converged control differently. In Figure 14 we plot the control as it converges for the FBSM and acceleration methods. In the code we provide on GitHub, users can view the control iterates of each method as they approach convergence. Visualising these methods as they converge gives insight into how they may be able to arrive at different fixed points; under certain circumstances the accelerated series of iterates may leave the basin of attraction for the fixed point found via fixed point iteration.



(a) No acceleration

(b) Wegstein

(c) Partial Steffensen

(d) Anderson

Fig. 14. Here we observe the iterates of the control in the AML continuous control problem as it converges, for (a) the FBSM with no acceleration, (b) the Wegstein method, (c) the partial Steffensen method and (d) Anderson Acceleration. Initial iterates are shown in light blue, while darker blue denotes later iterates. Results for the Aitken method are not shown as they are visually similar to the Steffensen result. We present the results corresponding to the tuning parameters that minimise $\mathcal{N}$, outlined in §5. Where it is visually distinguishable, we indicate the number of function evaluations corresponding to a particular iterate. While all methods produce the same eventual result for u^*, they follow considerably different series of iterates. Note that the vertical scale in (a) and (b) differs from that of (c) and (d).

6.3 Summary and outlook

In this work, we review the theory and implementation of the FBSM for solving TPBVPs that arise from application of PMP in solving optimal control problems. We study a single-variable linear model and a multiple-variable nonlinear model and consider continuous, bang-bang and fixed endpoint control

problems. Conceptualising the FBSM as a fixed point iteration, we leverage and adapt existing acceleration methods to significantly and robustly increase the convergence rate of the FBSM for a range of optimal control problems. The Anderson and partial Steffensen methods appear to perform best, without requiring prohibitive tuning.

Accelerating the convergence of the FBSM, and reducing the importance of appropriately selecting ω for a single control problem, is promising. That said, the real utility of the robust acceleration methods in this work is in application to families of control problems. We provide a glimpse of this benefit through considering fixed endpoint control problems, though other excellent opportunities for application arise due to the uncertainty prevalent in the life sciences. First, it is common for there to be uncertainty around model parameters and structure [27,32]. In this case solving optimal control problems over several model structures and sets of model parameters provides insight into the sensitivity of the control strategy [50,65]. Secondly, when performing multi-objective optimisation, a trade-off is made between objectives. For example, in Equation (21) we seek to minimise the cumulative negative impact of leukaemia and of the control; parameters a_1 and a_2 weight the relative important of each objective. In practical applications, it is not always clear how to determine these weightings. It can therefore be useful to generate a family of optimal controls that are each optimal for their specific combination of pay-off weighting parameters, akin to a Pareto frontier [4,40,46]. Producing these sets of control results benefits significantly from acceleration techniques such as the Anderson and Steffensen methods, where a consistent reduction in $\mathcal{N}$ is obtained without optimal tuning.

Here, we have only considered systems subject to a single control. While this is reflective of the vast majority of applications featured in the control literature, there are instances where we are interested in applying multiple controls simultaneously [9,19,71]. The FBSM can be readily applied to solve problems

with multiple controls [71]; a logical extension of this work is to adapt the acceleration methods or identify suitable alternative methods for accelerating convergence of the FBSM for problems with multiple controls.

Over a range of tuning parameters the Wegstein, Steffensen, and Anderson methods are able to induce convergence where the underlying FBSM fails to converge; such as in the AML control problems with $\omega = 0$. This behaviour has been documented for Anderson acceleration [77] and Wegstein's method [33] when applied to standard fixed point iteration problems. This presents an opportunity for future exploration, in identifying control problems that cannot be solved via the FBSM for any ω, and attempting to produce solutions using these acceleration techniques.

Acknowledgements

JAS acknowledges support from the Australian Government Research Training Program and the AF Pillow Applied Mathematics Trust. JAS and KB acknowledge support from the Australian Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS). MJS is supported by the Australian Research Council (DP200100177).

References

[1] Abbasi Z, Zamani I, Amiri Mehra AH, Shafieirad M, Ibeas A. 2020 Optimal Control Design of Impulsive SQEIAR Epidemic Models with Application to COVID-19. *Chaos, Solitons & Fractals* **139**, 110054. (doi.org/10.1016/j.chaos.2020.110054).

[2] Ackleh AS, Kearfott RB, Allen EJ, Padmanabhan S. 2010 *Classical and Modern Numerical Analysis: Theory, Methods and Practice.* London, United Kingdom:

Chapman & Hall/CRC, Taylor & Francis. (doi.org/10.1201/b12332).

[3] Aitken AC. 1927 On Bernoulli's numerical solution of algebraic equations. *Proceedings of the Royal Society of Edinburgh* **46**, 289–305. (doi.org/10.1017/S0370164600022070).

[4] Alvarez-Vázquez LJ, García-Chan N, Martinez A, Vázquez-Méndez ME. 2010 Multi-objective Pareto-optimal control: an application to wastewater management. *Computational Optimization and Applications* **46**, 135–157. (doi.org/10.1007/s10589-008-9190-9).

[5] Anderson DG. 1965 Iterative procedures for nonlinear integral equations. *Journal of the ACM* **12**, 547–560. (doi.org/10.1145/321296.321305).

[6] Anderson DG. 2019 Comments on "Anderson Acceleration, Mixing and Extrapolation". *Numerical Algorithms* **80**, 135–235. (doi.org/10.1007/s11075-018-0549-4).

[7] Arman A. 1986 *Acceleration algorithms for process design simulations.* Masters thesis, Oklahoma State University.

[8] Aspirin AP, de los Reyes V AA, Yangjin K. 2021 Polytherapeutic strategies with oncolytic virus–bortezomib and adjuvant NK cells in cancer treatment. *Journal of the Royal Society Interface* **18**, 20200669. (doi.org/10.1098/rsif.2020.0669).

[9] Baker CM, Armsworth PR, Lenhart SM. 2017 Handling overheads: optimal multi-method invasive species control. *Theoretical Ecology* **10**, 493–501. (doi.org/10.1007/s12080-017-0344-1).

[10] Bellen A, Vermiglio R. 1996 Some applications of continuous Runge-Kutta methods. *Applied Numerical Mathematics* **22**, 63–80. (doi.org/10.1016/S0168-9274(96)00026-8).

[11] Bellman RE. 1957 *Dynamic Programming.* Princeton, NJ: Princeton University Press.

[12] Benner P, Findeisen R, Flockerzi D, Reichl U, Sundmacher K. 2014 *Large-Scale Networks in Engineering and Life Sciences.* Basel, Switzerland: Birkhäuser Verlag.

[13] Brezinski C, Redivo–Zaglia M. 2019 The genesis and early developments of Aitken's process, Shanks' transformation, the ε-algorithm, and related fixed point methods. *Numerical Algorithms* **80**, 11–133. (doi.org/10.1007/s11075-018-0567-2).

[14] Broyden CG., 1965 A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation* **19**, 577–593. (doi.org/10.1090/S0025-5718-1965-0198670-6).

[15] Bryson AE, Ho YC. 1975 *Applied Optimal Control: Optimization, Estimation, and Control*. Abingdon, United Kingdom: Taylor & Francis.

[16] Bryson AE. 1996 Optimal control – 1950 to 1985. *IEEE Control Systems Magazine* **16**, 26–33. (doi.org/10.1109/37.506395).

[17] Burden RL, Faires DJ. 1985 *Numerical Analysis*. Massachusetts, NE: PWS Publishing Company.

[18] Burnett AK, Goldstone AH, Stevens RMF, Hann IM, Rees JHK, Gray RG, Wheatley K. 2001 Randomised comparison of addition of autologous bone-marrow transplantation to intensive chemotherapy for acute myeloid leukaemia in first remission: results of MRC AML 10 trial. *The Lancet* **351**, 700–708. (doi.org/10.1016/S0140-6736(97)09214-3).

[19] Bussell EH, Cunniffe NJ. 2020 Applying optimal control theory to a spatial simulation model of sudden oak death: ongoing surveillance protects tanoak while conserving biodiversity. *Journal of the Royal Society Interface* **17**, 20190671. (doi.org/10.1098/rsif.2019.0671).

[20] Butcher JC. 2008 *Numerical Methods for Ordinary Differential Equations*. Chichester, United Kingdom: John Wiley & Sons, Ltd.

[21] Carrasco LR, Baker R, MacLeod A, Knight JD, Mumford JD. 2010 Optimal and robust control of invasive alien species spreading in homogeneous landscapes. *Journal of the Royal Society Interface* **7**, 529–540. (doi.org/10.1098/rsif.2009.0266).

[22] Carré BA. 1961 The determination of the optimum accelerating factor for

successive over-relaxation. *The Computer Journal* **4**, 73–78. (doi.org/10.1093/comjnl/4.1.73).

[23] Crowe CM, Nishio M. 1975 Convergence promotion in the simulation of chemical processes - The general dominant eigenvalue method. *American Institute of Chemical Engineers Journal* **21**, 528–533. (doi.org/10.1002/cjce.5450490414).

[24] Crowell HL, MacLean AL, Stumpf MPH. 2016 Feedback mechanisms control coexistence in a stem cell model of acute myeloid leukaemia. *Journal of Theoretical Biology* **401**, 43–53. (doi.org/10.1016/j.jtbi.2016.04.002).

[25] Del Vecchio D, Dy AJ, Qian Y. 2016 Control theory meets synthetic biology. *Journal of the Royal Society Interface* **13**, 20160380. (doi.org/10.1098/rsif.2016.0380).

[26] Domijan M, Rand DA. 2015 Using constraints and their value for optimization of large ODE systems. *Journal of the Royal Society Interface* **12**, 20141303. (doi.org/10.1098/rsif.2014.1303).

[27] Doyle FJ, Stelling J. 2006 Systems interface biology. *Journal of the Royal Society Interface* **3**, 603–616. (doi.org/10.1098/rsif.2006.0143).

[28] Eyert V. 1996 A comparative study on methods for convergence acceleration of iterative vector sequences. *Journal of Computational Physics* **124**, 271–285. (doi.org/10.1006/jcph.1996.0059).

[29] Falisse A, Serrancolí G, Dembia CL, Gillis J, Jonkers I, De Groote F. 2019 Rapid predictive simulations with complex musculoskeletal models suggest that diverse healthy and pathological human gaits can emerge from similar control strategies. *Journal of the Royal Society Interface* **16**, 20190402. (doi.org/10.1098/rsif.2019.0402).

[30] Fang H, Saad Y. 2008 Two classes of multisecant methods for nonlinear acceleration. *Numerical Linear Algebra with Applications* **16**, 197–221. (doi.org/10.1002/nla.617).

[31] Goldstine HH. 1980 *A History of the Calculus of Variations from the 17th through to the 19th Century*. New York, NY: Springer-Verlag.

[32] Gutenkunst RN, Waterfall JJ, Casey FP, Brown KS, Myers CR, Sethna JP. 2007 Universally sloppy parameter sensitivities in systems biology models. *PLoS Computational Biology* **3**, e189. (doi.org/10.1371/journal.pcbi.0030189).

[33] Gutzler CH. 1959 *An iterative method of Wegstein for solving simultaneous nonlinear equations*. Masters thesis, Oregon State College.

[34] Hackbush W. 1978 A numerical method for solving parabolic equations with opposite orientations. *Computing* **20**, 229–240. (doi.org/10.1007/BF02251947).

[35] Hadjidimos A. 2000 Successive overrelaxation (SOR) and related methods. *Journal of Computational and Applied Mathematics* **123**, 177–199. (doi.org/10.1016/S0377-0427(00)00403-9).

[36] He F, Murabito E, Westerhoff HV. 2016 Synthetic biology and regulatory networks: where metabolic systems biology meets control engineering. *Journal of the Royal Society Interface* **13**, 20151046. (doi.org/10.1098/rsif.2015.1046).

[37] Henrici P. 1964 *Elements of Numerical Analysis*. New York, NY: John Wiley & Sons.

[38] Ishikawa F, Yoshida S, Saito Y, Hijikata A, Kitamura H, Tanaka S, Nakamura R, Tanaka T, Tomiyama H, Saito N, Fukata M, Miyamoto T, Lyons B, Ohshima K, Uchida N, Taniguchi S, Ohara O, Akashi K, Harada M, Shultz LD. 2007 Chemotherapy-resistant human AML stem cells home to and engraft within the bone-marrow endosteal region. *Nature Biotechnology* **25**, 1315–1321. (doi.org/10.1038/nbt1350).

[39] Kapitula T. 2015 *Ordinary Differential Equations and Linear Algebra: A Systems Approach*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

[40] Keane AJ, Scanlan JP. 2007. Design search and optimization in aerospace engineering. *Philosophical Transactions of the Royal Society A* **365**, 2501–2529. (doi.org/10.1098/rsta.2007.2019).

[41] Kulsrud HE. 1961 A practical technique for the determination of the optimum

relaxation factor of the successive over-relaxation method. *Communications of the ACM* **4**, 184–187. (doi.org/10.1145/355578.366504).

[42] Lai AKM, Dick TJM, Biewener AA, Wakeling JM. 2021 Task-dependent recruitment across ankle extensor muscles and between mechanical demands is driven by the metabolic cost of muscle contraction. *Journal of the Royal Society Interface* **18**, 20200765. (doi.org/10.1098/rsif.2020.0765).

[43] Lenhart S, Workman JT. 2007 *Optimal Control Applied to Biological Models.* London, United Kingdom: Chapman & Hall/CRC, Taylor & Francis.

[44] Li Q, Chen L, Tai C, E W. 2018 Maximum principle based algorithms for Deep Learning. *Journal of Machine Learning Research* **18**, 1–29. (jmlr.org/papers/v18/17-653).

[45] Liu X, Frank J. 2021 Symplectic Runge–Kutta discretization of a regularized forward–backward sweep iteration for optimal control problems. *Journal of Computational and Applied Mathematics* **383**, 113113. (doi.org/10.1016/j.cam.2020.113133).

[46] Logist F, Houska B, Diehl M, Van Impe JF. 2011 Robust multi-objective optimal control of uncertain (bio)chemical processes. *Chemical Engineering Science* **66**, 4670–4682. (doi.org/10.1016/j.ces.2011.06.018).

[47] MacLean AL, Celso CL, Stumpf MPH. 2013 Population dynamics of normal and leukaemia stem cells in the haematopoietic stem cell niche show distinct regimes where leukaemia will be controlled. *Journal of the Royal Society Interface* **10**, 20120968. (doi.org/10.1098/rsif.2012.0968).

[48] Manzoor H, Selam MA, Rahman FBA, Adham S, Castier M, Abdel-Wahab A. 2020 A tool for assessing the scalability of pressure-retarded osmosis (PRO) membranes. *Renewable Energy* **149**, 987–999. (doi.org/10.1016/j.renene.2019.10.098).

[49] McAsey M, Mou L, Han W. 2012 Convergence of the forward-backward sweep method in optimal control. *Computational Optimization and Applications* **43**, 207–226. (doi.org/10.1007/s10589-011-9454-7).

[50] Mwanga GG, Haario H, Capasso V. 2015 Optimal control problems of epidemic systems with parameter uncertainties: Application to a malaria two-age-classes transmission model with asymptomatic carriers. *Mathematical Biosciences* **261**, 1–12. (doi.org/10.1016/j.mbs.2014.11.005).

[51] Neilan RLM, Schaefer E, Gaff H, Fister KR, Lenhart S. 2010 Modeling optimal intervention strategies for cholera. *Bulletin of Mathematical Biology* **72**, 2004–2018. (doi.org/10.1007/s11538-010-9521-8).

[52] Nievergelt Y. 1991 Aitken's and Steffensen's accelerations in several variables. *Numerische Mathematik* **59**, 295–310. (doi.org/10.1007/BF01385782).

[53] Nievergelt Y. 1995 The condition of Steffensen's acceleration in several variables. *Journal of Computational and Applied Mathematics* **58**, 291–305. (doi.org/10.1016/0377-0427(94)00004-K).

[54] Noda T. 1984 The Steffensen iteration method for systems of nonlinear equations. *Proceedings of the Japan Academy* **60**, 18–21. (doi.org/10.3792/pjaa.60.18).

[55] Noda T. 1986 The Aitken-Steffensen formula for systems of nonlinear equations. III. *Proceedings of the Japan Academy* **62**, 174–177. (doi.org/10.3792/PJAA.62.174).

[56] Noda T. 1987 The Steffensen iteration method for systems of nonlinear equations. II. *Proceedings of the Japan Academy* **63**, 186–189. (doi.org/10.3792/pjaa.63.186).

[57] Orbach O, Crowe CM. 1971 Convergence promotion in the simulation of chemical processes with recycle – the dominant eigenvalue method. *Canadian Journal of Chemical Engineering* **49**, 509–513. (doi.org/10.1002/cjce.5450490414).

[58] Patil NS, Dingwell JB, Cusumano JP. 2020 Task-level regulation enhances global stability of the simplest dynamic walker. *Journal of the Royal Society Interface* **17**, 20200278. (doi.org/10.1098/rsif.2020.0278).

[59] Penrose R. 1955 A generalized inverse for matrices. *Mathematical Proceedings*

of the Cambridge Philosophical Society **51**, 406–413. (doi.org/10.1017/S0305004100030401).

[60] Polak E. 1973 An historical survey of computational methods in optimal control. *SIAM Review* **15**, 553–584. (doi.org/10.1137/1015071).

[61] Pontryagin LS, Boltyanskii VG, Gamkrelidze RV, Mischenko EF. 1962 *The Mathematical Theory of Optimal Processes [English translation]*. New York, NY: Interscience.

[62] Press WH. 2007 *Numerical recipes: the art of scientific computing*. New York, NY: Cambridge University Press.

[63] Ramière I, Helfer T. 2015 Iterative residual-based vector methods to accelerate fixed point iterations. *Computers and Mathematics with Applications* **70**, 2210 – 2226. (doi.org/10.1016/j.camwa.2015.08.025).

[64] Rao AV. 2009 A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences* **135**, 497–528.

[65] Rawson T, Wilkins KE, Bonsall MB. 2020 Optimal control approaches for combining medicines and mosquito control in tackling dengue. *Royal Society Open Science* **7**, 181843. (doi.org/10.1098/rsos.181843).

[66] Reid JK. 1966 A method for finding the optimum successive over-relaxation parameter. *The Computer Journal* **9**, 200–204. (doi.org/10.1093/comjnl/9.2.200).

[67] Rosen EM, Pauls AC. 1977 Computer aided chemical process design: the FLOWTRAN system. *Computers and Chemical Engineering* **1**, 11–21. (doi.org/10.1016/0098-1354(77)80003-3).

[68] Sargent RWH. 2000 Optimal control. *Journal of Computational and Applied Mathematics* **124**, 361–371. (doi.org/10.1016/S0377-0427(00)00418-0).

[69] Sethi PS. 2019 *Optimal Control Theory – Applications to Management Science and Economics, Third Edition*. Cham, Switzerland: Springer.

[70] Sharp JA, Browning AP, Mapder T, Burrage K, Simpson MJ. 2019 Optimal control of acute myeloid leukaemia. *Journal of Theoretical Biology* **470**, 30–42. (doi.org/10.1016/j.jtbi.2019.03.006).

[71] Sharp JA, Browning AP, Mapder T, Baker CM, Burrage K, Simpson MJ. 2020 Designing combination therapies using multiple optimal controls. *Journal of Theoretical Biology* **497**, 110277. (doi.org/10.1016/j.jtbi.2020.110277).

[72] Sipkins DA, Wei X, Wu JW, Runnels JM, Côtè D, Means TK, Luster AD, Scadden DT, Lin CP. 2005 In vivo imaging of specialized bone marrow endothelial microdomains for tumour engraftment. *Nature* **435**, 969–973. (doi.org/10.1038/nature03703).

[73] Skorych V, Dosta M, Heinrich S. 2020 Dyssol - An open-source flowsheet simulation framework for particulate materials. *SoftwareX* **12**, 100572. (doi.org/10.1016/j.softx.2020.100572).

[74] Snowden TJ, van der Graaf PH, Tindall MJ. 2017 Methods of model reduction for large-scale biological systems: a survey of current methods and trends. *Bulletin of Mathematical Biology* **79**, 1449–1486. (doi.org/10.1007/s11538-017-0277-2).

[75] Steffensen JF. 1933 Remarks on iteration. *Scandinavian Actuarial Journal* **1**, 64–72. (doi.org/10.1080/03461238.1933.10419209).

[76] Varga RS. 2000 *Matrix Iterative Analysis, Second Revised and Expanded Edition.* Heidelberg, Germany: Springer-Verlag.

[77] Walker HF, Ni P. 2011 Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis* **49**, 1715–1735. (doi.org/10.1137/10078356X).

[78] Wegstein JH. 1958 Accelerating convergence of iterative processes. *Communications of the ACM* **1**, 9–13. (doi.org/10.1145/368861.368871).

[79] Young DM. 1971 *Iterative Solutions of Large Linear Systems.* Orlando, FL: Academic Press, Inc.

[80] Zennaro M. 1986 Natural continuous extensions of Runge-Kutta methods. *Mathematics of Computation* **46**, 119–133. (doi.org/10.2307/2008218).