DIMSEMs — Diagonally Implicit Single-Eigenvalue Methods for the Numerical Solution of Stiff Ordinary Differential Equations on Parallel Computers

by

Robert Frederick Enenkel

A thesis submitted in conformity with the requirements

for the degree of Doctor of Philosophy

Graduate Department of Computer Science

University of Toronto

© Copyright by Robert Frederick Enenkel 1996

Abstract

This thesis derives a new class of general linear methods (GLMs) intended for the solution of stiff ordinary differential equations (ODEs) on parallel computers. Although GLMs were introduced by Butcher in the 1960s, the task of deriving formulas from the class with properties suitable for specific applications is far from complete. This thesis is a contribution to that work.

Our new methods have several properties suited for the solution of stiff ODEs on parallel computers. They are strictly diagonally implicit, allowing parallelism in the Newton iteration used to solve the nonlinear equations arising from the implicitness of the formula. The stability matrix has no spurious eigenvalues (that is, only one eigenvalue of the stability matrix is non-zero), resulting in a solution free from contamination from spurious solutions corresponding to non-dominant, non-zero eigenvalues. From these two properties arises the name DIMSEM, for Diagonally IMplicit Single-Eigenvalue Method. The methods have high stage order, avoiding the phenomenon of order reduction that occurs, for example, with some Runge-Kutta methods. The methods are L-stable, with the result that the chosen stepsize is dictated by convergence requirements rather than stability considerations imposed by the stiffness of the problem.

An introduction to GLMs is given and some order barriers for DIMSEMs are presented. DIMSEMs of orders 2–6 are derived, as well as an L-stable class of diagonal methods of all orders which do not, however, possess the single-eigenvalue property. A fixed-order, variable-stepsize implementation of the DIMSEMs is described, including the derivation of local error estimators, and the results of testing on both sequential and parallel computers is presented. The testing shows the DIMSEMs to be competitive with fixed-order versions of the popular solver LSODE on a practical test problem.

Acknowledgments

I wish to thank my research supervisor, Professor Ken Jackson; the members of my thesis committee: Professors Christina Christara, Wayne Enright, Allan Jepson and Rudi Mathon; and Professor Stuart Smith of the Department of Mathematics for serving as the internal examiner.

Thanks to Professor John Butcher of the University of Auckland, New Zealand, out of whose work the topic of this thesis emerged, for coming to Toronto to serve as the external examiner.

Thanks also to Professors Kevin Burrage, Phillipe Chartier, and Zdzislaw Jackiewicz for their discussion of my research.

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Information Technology Research Center of Ontario (ITRC).

Contents

1 Introduction and Background **1**

1.1 General Linear Methods 1

1.2 Parallelism in Implicit GLMs 3

1.3 Error Coefficients 5

1.4 Order of Accuracy and Simplifying Assumptions 7

1.5 Stability Regions of GLMs 8

1.6 Overview of the Thesis 9

2 Development of Diagonal Methods **11**

2.1 How We Came to DIMSEMs 11

2.1.1 DIMSIMs — Diagonally Implicit Multi-Stage Integration Methods 12

2.1.2 Application of the Single-Eigenvalue Property to Proving A- and L-Stability 13

2.2 Maximal Order of Single-Eigenvalue GLMs 15

2.3 Singly Implicit DIMSEMs and DIMSEM-MRKs — Some Negative Results . . . 23

2.4 Linking Stability and Order Conditions 26

2.5 A Class of L-stable Diagonal Methods (LDMs) 28

2.6 A Class of DIMSEMs 29

2.7 A Class of L-Stable DIMSEMs 34

2.7.1 Choice of Free Parameters for the DIMSEMs 35

2.8 Choice of Free Parameters for the LDMs 37

2.9 Fixed-Stepsize Numerical Testing 41

3 Variable-Stepsize Implementation and Testing **43**

3.1 Stepsize Change Scheme 43

3.2 Stability for Variable-Stepsize Methods 45

3.3 Local Error Estimators 54

3.3.1 DIMSEM Principal Error Coefficients . . . 55
3.3.2 Difference-Type Local Error Estimate . . . 59
3.3.3 Reference Solution-Type Local Error Estimates . . . 62
3.3.4 Essential Local Discretisation Error . . . 63
3.3.5 Testing and Selection of Local Error Estimators . . . 64
3.4 Preliminary Testing of Variable-Stepsize Method . . . 65
3.5 Sequential Variable-Stepsize Testing . . . 67
3.6 Variable-Stepsize Implementation Details . . . 76
3.7 Parallel Performance Testing . . . 82
3.7.1 Parallel Machine . . . 82
3.7.2 Test Problem . . . 83
3.7.3 DIMSEM Speedup Testing . . . 84
3.7.4 Comparison of Parallel DIMSEMs with Fixed-Order LSODE . . . 86

4 Conclusions and Future Work **89**

A Theorem 4 **91**

B Theorem 5 **99**

C Theorem 8 **154**

D Theorem 9 **156**

E Lemmas **159**
E.1 Lemma 2 . . . 159
E.2 Lemma 3 . . . 172
E.3 Lemma 4 . . . 174
E.4 Lemma 5 . . . 174
E.5 Lemma 6 . . . 176
E.6 Lemma 7 . . . 178
E.7 Lemma 8 . . . 178
E.8 Lemma 9 . . . 180
E.9 Lemma 10 . . . 181
E.10 Lemma 11 . . . 182
E.11 Lemma 12 . . . 182
E.12 Lemma 13 . . . 184

E.13 Lemma 14 185
E.14 Lemma 15 187

F Combinatorial Identities **195**

Chapter 1

Introduction and Background

In this thesis we derive numerical methods for the parallel solution of stiff initial-value problems (IVPs) for ordinary differential equations (ODEs) of the form

$$\begin{aligned} u'(x) &= f(u(x)) \quad \forall x \in [x_0, x_{end}], \\ u(x_0) &= u_0, \end{aligned}$$

where $u : \mathbb{R} \to \mathbb{R}^K$. For notational convenience we assume a scalar equation ($K = 1$) unless otherwise noted, but the formulas generalise easily to vectors with the use of tensor product notation. Methods for solving IVPs typically compute approximations to $u(x_n), n = 1, \ldots, N$, on some mesh $x_0 < x_1 < \cdots < x_N = x_{end}$, where the approximation at each successive step depends on one or more of the approximations at previous steps. We seek methods which permit parallel computation within each step of the solver, rather than attempting to replace the entire sequential step-by-step integration with a parallel process. Our methods will therefore be suitable for machines with a few processors, such as shared-memory machines, rather than the hundreds or thousands of processors available with local-memory, message-passing architectures.

1.1 General Linear Methods

The methods developed here are members of the class of general linear methods (GLMs) introduced by Butcher. (See [9] for a survey.) They are also known as multivalue methods (MVMs) [5]. Given a vector y_n of external stages on the current step, they first compute a

vector of internal stages,

$$Y_n = A_1 y_n + h B_1 F(Y_n) \in \mathbb{R}^s,$$

and then compute a new vector y_{n+1} of external stages from it by

$$y_{n+1} = A_2 y_n + h B_2 F(Y_n) \in \mathbb{R}^r.$$

Here h is a fixed stepsize (extension to a variable stepsize will be considered later), $F(Y_n)$ is the vector formed by applying f to each component of Y_n, and A_1, B_1, A_2 and B_2 are matrices of constant coefficients which determine the method. The components of the external stages y_n may in different methods represent approximations to the solution at various points, derivatives of the solution, or other quantities. The internal stages are approximations to the solution at points determined by the internal abscissae $c_1, \ldots, c_s$,

$$Y_{n,i} \approx u(x_n + c_i h), \quad i = 1, \ldots, s.$$

A starting method is required to provide the initial approximation y_0, and, if the solution itself is not a component of y_n, a finishing method is needed to recover it [4, 9].

Two common types of GLM are Nordsieck methods, in which the external stages approximate scaled derivatives of the solution of the ODE, and multistep-type GLMs, in which they approximate the solution itself at various points. That is, for a Nordsieck GLM,

$$y_{n,i} \approx \frac{h^{i-1}}{(i-1)!} u^{(i-1)}(x_n), \quad i = 1, \ldots, r, \tag{1.1}$$

while for a multistep-type GLM,

$$y_{n,i} \approx u(x_n + d_i h), \quad i = 1, \ldots, r. \tag{1.2}$$

The d_i's are called external abscissae. A special type of multistep-type GLM is the Multistep Runge-Kutta (MRK) method, in which $d_i = i - r$, $i = 1, \ldots, r$,

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ \alpha_1 & & \cdots & & \alpha_r \end{bmatrix}, \tag{1.3}$$

and

$$B_2 = \begin{bmatrix} 0 & \cdots & 0 \\ & \vdots & \\ 0 & \cdots & 0 \\ \gamma_1 & \cdots & \gamma_s \end{bmatrix}, \tag{1.4}$$

for some α_i's and γ_i's.

The representation of GLMs by their coefficient matrices A_1, B_1, A_2, and B_2, is in some sense not unique. Two different methods may be equivalent in that their solution vectors y_n are related by a linear transformation. In particular, any Nordsieck GLM has an equivalent multistep-type GLM in this sense, and vice-versa.

A GLM is called explicit if B_1 is strictly lower triangular, that is, lower triangular with zeroes on the diagonal. Otherwise it is implicit. Although the computation of the internal stages Y_n is considerably less expensive in an explicit method, only implicit methods are suitable for the solution of stiff problems (Theorem 8). In an implicit method, computing Y_n at each step requires the solution of a system of generally nonlinear equations,

$$Y_n = A_1 y_n + h B_1 F(Y_n).$$

This is typically performed using a Newton iteration,

$$\begin{aligned} (I_s - h B_1 F'(Y_n^{(k)}))\Delta_n^{(k)} &= -R_n(Y_n^{(k)}) \\ Y_n^{(k+1)} &= Y_n^{(k)} + \Delta_n^{(k)}, \end{aligned} \tag{1.5}$$

(or some variant in which $F'(Y_n^{(k)})$ is replaced by an approximation) where

$$R_n(Y) = Y - A_1 y_n - h B_1 F(Y)$$

is the function whose zero is the internal stage vector Y_n, and $Y_n^{(k)}$ is the k-th iterate approximating Y_n.

1.2 Parallelism in Implicit GLMs

The parallelism inherent in an implicit GLM depends on the structure of B_1. (Note that, in the following, the names given to GLMs of various structures vary with different authors.)

A *diagonally implicit* GLM has

$$B_1 = \begin{bmatrix} \lambda_1 & & & & \\ b_{21} & \lambda_2 & & & \\ b_{31} & b_{32} & \lambda_3 & & \\ \vdots & \vdots & & \ddots & \\ b_{s1} & b_{s2} & \cdots & b_{s,s-1} & \lambda_s \end{bmatrix}.$$

With this structure, each internal stage value $Y_{n,i}$ is coupled only to those before it, so that the $Y_{n,i}$'s may be solved for in sequence. If $\lambda_1 = \cdots = \lambda_s$, the method is called *singly diagonally implicit*. Such methods have been investigated, for instance, by Burrage and Chipman [6], for use on sequential machines.

In a *strictly diagonally implicit* GLM, B_1 is a diagonal matrix,

$$B_1 = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_s \end{bmatrix}.$$

This decouples the component equations of the Newton iteration (1.5) into

$$(1 - h\lambda_i f'(Y_{n,i}^{(k)}))\Delta_{n,i}^{(k)} = -R_n(Y_n^{(k)})_i, \quad i = 1, \ldots, s, \tag{1.6}$$

with the effect that $\Delta_{n,i}^{(k)}$, $i = 1, \ldots, s$, can be solved for independently. For a K-dimensional IVP, (1.6) becomes

$$(I_K - h\lambda_i f'(Y_{n,i}^{(k)}))\Delta_{n,i}^{(k)} = -R_n(Y_n^{(k)})_i, \quad i = 1, \ldots, s, \tag{1.7}$$

with $\Delta_{n,i}^{(k)} \in \mathbb{R}^K$, and a separate factorisation of

$$(I_K - h\lambda_i f'(Y_{n,i}^{(k)}))$$

is required for each distinct λ_i. (Or, some other scheme must be used to accommodate the distinct λ_i's, such as Enright's factorisation [17] or an iterative linear solver.) The s factorisations however can be done in parallel, as can the s solves for $\Delta_{n,i}^{(k)}$, $i = 1, \ldots, s$.

If we restrict B_1 to be a multiple of the identity, $B_1 = \lambda I$, the result is a *singly strictly*

diagonally implicit GLM. For a K-dimensional problem, (1.7) becomes

$$(I_K - h\lambda f'(Y_{n,i}^{(k)}))\Delta_{n,i}^{(k)} = -R_n(Y_n^{(k)})_i, \quad i = 1, \ldots, s,$$

where $\Delta_{n,i}^{(k)} \in \mathbb{R}^K$. On a sequential machine, it is common to approximate each $(I_K - h\lambda f'(Y_{n,i}^{(k)}))$, $i = 1, \ldots, s$, by a single matrix $N_n^{(k)}$, to avoid having to perform s factorisations per iteration. The constant diagonal λ in a singly implicit method is therefore an advantage on a sequential machine. On a parallel machine, however, each factorisation, as well as each of the subsequent solves, can be assigned to a different processor, so there is less incentive to limit the structure of B_1 in this way.

Butcher recently introduced a class of methods called DIMSIMs [11, 12, 13, 14] which contains methods with each of the above structures. Because of their particular relevance to our work, these methods will be discussed separately later.

1.3 Error Coefficients

An important property of a method is the accuracy of the generated solution, and the dependence of this accuracy on the stepsize. This leads to the concepts of local and global discretisation error, error coefficients and order of accuracy. Local discretisation error is the error in the solution produced at the end of a single step started with exact information, while global discretisation error (global error for short) is the error after a sequence of steps, only the first of which is started with exact information. In our formulation, error coefficients are a problem-independent characterisation of the local discretisation error.

A valuable tool for the analysis of numerical methods for ODEs is the series expansion known as a Butcher series (or B-series),

$$B(\varphi, u) = \sum_{t \in T} \varphi(t) \frac{h^{\rho(t)}}{\rho(t)!} D(t)(u). \tag{1.8}$$

Here T is the set of monotonically labeled rooted trees, $\varphi : T \to \mathbb{R}^l$ for some l (in our application l is either r or s), $\rho(t)$ is the order of tree t (that is, the number of nodes in t), and $D(t)$ is the elementary differential associated with the ODE for the tree t. Let ϕ be the empty tree, which is the tree with no nodes, and let $\bullet$ be the tree with only a root node. Let $t = [t_1, \ldots, t_m]$ represent the tree formed by joining each of the trees $t_1, \ldots, t_m$ with a new branch from each of their respective root nodes to a new node, which becomes the root of

t. Then for the ODE $u'(x) = f(u(x))$, $D(t)$ is defined recursively by

$$D(\phi)(u) = u,$$

$$D(\bullet)(u) = f(u),$$

and

$$D(t)(u) = f^{(k)}(u)\,(D(t_1)(u)\cdots D(t_k)(u)) \quad \text{for} \quad t = [t_1, \ldots, t_k].$$

If $K > 1$, the multiplication of vectors is done componentwise. More on Butcher series may be found in [4, 5, 10, 20, 21, 22, 24]. As an example, if $\varphi(t) = c^{\rho(t)}$, then $B(\varphi, u(x_n))$ is the Taylor series for $u(x_n + ch)$ expanded about the point x_n.

We now summarise the result of using Butcher series to analyse the local discretisation error in a GLM stepping from y_n to y_{n+1}. Let $\hat{y}_k$ be the vector of exact values to which the external stage vector y_k is an approximation, for every step k. By appropriate choice of what is known as the true-value function, $q : T \to \mathbb{R}^r$, we can expand $\hat{y}_k$ in a Butcher series around x_k,

$$\hat{y}_k = B(q, u(x_k)) \quad \forall k, \tag{1.9}$$

where $u : \mathbb{R} \to \mathbb{R}$ is the true solution of the IVP. Let $y_n = \hat{y}_n$, so that the method starts the n-th step with exact information. The local discretisation error l_n is then defined by

$$l_n = y_{n+1} - \hat{y}_{n+1} \tag{1.10}$$

$$= B(e, u(x_n)) \tag{1.11}$$

for an appropriate choice of $e : T \to \mathbb{R}^r$, where $e(t)$ is the vector of (local discretisation) error coefficients of the method for the tree t. Using the theorem of Hairer and Wanner [21] on the composition of Butcher series, we can express $e(t)$ recursively in terms of error coefficients of trees of lower order, the true-value function q, and the coefficient matrices A_1, B_1, A_2, B_2 of the method. (More details may be found in the proof of Theorem 7 in Section 3.3.1.) The resulting expressions are the basis for the computer programs we wrote to find error coefficients for our methods.

1.4 Order of Accuracy and Simplifying Assumptions

The order of a method is a characterisation of the sensitivity of the error to changes in stepsize. The global error is defined as

$$g_n = y_{n+1} - \hat{y}_{n+1},$$

where

$$y_0 = \hat{y}_0.$$

A method is said to have global order p (or, simply, order p) if p is the largest value such that $g_n = O(h^p)$ for all n and all sufficiently differentiable problems. Similarly, a method is said to have local order $p+1$ if p is the largest value such that $l_n = O(h^{p+1})$ for all n and all sufficiently differentiable problems. For a stable method, a global error of order p usually corresponds to a local discretisation error of order $p+1$, although it is possible in certain situations to have local and global discretisation errors of the same order [16, 29].

The equations which must be satisfied for a method to have a certain local order are called the order conditions. For a method of local order $p+1$, we require that $e(t) = \mathbf{0}$ for all trees t of order $\rho(t) \leq p$, where $e(t)$ are the error coefficients. This results, for most methods of interest, in global order at least p, provided the method is stable. Solving these order conditions directly for methods of order greater than four is extremely difficult, however, as they consist of large systems of nonlinear equations. For this reason it is customary to employ simplifying assumptions, which are sufficient, but not necessary, conditions for the order conditions to hold.

The methods developed in this thesis make use of the A, B and C simplifying assumptions of Burrage and Moss [7]. These are defined as

$$A(w) \quad : \quad q(t) = q(\tau^{\rho(t)}), \quad \forall t \ni \rho(t) \leq w, \tag{1.12}$$

$$B(w) \quad : \quad A_2 q(\tau^k) + k B_2 c^{k-1} = \sum_{i=0}^{k} \left(\begin{array}{c} k \\ i \end{array} \right) q(\tau^i), \quad k = 0, \ldots, w, \tag{1.13}$$

$$C(w) \quad : \quad A_1 q(\tau^k) + k B_1 c^{k-1} = c^k, \quad k = 0, \ldots, w, \tag{1.14}$$

where kB_1c^{k-1} and kB_2c^{k-1} are taken as $\mathbf{0}$ when $k = 0$, and $\tau^k = [\bullet, \ldots, \bullet]$ is what is known as the *bushy* tree of order k. The importance of these simplifying assumptions is given by a theorem of Burrage and Moss [7], which states that a GLM satisfying $A(p)$, $B(p)$, and $C(p-1)$ has order at least p, provided the method is stable. Furthermore, the stage

order (the order of the internal stages) is at least $p-1$, and if $C(p)$ is satisfied, it is at least p. The solution of these simplified order conditions, while by no means trivial, is much easier than that of the raw order conditions described earlier. More on simplifying assumptions for GLMs may be found in [3, 4, 7].

1.5 Stability Regions of GLMs

When a GLM is used with a sufficiently large fixed stepsize, the computed solution may become unstable, that is, may grow without bound, even though the true solution remains bounded or decays. For certain methods applied to so-called *stiff* problems, the stepsize necessary for stability may be much smaller than that necessary to produce a sufficiently accurate solution with a more appropriate method. An important indication of the suitability of a method is the size of its stability region, which we define for GLMs as follows:

Definition 1 *The stability region R of a GLM is given by*

$$\begin{aligned} R \;=\; \{h\mu : \exists c \in \mathbb{R} \ni & \textit{ the GLM applied with stepsize } h \textit{ to the test problem} \\ & u' = \mu u, \mu \in \mathbb{C}, \textit{ produces a solution } y_n \textit{ satisfying } \|y_n\| \le c\|y_0\| \ \forall n\}. \end{aligned}$$

This definition is somewhat stronger than an alternative formulation in which the external stage vector y_n is replaced by u_n, the approximation of $u(x_n)$ to which y_n corresponds. However, a method with bounded $\|u_n\|$'s and unbounded $\|y_n\|$'s is in some sense degenerate, and can likely be re-written without the use of any unbounded $y_{n,i}$'s.

Definition 1 can be shown to be equivalent to

$$R = \{z \in \mathbb{C} : \|M(z)^n\| \text{ is bounded } \forall n \ge 1\},$$

where

Definition 2 *The stability matrix $M(z)$ of a GLM is given by*

$$M(z) = A_2 + zB_2(I - zB_1)^{-1}A_1.$$

A third equivalent definition of R is

$$\begin{aligned} R \;=\; \{z \in \mathbb{C} : & |\gamma| \le 1 \text{ for all non-defective eigenvalues } \gamma \text{ of } M(z) \\ & \text{and } |\gamma| < 1 \text{ for all defective eigenvalues } \gamma \text{ of } M(z)\}. \end{aligned}$$

It is generally believed that explicit GLMs can have only finite stability regions: to achieve infinite stability regions an implicit method is required. (We prove this for a subclass of GLMs in Theorem 8.) This means explicit methods are suitable only for non-stiff problems, and stiff problems require the use of an implicit method to avoid the penalty of stability imposed restrictions on the stepsize.

A desirable property for a method intended for the solution of stiff problems is that its stability region R include the left complex half-plane. A method with this property is called A-stable, and has the favourable characteristic that the stepsize is never restricted by stability on linear constant-coefficient problems, regardless of stiffness. If additionally $\|M(z)\| \to 0$ as $z \to \infty$, the method is called L-stable.

The previous discussion has considered stability for the scalar test problem

$$u' = \mu u, \quad \mu \in \mathbb{C}.$$

However, stability for the vector problem

$$u' = Cu, \quad C \in \mathbb{R}^{K \times K},$$

where C is diagonalisable, can be shown to follow from stability on each of the scalar problems

$$u' = \mu_i u, \quad i = 1, \ldots, K,$$

where $\mu_1, \ldots, \mu_K$ are the eigenvalues of C. This, as well as a weaker result which holds when C is not diagonalisable, is given by Theorem 9 in Appendix D.

1.6 Overview of the Thesis

We now give an overview of the remainder of the thesis, and identify the main results. The next chapter, Chapter 2, describes the development of our diagonal methods. In it, Section 2.1 defines the class of methods we refer to as DIMSEMs (Diagonally IMplicit Single-Eigenvalue Methods) and gives some motivation for this subclass of GLMs. Sections 2.2 and 2.3 give results on the order properties of single-eigenvalue methods, including the DIMSEMs, and show that the single-eigenvalue property enables some Runge-Kutta order bounds to be extended to this class of methods. Section 2.4 develops an approach for the derivation of DIMSEMs, and Section 2.5 describes an offshoot class of methods, LDMs (for

L-stable diagonal methods), which contains L-stable methods of all orders.

The main contribution of this thesis is in two theorems, Theorems 4 and 5, (stated in Section 2.6 and proven in Appendices A and B) which describe how to construct DIMSEMs of any order greater than or equal to two. The remainder of the thesis is concerned with selecting from this class specific methods of orders 2 to 6 and testing them.

More specifically, Section 2.7 deals with finding L-stable DIMSEMs and optimising the error coefficients through the selection of free parameters, while Section 2.8 considers optimisation of the LDMs. Preliminary fixed-stepsize sequential testing of the DIMSEMs and LDMs is discussed in Section 2.9. Variable-stepsize implementations and testing of the DIMSEMs are covered in Chapter 3, beginning with the investigation of variable-stepsize stability and the selection and testing of error estimators and stepsize control schemes, and concluding with performance testing of the DIMSEMs on sequential and parallel machines. The testing demonstrates the viability of the methods by showing them to be competitive with a fixed-order version of the popular integrator LSODE on a practical test problem.

The thesis continues with conclusions and future work in Chapter 4, followed by appendices containing proofs of the major theorems and related lemmas.

Chapter 2

Development of Diagonal Methods

2.1 How We Came to DIMSEMs

The goal of our research is to produce A-stable GLMs with diagonal B_1. Such methods are suitable for solving stiff problems on a parallel machine. A second goal is to find a whole class of such methods, together with a methodology for generating methods of any desired order. The first subclass of GLMs which we considered was Diagonally Implicit Multistep Runge Kutta methods (DIMRKs), which are MRKs with diagonal B_1. Two classes of these methods, arising from modifications to the methods of Burrage and Chipman [6], were investigated initially. The first satisfies $B(r+s-1)$ and $C(r-1)$, and hence has order at least r, with $c_1, \ldots, c_s$ and $\lambda_1, \ldots, \lambda_s$ as free parameters. The second satisfies $B(r)$ and $C(r)$, also has order at least r, and has $\gamma_1, \ldots, \gamma_{s-1}$ (defined in (1.4)) as free parameters. The simplified order conditions were solved algebraically using Maple [30], and the free parameter space was numerically searched for methods that appeared to be A-stable. The A-stability of these candidate methods was then verified theoretically using a technique based on the Schur criterion and the maximum principle of complex analysis [6]. Although methods of orders 2 to 5 were found this way, this simple approach does not appear to be extensible to higher orders due to the rapidly increasing cost, as order increases, of solving the order conditions by brute force. Another disadvantage of this approach is that it requires trial-and-error numerical searching to find A-stable methods. Little insight is available into the nature of A-stable choices of the parameters, as the generation of algebraic expressions for objects such as the characteristic polynomial of the stability matrix, or the eigenvalues, is computationally intractable.

Our eventual focus on a new class of methods for which it is easier to test for A-stability

grew out of a seminar [8] given by Butcher, in which he introduced his work on DIMSIM methods. Although Butcher's subsequent development of these methods takes a different direction from ours, the two classes are sufficiently related to warrant discussing DIMSIMs in the next section.

2.1.1 DIMSIMs — Diagonally Implicit Multi-Stage Integration Methods

Although our introduction to the GLMs known as DIMSIMs was through an early seminar by Butcher, they have now been described in published work [11, 12, 13, 14]. The methods are classified into four types, depending on the structure of B_1. These are defined in Fig. 2.1, which also indicates the type of problem and kind of architecture for which each type is suitable. Types 1 and 3 are explicit methods, intended for non-stiff problems, while type 2 is

DIMSIM	B_1 Structure	Problem	Machine
Type 1	$\begin{bmatrix} 0 & & & & \\ b_{21} & 0 & & & \\ b_{31} & b_{32} & \ddots & & \\ \vdots & \vdots & \ddots & 0 & \\ b_{s1} & b_{s2} & \cdots & b_{ss} & 0 \end{bmatrix}$	non stiff	serial
Type 2	$\begin{bmatrix} \lambda & & & & \\ b_{21} & \lambda & & & \\ b_{31} & b_{32} & \ddots & & \\ \vdots & \vdots & \ddots & \lambda & \\ b_{s1} & b_{s2} & \cdots & b_{ss} & \lambda \end{bmatrix}$	stiff	serial
Type 3	$\begin{bmatrix} 0 & & & & \\ 0 & 0 & & & \\ 0 & 0 & \ddots & & \\ \vdots & \vdots & \ddots & 0 & \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}$	non stiff	parallel
Type 4	$\begin{bmatrix} \lambda & & & & \\ & \lambda & & & \\ & & \ddots & & \\ & & & \lambda & \\ & & & & \lambda \end{bmatrix}$	stiff	parallel

Figure 2.1: Butcher's four DIMSIM types [11]

singly diagonally implicit, and type 4 is singly strictly diagonally implicit.

The DIMSIMs described in [11, 12, 13] are constructed with $r = s$, $A_1 = I$, $A_2 e = e$, where e is an r-vector of all 1's, and B_2 is determined from the simplified order conditions $A(r)$, $B(r)$, and $C(r)$ in terms of the c_i's, B_1, and A_2. This gives an order and a stage order of r. Although Butcher gives his order conditions in terms of the exponential function and integrals of interpolating polynomials instead of the expressions (1.12), (1.13), and (1.14), the two representations can be shown to be equivalent. Additional DIMSIMs in which various combinations of r, s, the order, and the stage order differ by one from the values above, are developed by Butcher and Jackiewicz in [14].

In his seminar, Butcher exhibited a type-2, order-2 DIMSIM which had been constructed to have the same stability region as a certain diagonally implicit Runge-Kutta (DIRK) method. This was achieved by making one eigenvalue of the stability matrix $M(z)$ zero for all z, and making the other eigenvalue equal to the stability function of the DIRK method. A generalisation of this example leads to the idea of *Single-Eigenvalue Methods* (SEMs), in which all but one eigenvalue of $M(z)$ is zero for all z. We use the term DIMSEM — *Diagonally IMplicit Single-Eigenvalue Method* — to describe a SEM having a diagonal B_1 matrix. Butcher [11] mentions it as a useful stability property, also possessed by Runge-Kutta methods. One benefit of a SEM is the absence of spurious solutions corresponding to non-zero, non-principal eigenvalues, although other factors tend to be more significant in determining the suitability of a particular method in practice. Nevertheless, we have found the single-eigenvalue property helpful for the simplification it provides when searching for A-stable methods. Although their DIMSIMs are not SEMs, Butcher and Jackiewicz [12, 13, 14] have since been able to find A-stable methods by deriving tractable algebraic expressions for the eigenvalues of their stability matrices.

2.1.2 Application of the Single-Eigenvalue Property to Proving A- and L-Stability

The problem of determining A- and L-stability of a GLM simplifies considerably if the stability matrix has only one nonzero eigenvalue. If we let

$$\text{eig}(M(z)) = \{e_1(z), \ldots, e_r(z)\},$$

and suppose that

$$e_2(z) = \cdots = e_r(z) = 0 \quad \forall z \in \mathbb{C},$$

then

$$\text{tr}(M(z)) = \sum_{i=1}^{r} e_i(z) = e_1(z).$$

Now recall that

$$M(z) = A_2 + zB_2(I - zB_1)^{-1}A_1,$$

so that the trace is a rational function of z, say

$$e_1(z) = \frac{P(z)}{Q(z)}. \tag{2.1}$$

For a strictly diagonal method, where

$$B_1 = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_s \end{bmatrix},$$

it follows easily that

$$Q(z) = \prod_{i=1}^{s}(1 - \lambda_i z), \tag{2.2}$$

so that $e_1(z)$ is analytic and has no poles in $\mathbb{C}^-$ provided all the λ_i's have positive real parts. (For general B_1, (2.2) holds if $\lambda_1, \ldots, \lambda_s$ are the eigenvalues of B_1. This follows from the later result (2.8).) By the maximum principle, it suffices to check that $|e_1(iy)| \leq 1 \ \forall y \in \mathbb{R}$ to ensure A-stability. Now, if we define the *E-polynomial*

$$E(y) = |P(iy)|^2 - |Q(iy)|^2, \tag{2.3}$$

then

$$|e_1(iy)| \leq 1 \Leftrightarrow E(y) \leq 0,$$

and, assuming the polynomials P and Q have real coefficients, a sufficient condition for A-stability is that $\lambda_i \in \mathbb{C}^+$ for $i = 1, \ldots, s$ and $E(y)$ has all non-positive coefficients. (This follows from the observation that, if P and Q are real polynomials, then $E(y)$ is an even polynomial, that is, the coefficient of each odd power of y in $E(y)$ is zero.) An A-stable SEM is also L-stable iff $|e_1(z)| \to 0$ as $|z| \to \infty$. From (2.1), this is equivalent to $\deg(P) < \deg(Q)$. Given the coefficient matrices of a candidate method, not only can these conditions be easily checked, but they are often simple enough to provide insight into what choices of the free parameters are required to ensure A- and L-stability.

2.2 Maximal Order of Single-Eigenvalue GLMs

Before beginning the construction of DIMSEMs, we develop an upper bound on the order they might attain. The following three theorems and two corollaries apply to a larger class of methods than DIMSEMs, in that B_1 need not be diagonal. They extend to single-eigenvalue GLMs the corresponding result of Jackson and Nørsett for Runge-Kutta methods [26, Theorem 2.1].

Theorem 1 *Let $e_1(z), \ldots, e_r(z)$ be the eigenvalues of the stability matrix $M(z)$ of a GLM. Assume $e_1(z) = e^z + O(z^{p+1})$, and $e_2(z) = \cdots = e_r(z) = 0$ for all z. Then*

$$p \leq d + \delta + \sum_{i=1}^{k} r_i,$$

where B_1 has eigenvalues $\gamma_1, \ldots, \gamma_k$ with no repetitions, r_i is the size of the largest Jordan block associated with γ_i, d is the number of distinct complex non-real eigenvalues of B_1, and δ is 1 if B_1 has at least one nonzero real eigenvalue and 0 otherwise.

Remark. We have not assumed B_1 is diagonal in the theorem above. Also, d is even if B_1 is real.

Proof. Let S be a matrix that transforms B_1 to its Jordan canonical form,

$$\hat{B}_1 = S^{-1} B_1 S = \mathsf{diag}(J_i),$$

where the Jordan blocks can be written as

$$J_i = \gamma_i I + E_i, \quad E_i^{r_i} = 0, \quad i = 1, \ldots, k.$$

Then the stability matrix is

$$\begin{aligned} M(z) &= A_2 + z B_2 (I - z B_1)^{-1} A_1 \\ &= A_2 + z \hat{B}_2 (I - z \hat{B}_1)^{-1} \hat{A}_1 \\ &= A_2 + z \hat{B}_2 X(z) \hat{A}_1, \end{aligned}$$

where $\hat{B}_2 = B_2 S$, $\hat{A}_1 = S^{-1} A_1$ and $X(z) = (I - z\hat{B}_1)^{-1}$. Also,

$$I - z\hat{B}_1 = I - z\,\mathsf{diag}(\gamma_i I + E_i)$$

$$= \quad \text{diag}\left((1-\gamma_i z)I - zE_i\right),$$

so that

$$X(z) = \begin{bmatrix} X_1(z) & & \\ & \ddots & \\ & & X_k(z) \end{bmatrix}, \tag{2.4}$$

where

$$\begin{aligned} X_i(z) &= ((1-\gamma_i z)I - zE_i)^{-1} \\ &= \left((1-\gamma_i z)\left(I - \frac{z}{1-\gamma_i z}E_i\right)\right)^{-1} \\ &= \frac{1}{1-\gamma_i z}\left(I - \frac{z}{1-\gamma_i z}E_i\right)^{-1} \\ &= \frac{1}{1-\gamma_i z}\sum_{q=0}^{r_i-1}\left(\frac{z}{1-\gamma_i z}E_i\right)^q, \quad i = 1,\ldots,k, \end{aligned} \tag{2.5}$$

since $E_i^{r_i} = 0$. Then the nonzero eigenvalue of $M(z)$ is

$$\begin{aligned} e_1(z) &= \text{tr}(M(z)) \\ &= \text{tr}(A_2) + \text{tr}(z\hat{B}_2 X(z)\hat{A}_1), \end{aligned} \tag{2.6}$$

and

$$\begin{aligned} \left(z\hat{B}_2 X(z)\hat{A}_1\right)_{ll} &= z\sum_m (\hat{B}_2)_{lm}(X(z)\hat{A}_1)_{ml} \\ &= z\sum_m (\hat{B}_2)_{lm}\sum_n X(z)_{mn}(\hat{A}_1)_{nl} \\ &= z\sum_m\sum_n \alpha_{mnl}X(z)_{mn} \end{aligned} \tag{2.7}$$

for suitable α_{mnl}'s. From (2.4) however it follows that

$$(2.7) = z\sum_{i=1}^{k}\sum_m\sum_n \beta_{imnl}X_i(z)_{mn}$$

for suitable β_{imnl}'s and so from (2.5) it follows that

$$zX_i(z)_{mn} = \sum_{q=0}^{r_i-1}\frac{z^{q+1}(1-\gamma_i z)^{r_i-(q+1)}}{(1-\gamma_i z)^{r_i}}(E_i^q)_{mn},$$

and, using this in (2.7), we get

$$(z\hat{B}_2X(z)\hat{A}_1)_{ll} = \sum_{i=1}^{k}\sum_{m}\sum_{n}\beta_{imnl}\sum_{q=0}^{r_i-1}(E_i^q)_{mn}\frac{z^{q+1}(1-\gamma_i z)^{r_i-(q+1)}}{(1-\gamma_i z)^{r_i}}.$$

Therefore, for suitable α_{iq}'s,

$$\begin{aligned}\mathsf{tr}(z\hat{B}_2X(z)\hat{A}_1) &= \sum_{i=1}^{k}\sum_{q=0}^{r_i-1}\alpha_{iq}\frac{z^{q+1}(1-\gamma_i z)^{r_i-(q+1)}}{(1-\gamma_i z)^{r_i}}\\ &= \sum_{i=1}^{k}\frac{P_i(z)}{Q_i(z)},\end{aligned}$$

where

$$P_i(z) = \sum_{q=0}^{r_i-1}\alpha_{iq}z^{q+1}(1-\gamma_i z)^{r_i-(q+1)}$$

has degree at most r_i, and

$$Q_i(z) = (1-\gamma_i z)^{r_i}.$$

Thus from (2.6)

$$e_1(z) = \mathsf{tr}(A_2) + \sum_{i=1}^{k}\frac{P_i(z)}{Q_i(z)}. \tag{2.8}$$

Letting $a = \mathsf{tr}(A_2)$ gives

$$\begin{aligned}e_1(z) &= a + \sum_{i=1}^{k}\frac{P_i(z)}{Q_i(z)}\\ &= \frac{a\prod_i Q_i(z) + \sum_i\left(P_i(z)\prod_{j\neq i}Q_j(z)\right)}{\prod_i Q_i(z)}\\ &= \frac{P(z)}{Q(z)},\end{aligned}$$

where

$$\deg(P(z)) \leq \sum_{i=1}^{k} r_i.$$

It then follows from Theorem 8 in [23] that, if

$$e_1(z) = e^z + O(z^{p+1}), \tag{2.9}$$

then

$$p \leq d + \delta + \sum_{i=1}^{k} r_i$$

as required. □

In several of the following theorems, we need to guarantee that the exact values $\hat{y}_n$ approximated by the external stages for the linear test problem $u' = \mu u$ are nonzero. One way to ensure this is to require that the true-value function satisfies $q(\phi) \neq 0$. This property is implied by the preconsistency condition of [5, 7] and is satisfied by any useful method, including both multistep- and Nordsieck-type GLMs.

Lemma 1 *Let $e_1(z), \ldots, e_r(z)$ be the eigenvalues of the stability matrix $M(z)$ of a GLM for which $q(\phi) \neq 0$. Assume the method has local discretisation error of order $q + 1$ and $e_2(z) = \cdots = e_r(z) = 0$ for all z. Then $e_1(z) = e^z + O(z^{q+1})$.*

Proof. We apply the method to the linear test problem $u' = \mu u$, $u(0) = 1$, for which $u(x) = e^{\mu x}$ and

$$y_{n+1} = M(z) y_n, \tag{2.10}$$

where $z = h\mu$. The local discretisation error is

$$l_n = y_{n+1} - \hat{y}_{n+1}, \tag{2.11}$$

where $y_n = \hat{y}_n$ and, for all k, $\hat{y}_k = B(q, u(x_k))$. For $u' = \mu u$, the elementary differentials satisfy $D(t)(u) = \mu^i u$ when $t = \tau_i$, the tall tree of order i, and $D(t)(u) = 0$ otherwise. Therefore

$$\hat{y}_k = \sum_{i=0}^{\infty} q(\tau_i) \frac{z^i}{i!} u(x_k). \tag{2.12}$$

But $u(x_{n+1}) = e^z u(x_n)$, so that comparing (2.12) for $k = n$ and $k = n + 1$ gives

$$\hat{y}_{n+1} = e^z \hat{y}_n. \tag{2.13}$$

Combining this with (2.10) and (2.11) gives

$$O(z^{q+1}) = l_n = M(z)\hat{y}_n - e^z \hat{y}_n,$$

which implies

$$M(z)\hat{y}_n = e^z \hat{y}_n + O(z^{q+1}). \tag{2.14}$$

From (2.12),

$$\hat{y}_n = q(\phi)u(x_n) + O(z),$$

which is nonzero for z sufficiently small. Therefore, from (2.14), there is a matrix $E_n = O(z^{q+1})$ such that

$$(M(z) + E_n)\,\hat{y}_n = e^z \hat{y}_n. \tag{2.15}$$

That is, $M(z) + E_n$ has an eigenvalue e^z. This eigenvalue is simple for z sufficiently small, otherwise, as $z \to 0$, $M(0)$ would have a multiple eigenvalue at 1, contradicting the single-eigenvalue property. From a theorem of Wilkinson [31, Chapter 2, Section 5] on the perturbation of eigenvalues, it follows that $M(z)$ has an eigenvalue $e(z) = e^z + O(z^{q+1})$. Since $e(0) = 1$, $e(z)$ is not any of the zero eigenvalues $e_2(z), \ldots, e_r(z)$. Therefore $e(z) = e_1(z)$ and $e_1(z) = e^z + O(z^{q+1})$, as required. □

The previous lemma can now be used to bound the local order of the method.

Theorem 2 *Let $e_1(z), \ldots, e_r(z)$ be the eigenvalues of the stability matrix $M(z)$ of a GLM for which $q(\phi) \neq 0$. Assume $e_2(z) = \cdots = e_r(z) = 0$ for all z and $p \geq 0$ is the largest value such that $e_1(z) = e^z + O(z^{p+1})$. Then the order of the local discretisation error of the method is at most $p + 1$.*

Proof. Suppose $l_n = O(z^{q+1})$ for some $q \geq 0$. We need to show $q \leq p$. By Lemma 1,

$$e_1(z) = e^z + O(z^{q+1}). \tag{2.16}$$

But p is the largest such q in (2.16), so $q \leq p$, and the order of the local discretisation error of the GLM is at most $p + 1$. □

The next result follows immediately from Theorems 1 and 2.

Corollary 1 *Let $e_1(z), \ldots, e_r(z)$ be the eigenvalues of the stability matrix $M(z)$ of a GLM for which $q(\phi) \neq 0$ and assume $e_2(z) = \cdots = e_r(z) = 0$ for all z. Then the order of the*

local discretisation error of the GLM is at most $p+1$*, where*

$$p = d + \delta + \sum_{i=1}^{k} r_i,$$

B_1 *has eigenvalues* $\gamma_1, \ldots, \gamma_k$ *with no repetitions,* r_i *is the size of the largest Jordan block associated with* γ_i*,* d *is the number of distinct complex non-real eigenvalues of* B_1*, and* δ *is* 1 *if* B_1 *has at least one nonzero real eigenvalue and* 0 *otherwise.*

We now state a theorem that bounds the order of the global error of a SEM.

Theorem 3 *Let* $e_1(z), \ldots, e_r(z)$ *be the eigenvalues of the stability matrix* $M(z)$ *of a GLM for which* $q(\phi) \neq 0$*. Assume* $e_2(z) = \cdots = e_r(z) = 0$ *for all* z *and* $p \geq 0$ *is the largest value such that* $e_1(z) = e^z + O(z^{p+1})$*. Then the order of the global error of the method is at most* p*.*

Proof. By Theorem 2, the order of the local discretisation error of the method is at most $p+1$. By [29, Theorem 5.3] for example, it follows that the order of the global error is at most $p+1$ also. We prove below by contradiction that the order of the global error is not $p+1$, so it is at most p.

To this end, apply the method to the linear test problem $u' = \mu u$, $u(0) = 1$, for which $u(x) = e^{\mu x}$ and $y_{n+1} = M(z)y_n$ for $z = h\mu$. Let $e_1(z)$ be the nonzero eigenvalue of $M(z)$. For now fix $z = h\mu$ and let $M = M(z)$ and $e_1 = e_1(z)$. Let T be a matrix which transforms M to its Jordan canonical form, $J = T^{-1}MT$, where

$$J = \begin{bmatrix} e_1 & & & & \\ & 0 & \delta_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \delta_{r-1} \\ & & & & 0 \end{bmatrix}, \qquad \delta_i = 0 \text{ or } 1.$$

Let T_i, $i = 1, \ldots, r$, be the columns of T, known as generalised eigenvectors. Then $MT = TJ$ implies

$$\begin{aligned} MT_1 &= e_1 T_1 \\ MT_2 &= 0 \\ MT_3 &= \delta_2 T_2 \Longrightarrow M^2 T_3 = \delta_2 M T_2 = 0 \end{aligned} \tag{2.17}$$

$$\vdots$$

$$MT_r = \delta_{r-1}T_{r-1} \Longrightarrow M^{r-1}T_r = \delta_{r-1}\cdots\delta_2 MT_2 = 0,$$

so that

$$M^{k-1}T_k = 0, \quad \text{for } k = 2, \ldots, r. \tag{2.18}$$

Since T is nonsingular, $T_1, \ldots, T_r$ are linearly independent, so there exist scalars $\alpha_1, \ldots, \alpha_r$ such that

$$y_0 = \sum_{i=1}^{r} \alpha_i T_i.$$

Since we may multiply T by a constant, we may assume without loss of generality that $\alpha_1 = 1$. Then

$$\begin{aligned} y_n &= M^n y_0 \\ &= \sum_{i=1}^{r} \alpha_i M^n T_i \\ &= e_1^n T_1, \quad \text{for } n \geq r-1, \end{aligned} \tag{2.19}$$

by (2.17) and (2.18). From (2.13) we have $\hat{y}_{n+1} = e^z \hat{y}_n$, so that

$$\hat{y}_n = e^{nz}\hat{y}_0. \tag{2.20}$$

Then, for $y_0 = \hat{y}_0$, the global error is

$$\begin{aligned} g_{n-1} &= y_n - \hat{y}_n, \\ &= e_1^n T_1 - e^{nz}\hat{y}_0, \quad \text{for } n \geq r-1, \end{aligned}$$

from (2.19) and (2.20). Now allowing z to vary again and making explicit the dependencies on z gives

$$g_{n-1}(z) = e_1(z)^n T_1(z) - e^{nz}\hat{y}_0(z), \quad \text{for } n \geq r-1. \tag{2.21}$$

Recall that p is the largest integer such that

$$e_1(z) = e^z + O(z^{p+1}). \tag{2.22}$$

It therefore follows that p is the largest integer such that the following two relations hold:

$$e_1(z)^r = e^{rz} + O(z^{p+1}), \tag{2.23}$$

$$e_1(z)^n = e^{nz} + O(z^p), \tag{2.24}$$

where r is the number of external stages of the method and $nz = C$ for some constant $C > 0$ in (2.24).

Now we assume that the order of the global error is $p + 1$ and show that this leads to a contradiction, from which it follows that the order of the global error is at most p.

To begin, let $n = r$. If the order of the global error were $p + 1$ for $p \geq 0$, then it follows from (2.21) that

$$g_{r-1}(z) = e_1(z)^r T_1(z) - e^{rz}\hat{y}_0(z) = O(z^{p+1}). \tag{2.25}$$

As $z \to 0$, $g_{r-1}(z) \to 0$, $e^{rz} \to 1$, $e_1(z)^r \to 1$ by (2.23) and $\hat{y}_0(z) = B(q, u(0)) = q(\phi)u(0) + O(z) \to q(\phi)$, since $u(0) = 1$. Therefore, $T_1(z) \to q(\phi)$. In particular, $T_1(z)$ is bounded away from both **0** and ∞ as $z \to 0$ since, by the hypothesis of the theorem, $q(\phi) \neq 0$.

We can refine the last result by rewriting (2.25) as

$$g_{r-1}(z) = \left[e_1(z)^r - e^{rz}\right] T_1(z) + e^{rz}\left[T_1(z) - \hat{y}_0(z)\right] = O(z^{p+1}) \tag{2.26}$$

and by noting that $e_1(z)^r - e^{rz} = O(z^{p+1})$ by (2.23), $T_1(z)$ is bounded and $e^{rz} \to 1$ as $z \to 0$, so that

$$T_1(z) - \hat{y}_0(z) = O(z^{p+1}). \tag{2.27}$$

Note that (2.27) does not depend on n, and so holds even when the assumption that $n = r$ is removed.

Now let $nz = C$ for some constant $C > 0$ as $z \to 0$ and rewrite (2.21) as

$$g_{n-1}(z) = \left[e_1(z)^n - e^{nz}\right] T_1(z) + e^{nz}\left[T_1(z) - \hat{y}_0(z)\right] = O(z^{p+1}), \tag{2.28}$$

where we have again assumed that the order of the global error is $p + 1$. Since $e^{nz} = e^C$ and, as $z \to 0$, $T_1(z)$ is bounded away from **0** and $T_1(z) - \hat{y}_0(z) = O(z^{p+1})$ by (2.27), (2.28) implies that $e_1(z)^n - e^{nz} = O(z^{p+1})$ also, contradicting (2.24) and thereby completing the proof. □

The next result follows immediately from Theorems 1 and 3.

Corollary 2 *Let $e_1(z), \ldots, e_r(z)$ be the eigenvalues of the stability matrix $M(z)$ of a GLM for which $q(\phi) \neq 0$ and assume $e_2(z) = \cdots = e_r(z) = 0$ for all z. Then the order of the global error of the GLM is at most p, where*

$$p = d + \delta + \sum_{i=1}^{k} r_i,$$

B_1 has eigenvalues $\gamma_1, \ldots, \gamma_k$ with no repetitions, r_i is the size of the largest Jordan block associated with γ_i, d is the number of distinct complex non-real eigenvalues of B_1, and δ is 1 if B_1 has at least one nonzero real eigenvalue and 0 otherwise.

2.3 Singly Implicit DIMSEMs and DIMSEM-MRKs — Some Negative Results

Applying Corollaries 1 and 2 to DIMSEMs, for which $B_1 = \text{diag}(\lambda_i)$ with $\lambda_i \in \mathbb{R}$ for $i = 1, \ldots, s$, we see that the orders of the local and global discretisation errors are at most $s + 2$ and $s + 1$, respectively, since $d = 0$, $\delta = 1$, each $r_i = 1$ and $k \leq s$ imply $p \leq s + 1$. Moreover, the maximal values $s + 2$ and $s + 1$ occur only if all the λ_i's are distinct. It remains to be seen whether this order is attainable for general s.

These corollaries also show that high order, singly implicit DIMSEMs do not exist. In particular, the orders of the local and global discretisation errors of a DIMSEM with $B_1 = \lambda I$, where $\lambda \in \mathbb{R}$, are at most 3 and 2, respectively, since $d = 0$, $\delta = 1$, $k = 1$ and $r_1 = 1$ imply $p = 2$.

A consequence of these results is that we need to allow a general diagonal matrix B_1 in DIMSEMs. Our search for such methods began with the class of MRKs with $r = s$, and satisfying the simplifying assumptions $B(r)$ and $C(r - 1)$. These methods have order at least r. It turns out, however, that such a DIMSEM with $r = 3$ cannot be A-stable. In fact, as we show in the remainder of this section, it has a finite stability region.

We begin by solving the order conditions. $C(r-1)$ imposes r^2 independent conditions on A_1, which determine it in terms of B_1, the c_i's and $q : T \to \mathbb{R}^r$, which are free parameters. (This follows rigorously from (2.34), and the non-singularity of the matrix Q for MRKs in Section 2.7.) $B(r)$ imposes $r(r + 1)$ conditions on A_2 and B_2, which determine the γ_i's and α_r in (1.3) and (1.4) in terms of the free parameters. (r^2 of the conditions are trivially

satisfied by the special structure of A_2 and B_2.) The remaining α_i's in A_2 are determined by the requirement that the stability matrix has $r-1$ zero eigenvalues, as follows.

Because of the special form of A_2 and B_2, the stability matrix,

$$M(z) = A_2 + zB_2(I - zB_1)^{-1}A_1,$$

of an MRK has a particularly simple characteristic polynomial:

$$\begin{aligned} \varphi(w,z) &= \det(wI - M(z)) \\ &= w^r - \beta_{r-1}(z)w^{r-1} - \cdots - \beta_1(z)w - \beta_0(z), \end{aligned} \tag{2.29}$$

where

$$\beta_{i-1}(z) = \alpha_i + \sum_{j=1}^{s} \frac{\gamma_j a_{ji} z}{1 - \lambda_j z}, \quad i = 1, \ldots, r, \tag{2.30}$$

and $a_{ji} = (A_1)_{ji}$. The requirement that $M(z)$ has $r-1$ zero eigenvalues implies that φ has w^{r-1} as a factor. This requires $\beta_i(z) = 0$ for all z for $i = 0, \ldots, r-2$, from which it follows by letting $z = 0$ in (2.30) that $\alpha_i = 0$ for $i = 1, \ldots, r-1$. Thus all the coefficient matrices of the method are determined in terms of c_i's and λ_i's, which are still free.

We now examine further the implications of the zero-eigenvalue condition. Substituting

$$\frac{1}{1 - \lambda_j z} = \sum_{k=0}^{\infty} \lambda_j^k z^k, \quad |z| < \frac{1}{|\lambda_j|},$$

into (2.30) and re-arranging terms, we get

$$\beta_{i-1}(z) = \sum_{k=0}^{\infty} z^{k+1} \sum_{j=1}^{s} \gamma_j a_{ji} \lambda_j^k = 0 \quad \forall |z| < \frac{1}{\max_j |\lambda_j|}, \quad i = 1, \ldots, r-1,$$

from which it follows that

$$\sum_{j=1}^{s} \gamma_j a_{ji} \lambda_j^k = 0, \quad i = 1, \ldots, r-1, \quad k = 0, 1, 2, \ldots \tag{2.31}$$

Now let n be the number of distinct λ_i's, that is,

$$S = \{\lambda_1, \ldots, \lambda_s\} = \{\tilde{\lambda}_1, \ldots, \tilde{\lambda}_n\},$$

where the $\tilde{\lambda}_i$'s are distinct. Let

$$S_i = \left\{ j : \lambda_j = \tilde{\lambda}_i \right\}.$$

Then

$$\bigcup_{i=1}^{n} S_i = \{1, \ldots, s\},$$

and it follows from (2.31) that

$$\begin{aligned} 0 &= \sum_{l=1}^{n} \sum_{j \in S_l} \gamma_j a_{ji} \lambda_j^k \\ &= \sum_{l=1}^{n} \tilde{\lambda}_l^k w_{il}, \quad i = 1, \ldots, r-1, \quad k = 0, 1, 2, \ldots, \end{aligned}$$

where

$$w_{il} = \sum_{j \in S_l} \gamma_j a_{ji}.$$

This implies that, for $i = 1, \ldots, r-1$,

$$\begin{bmatrix} 1 & \cdots & 1 \\ \tilde{\lambda}_1 & \cdots & \tilde{\lambda}_n \\ & \vdots & \\ \tilde{\lambda}_1^{n-1} & \cdots & \tilde{\lambda}_n^{n-1} \end{bmatrix} \begin{bmatrix} w_{i1} \\ \vdots \\ \vdots \\ w_{in} \end{bmatrix} = 0.$$

Since the $\tilde{\lambda}_i$'s are distinct, the Vandermonde matrix is non-singular, and hence the w_{ij}'s must be zero. Thus we have that

$$\sum_{j \in S_l} \gamma_j a_{ji} = 0, \quad i = 1, \ldots, r-1, \quad l = 1, \ldots, n. \tag{2.32}$$

We now consider the consequences of this for various numbers of distinct λ_i's.

In particular, if all the λ_i's are distinct, then

$$\gamma_j a_{ji} = 0, \quad i = 1, \ldots, r-1, \quad j = 1, \ldots, s,$$

so that for each $j = 1, \ldots, s$, either $\gamma_j = 0$ or $a_{ji} = 0$, $i = 1, \ldots, r-1$. In the case $r = s = 3$, there turns out to be no method that satisfies these conditions. (Recall that the γ_j's and a_{ji}'s are already determined in terms of the λ_i's and c_i's.)

As noted at the beginning of this section, a DIMSEM with all λ_i's equal can be of order

at most 2. Since this MRK is of order at least 3, the λ_i's cannot all be equal.

The only remaining choice is two λ_i's equal and one different. Suppose $\lambda_1 = \lambda_2 \neq \lambda_3$. (Because of symmetry, the other permutations turn out to behave identically.) In this case $S_1 = \{1, 2\}$, $S_2 = \{3\}$, and (2.32) has the unique solution

$$\begin{aligned} \lambda_1 &= \lambda_2 = 1/3, & \lambda_3 &= 0, \\ c_1 &= \frac{1}{3}\,\frac{21c_2 - 50}{3c_2 - 7}, & c_3 &= 2, \end{aligned}$$

with c_2 free. The remaining non-zero eigenvalue however is

$$\beta_{r-1}(z) = \frac{1}{2}\,\frac{6 + 4z + z^2}{3 - z},$$

for which $|\beta_{r-1}(z)| \to \infty$ as $|z| \to \infty$. Therefore the method has a finite stability region.

In conclusion, singly implicit DIMSEMs, as well as the subclass of DIMSEM-MRKs considered above, cannot form the basis for a class of A-stable DIMSEMs of high order. It is necessary to relax the conditions imposed in this section to find a suitable class of DIMSEMs for solving stiff problems. We chose to look next at general DIMSEM-GLMs.

2.4 Linking Stability and Order Conditions

To produce a method with $r - 1$ zero eigenvalues, it is necessary to zero the coefficient functions $\beta_i(z)$ of w^i in the characteristic polynomial $\varphi(w, z)$ in (2.29), for $i = 0, \ldots, r - 2$. As each $\beta_i(z)$ is a rational function of z, this amounts to zeroing the numerator polynomial, which involves setting all the coefficients of each polynomial to zero. These coefficients are complicated nonlinear functions of the coefficients of the method. We found solving these equations, together with the simplified order conditions, to be an intractable problem when done algebraically to facilitate later choice of the remaining free parameters for A-stability. The computation is eased somewhat by transforming the stability matrix in such a way that the simplified order conditions can be incorporated into it. Then the $r-1$ eigenvalues can be set to zero, with the assurance that a method of the desired order can be recovered later. To do this, we choose the simplifying assumptions $A(r-1)$, $B(r-1)$ and $C(r-1)$, because, from (1.13) and (1.14), $B(r-1)$ and $C(r-1)$ can be conveniently written in matrix form as follows.

$$B(r-1) \quad : \quad A_2 Q + B_2 W = QP, \tag{2.33}$$

$$C(r-1) \quad : \quad A_1 Q + B_1 W = V, \tag{2.34}$$

where

$$\begin{aligned}
Q &= [q_0, \ldots, q_{r-1}] \in \mathbb{R}^{r\times r}, \\
q_k &= q(\tau^k) = q(t) \quad \forall t \ni \rho(t) = k, \quad k = 0, \ldots, r-1, \\
V_{ij} &= c_i^{j-1}, \quad V \in \mathbb{R}^{s\times r}, \\
W_{ij} &= (j-1)c_i^{j-2}, \quad W \in \mathbb{R}^{s\times r}, \\
P_{ij} &= \left\{ \begin{array}{ll} \left(\begin{array}{c} j-1 \\ i-1 \end{array} \right), & j \geq i \\ 0, & \text{otherwise} \end{array} \right\}, \quad P \in \mathbb{R}^{r\times r},
\end{aligned}$$

and 0^0 is taken to be 1. In the following, we assume that Q is nonsingular. Let $\bar{B}_2 = Q^{-1}B_2$, and consider the transformed stability matrix

$$\begin{aligned}
\bar{M}(z) &= Q^{-1}M(z)Q \\
&= Q^{-1}\left(A_2 + zB_2(I - zB_1)^{-1}A_1\right)Q \\
&= Q^{-1}(QP - B_2W) + zQ^{-1}B_2(I - zB_1)^{-1}(V - B_1W) \\
&= P - \bar{B}_2W + z\bar{B}_2(I - zB_1)^{-1}(V - B_1W) \\
&= P + \bar{B}_2\left[z(I - zB_1)^{-1}(V - B_1W) - W\right] \\
&= P + \bar{B}_2\left[(I - zB_1)^{-1}\left(zV - zB_1W - (I - zB_1)W\right)\right] \\
&= P + \bar{B}_2(I - zB_1)^{-1}(zV - W).
\end{aligned} \tag{2.35}$$

If we define

$$U = \begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & r-1 \\ & & & 0 \end{bmatrix},$$

then $VU = W$ and

$$\bar{M}(z) = P + \bar{B}_2(I - zB_1)^{-1}V(zI - U). \tag{2.36}$$

The matrix $\bar{M}(z)$ has the same eigenvalues as $M(z)$, but, by incorporating the order conditions (2.33) and (2.34), the A_1 and A_2 matrices have been eliminated. The problem of finding a DIMSEM has now been reduced to

- picking values for the free parameters λ_i, c_i, and $\bar{B}_2$, so that $\bar{M}(z)$ has $r-1$ zero eigenvalues, and
- recovering the remaining method coefficients using $B_2 = Q\bar{B}_2$ and the order conditions

$$A_1 = (V - B_1 W)Q^{-1} \tag{2.37}$$

$$A_2 = (QP - B_2 W)Q^{-1}. \tag{2.38}$$

The resulting method (if it exists and is stable) has order at least $r-1$ and stage order at least $r-1$.

2.5 A Class of L-stable Diagonal Methods (LDMs)

We now describe an interesting class of methods which, although not DIMSEMs, are singly strictly diagonally implicit and L-stable. (In Butcher's classification, Fig. 2.1.1, they are type-4 DIMSIMs.)

If $\lambda_i = \lambda$ for $i = 1, \ldots, s$, then the transformed stability matrix (2.36) simplifies to

$$\bar{M}(z) = P + \frac{1}{1-\lambda z} T(zI - U),$$

where $T = \bar{B}_2 V$. We choose T below and then recover $\bar{B}_2 = TV^{-1}$. This requires that $r = s$ and the c_i's are distinct so that V is nonsingular. If we let T be upper triangular, $\bar{M}(z)$ is also upper triangular, and so the eigenvalues are simply the diagonal elements

$$e_i(z) = 1 + \frac{T_{ii} z}{1-\lambda z}, \quad i = 1, \ldots, r. \tag{2.39}$$

For $\lambda \in \mathbb{R}^+$ and any choice of

$$T_{ii} \in (0, 2\lambda), \quad i = 1, \ldots, r, \tag{2.40}$$

it follows that

$$|e_i(z)| < 1 \quad \forall z \in \mathbb{C}^- \setminus \{0\}, \quad i = 1, \ldots, r.$$

However, $e_i(0) = 1$ for $i = 1, \ldots, r$, so the method will be A-stable if and only if $\bar{M}(0)$ is

diagonalisable. If we choose

$$T_{ij} = \frac{1}{j} \begin{pmatrix} j \\ i-1 \end{pmatrix}, \quad i = 1, \ldots, r-1, \quad j = i, \ldots, r-1, \tag{2.41}$$

then $\bar{M}(0) = P - TU = I$ is diagonal. Moreover, since $T_{ii} = 1$ for $i = 1, \ldots, r-1$, (2.40) is satisfied if

$$\lambda > \max(1/2, T_{rr}/2).$$

Thus, we have found a class of A-stable, singly strictly diagonally implicit GLMs of order and stage order $r-1$ for any $r \geq 1$. The c_i's, λ, T_{rr}, and q are still free. If we choose $\lambda = T_{rr} = 1$, then (2.39) reduces to

$$e_i(z) = \frac{1}{1-z}, \quad i = 1, \ldots, r,$$

from which it follows immediately that $|e_i(z)| \to 0$ as $|z| \to \infty$. This L-stable class has the c_i's and q free.

Interestingly, recovering the original coefficient matrix A_2 using (2.38) gives

$$\begin{aligned} A_2 &= (QP - Q\bar{B}_2 W)Q^{-1} \\ &= (QP - QTV^{-1}VU)Q^{-1} \\ &= Q(P - TU)Q^{-1} \\ &= QIQ^{-1} \\ &= I. \end{aligned}$$

This is also the design choice made in the DIMSIMs of [11, 12, 13, 14], of which the type-4 variants are also singly strictly diagonally implicit. The DIMSIMs however have higher orders and the remaining coefficients are chosen differently.

2.6 A Class of DIMSEMs

This section contains the main contribution of this thesis. We introduce two theorems — Theorems 4 and 5 — which define a class of DIMSEMs of order $r-1$, for any $r \geq 2$. (The proofs of these theorems, because of their length, appear separately in Appendices A and B, respectively.) The choice of method coefficients made in Theorem 4 arose as a

generalisation of patterns observed in methods of orders 2–6, which were derived by solving the zero-eigenvalue equations in Maple. The computational feasibility of solving these equations depended on the simplification achieved by taking all the internal abscissae c_i's to be equal. Although this results in the Vandermonde matrix V being singular, this choice is acceptable, since our technique described in Section 2.4 for determining the coefficients of a GLM satisfying the zero-eigenvalue equations does not require V to be nonsingular.

We initially applied this somewhat unusual choice of parameters with all the c_i's equal to one, resulting in the creation of A-stable DIMSEMs up to order four, but greater success was obtained by setting all the c_i's to zero. In this case, the Vandermonde matrix V simplifies to

$$V = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \vdots & & \vdots & \\ 1 & 0 & \cdots & 0 \end{bmatrix},$$

so that the quantity $V(zI - U)$ appearing in the transformed stability matrix

$$\bar{M}(z) = P + \bar{B}_2(I - zB_1)^{-1}V(zI - U)$$

becomes

$$V(zI - U) = \begin{bmatrix} z & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \\ z & -1 & 0 & \cdots & 0 \end{bmatrix}.$$

The transformed stability matrix itself becomes

$$\bar{M}(z) = P + \begin{bmatrix} za_1(z) & -a_1(z) & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \\ za_r(z) & -a_r(z) & 0 & \cdots & 0 \end{bmatrix}, \tag{2.42}$$

where

$$a_i(z) = \sum_{k=1}^{s} \frac{b_{ik}}{1 - \lambda_k z}, \quad i = 1, \ldots, r, \tag{2.43}$$

$b_{ik} = (\bar{B}_2)_{ik}$ and $B_1 = \text{diag}(\lambda_1, \ldots, \lambda_s)$. This form of $\bar{M}(z)$ is simple enough to enable direct algebraic computation of the characteristic polynomial up to $r = 5$ (corresponding to order $r - 1 = 4$) using Maple. The advantage of this technique, however, is that relationships observed between elements of the $\bar{B}_2$ matrix of the resulting DIMSEMs can be generalised to permit the derivation of methods of higher order.

When solving the zero-eigenvalue equations using Maple, certain patterns were observed. When $s = r - 1$, it appeared that there was always a unique solution which leaves the λ_i's free. There also appeared to be a direct recurrence expressing the rest of the $\bar{B}_2$ matrix in terms of its last row. These observations gave rise to Theorem 4, the proof of which relies on an expression for the characteristic polynomial of the transformed stability matrix given by Theorem 5.

Theorem 4 *Let* $b_{ij} = (\bar{B}_2)_{ij}$. *Let*

$$s = r - 1 \geq 1,$$

$$c_i = 0, \quad i = 1, \ldots, s,$$

$$\begin{aligned} b_{rj} &= \frac{1}{s!} \frac{\sum_{k=0}^{s-1} \frac{\theta_{kj}^{(s)}}{(s-k)!}}{\prod_{\substack{l=1 \\ l \neq j}}^{s} (\lambda_l - \lambda_j)} \\ &= \frac{1}{s! \lambda_j^{s-1} p_j(\frac{1}{\lambda_j})} \sum_{k=0}^{s-1} \frac{\theta_{kj}^{(s)}}{(s-k)!}, \end{aligned}$$

where

$$\theta_{kj}^{(s)} \text{ is the coefficient of } x^k \text{ in } p_j(x)$$

and

$$p_j(x) = \prod_{\substack{l=1 \\ l \neq j}}^{s} (\lambda_l x - 1).$$

Let

$$b_{ij} = T_{ij}(\lambda) b_{rj}, \quad i = 1, \ldots, r-1,$$

where $\lambda = (\lambda_1, \ldots, \lambda_s)$ *with* $\lambda_i, i = 1, \ldots, s$, *distinct and nonzero,*

$$T_{ij}(\lambda) = \frac{1}{(i-1)!} \sum_{l=0}^{r-i} \left[(i-1+l)! \gamma_{s-i-l+1}^{(s)} \lambda_j^l \right],$$

and

$$\gamma_k^{(s)} = \begin{cases} \sum_{1 \leq i_1 < \cdots < i_k \leq s} i_1 \cdots i_k, & k \geq 1, \\ 1, & k = 0. \end{cases}$$

Then the transformed stability matrix $\bar{M}(z)$ *has* $r - 1$ *zero eigenvalues for all* $z \neq 1/\lambda_i$, $i = 1, \ldots, s$.

Remark. The requirement that $z \neq 1/\lambda_i$ for $i = 1, \ldots, s$ is a necessary restriction, since the factor $(I - zB_1)$ that occurs in both $M(z)$ and $\bar{M}(z)$ is singular if and only if $z = 1/\lambda_i$ for some $i = 1, \ldots, s$. Note that, if we choose $\lambda_i > 0$ for $i = 1, \ldots, s$, then $M(z)$ and $\bar{M}(z)$ are well defined and the single eigenvalue property holds for all $z \in \mathbb{C}^-$.

Proof. See Appendix A.

Theorem 5 *Assume $z \neq 1/\lambda_i$ for $i = 1, \ldots, s$ and let*

$$\phi_r(w, z) = \det(wI - \bar{M}_r(z)),$$

where $\bar{M}_r(z) = \bar{M}(z)$ defined in (2.36). Let

$$s = r - 1 \geq 1,$$

$$c_i = 0, \quad i = 1, \ldots, s.$$

Then

$$\phi_r(w, z) = \sum_{k=0}^{r} \phi_k^{(r)}(z) w^k,$$

where

$$\phi_k^{(r)}(z) = \begin{cases} \delta_k^{(s)} + \sum_{i=1}^{r} \left[(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z) a_i(z) \right], & k = 0, \ldots, r-1, \\ 1, & k = r, \end{cases}$$

$$\delta_k^{(s)} = (-1)^{s+1-k} \begin{pmatrix} s+1 \\ k \end{pmatrix},$$

$$\alpha_{k,i}^{(s)} = \begin{cases} 0, & i = 1, \\ -(i-1)\beta_{k,i-1}^{(s)}, & i \geq 2, \end{cases}$$

$$\beta_{k,i}^{(s)} = \begin{cases} (-1)^{s+1-k} \begin{pmatrix} s \\ k \end{pmatrix}, & i = 1, \\ (-1)^{s-k} \left[\begin{pmatrix} s-1 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-1} (-1)^l \begin{pmatrix} s-l-1 \\ k \end{pmatrix} \Sigma_l^{i-1} \right], & i \geq 2, \end{cases}$$

$$a_i(z) = \sum_{k=1}^{s} \frac{b_{ik}}{1 - \lambda_k z},$$

and

$$\Sigma_l^n = \begin{cases} \sum_{1 \le i_1 < \cdots < i_l \le n-1} \binom{i_2}{i_1} \binom{i_3}{i_2} \cdots \binom{i_l}{i_{l-1}} \binom{n}{i_l}, & l \ge 1, \\ 1, & l = 0. \end{cases}$$

Remark. Again note that the requirement that $z \neq 1/\lambda_i$ for $i = 1, \ldots, s$ is a necessary restriction, since the factor $(I - zB_1)$ that occurs in both $M(z)$ and $\bar{M}(z)$ is singular if and only if $z = 1/\lambda_i$ for some $i = 1, \ldots, s$. If we choose $\lambda_i > 0$ for $i = 1, \ldots, s$, then $M(z)$ and $\bar{M}(z)$ are well defined and the single-eigenvalue property obtained by Theorem 4 holds for all $z \in \mathbb{C}^-$.

Proof. See Appendix B.

As mentioned in Section 2.3, the characteristic polynomial associated with a DIMSEM has the form

$$\begin{aligned} \varphi(w, z) &= \det(wI - M(z)) \\ &= w^r - e_1(z)w^{r-1} - \beta_{r-2}(z)w^{r-2} - \cdots - \beta_1(z)w - \beta_0(z), \end{aligned} \tag{2.44}$$

where

$$\beta_i(z) = \frac{P_i(z)}{Q_i(z)}, \quad i = 0, \ldots, r-2,$$

P_i and Q_i are polynomials and $e_1(z)$ is a rational function of z. The requirement that $M(z)$ have $r - 1$ zero eigenvalues is equivalent to $\beta_0(z) = \cdots = \beta_{r-2}(z) = 0$ for all z, which in turn is equivalent to $P_0(z) = \cdots = P_{r-2}(z) = 0$ for all z, which in turn is equivalent to having all the coefficients of $P_i(z)$ equal to zero for $i = 0, \ldots, r-2$. This last condition is what we call the zero-eigenvalue equations. Then $e_1(z)$ is the remaining nonzero eigenvalue of $M(z)$, so it governs the stability of the method.

A-stability is easy to check, even without computing $e_1(z)$ from the characteristic polynomial. This is important, because algebraic computation of the characteristic polynomial is too complicated to be feasible for high orders. From (2.42), (2.43), and the zero-eigenvalue property, it follows that

$$\begin{aligned} e_1(z) &= \mathsf{tr}(\bar{M}(z)) \\ &= r + za_1(z) - a_2(z) \\ &= r + \sum_{k=1}^{s} \frac{zb_{1k} - b_{2k}}{1 - \lambda_k z}. \end{aligned}$$

This can be expressed in the form

$$e_1(z) = \frac{P(z)}{Q(z)}, \tag{2.45}$$

and the associated E-polynomial (2.3) constructed. Recall that a sufficient condition for A-stability is that the coefficients of the E-polynomial are negative and the λ_i's are positive. It is sometimes possible to solve these inequalities directly for the remaining free parameters in the low-order methods. The procedure followed to construct L-stable DIMSEMs up to order 6 for numerical testing is described in the next two sections.

2.7 A Class of L-Stable DIMSEMs

Until now, it has not been necessary to specify the true-value function q. The only restrictions on q have been the simplifying assumption $A(r-1)$, which states that $q(t)$ depends only on the order of the tree t, up to order $r-1$, and the requirement that the matrix Q be nonsingular. Both of the common choices of true-value function, based on Nordsieck and multistep implementations, satisfy these requirements. Because our methods satisfy $A(r-1)$, we will write q_k for $q(t)$, where t is any tree of order $k \leq r-1$. Also, let e_k be the vector in $\mathbb{R}^r$ with 1 as the k-th component and 0 elsewhere. Then in Nordsieck-type methods (1.1), q is given by

$$q_k = e_{k+1}, \quad k = 0, \ldots, r-1,$$

so that

$$Q = I_r \in \mathbb{R}^{r \times r}.$$

On the other hand, for multistep GLMs,

$$q_k = d^k, \quad k = 0, \ldots, r-1, \tag{2.46}$$

where $d \in \mathbb{R}^r$ is the vector of external abscissae, and d^k is taken componentwise with $0^0 = 1$. The corresponding matrix Q is

$$Q = [d^0, d^1, \ldots, d^{r-1}] \in \mathbb{R}^{r \times r}.$$

This corresponds to the external stages approximating the solution at various points,

$$y_{n,i} \approx u(x_n + d_i h), \quad i = 1, \ldots, r.$$

In deriving specific methods from our classes for testing purposes, we chose the multistep-type true-value function, and used the same external abscissae

$$d_i = i - r, \quad i = 1, \ldots, r,$$

as used in MRKs. Unlike MRKs, however, our A_2 and B_2 are general matrices, and not the restricted forms (1.3) and (1.4).

The first step in the creation of the methods was the algebraic solution of the zero-eigenvalue equations using the techniques of the previous chapter. This produced classes of DIMSEMs of order $p = 2, \ldots, 6$, $p = r - 1 = s$, with the λ_i's as free parameters. Then the E-polynomial was formed and A- and L-stability investigated.

2.7.1 Choice of Free Parameters for the DIMSEMs

The DIMSEMs of order $s = r - 1$ defined by Theorem 4 have free parameters $\lambda_1, \ldots, \lambda_s$, where $B_1 = \text{diag}(\lambda_1, \ldots, \lambda_s)$. The values of $\lambda_1, \ldots, \lambda_s$ were chosen to make the methods L-stable and to achieve a good compromise between small error coefficients and reasonably sized entries in the coefficient matrices A_1, A_2, B_1 and B_2.

The error coefficient function $e(t)$ defined in (1.11) is a vector-valued function, the i-th component of which gives the local discretisation error associated with the tree t for the i-th external stage. For optimisation purposes, we used the quantities T_1, T_2 and T_3, where

$$T_k = \frac{1}{(p+k)!} \max \left\{ \| \, e(t) \, \|_\infty : \rho(t) = p + k \right\}, \quad k = 1, 2, 3, \ldots$$

and p is the order of the method. These are the principal, secondary, tertiary etc. normalised, maximum-norm error coefficients associated with the method. We developed a Maple program, incorporating a data structure for rooted trees, to compute algebraic expressions for the error coefficients. All computation involved in the choice of the free parameters was done algebraically using Maple.

The first step in the procedure was to determine λ_s in terms of $\lambda_1, \ldots, \lambda_{s-1}$ so that $\deg(P) < \deg(Q)$, where the one nonzero eigenvalue of $M(z)$ is $e_1(z) = P(z)/Q(z)$. This ensures that, if the method is A-stable, it will also be L-stable. The next step was

to heuristically choose values for $\lambda_3, \ldots, \lambda_{s-1}$, leaving at most λ_1 and λ_2 to be selected based on the 3-dimensional plots described below. To achieve A-stability, the region where all the coefficients of the E-polynomial (2.3) are negative, and the region where $\lambda_i > 0$, $i = 1, \ldots, s$, were plotted. The intersection of these regions indicates choices of the free parameters which result in an A-stable method, and hence an L-stable one, because of the earlier choice of λ_s. Then the error coefficients T_1, T_2 and T_3, and the maxcf function were plotted, where

$$\begin{aligned} \text{maxcf} &= \max\{\langle A_1\rangle, \langle B_1\rangle, \langle A_2\rangle, \langle B_2\rangle\}, \\ \langle A\rangle &= \{\max|A_{ij}| : i, j\}. \end{aligned} \tag{2.47}$$

Finally, a line in (λ_1,λ_2)-space was selected which appeared to pass through an L-stable region with a favourable compromise between error coefficients and maxcf. Plots of the various functions along this line were then made, from which the final values for λ_1 and λ_2 were chosen. Often, with the high-order methods, the initial heuristic choice of λ_i's did not permit all the objectives to be satisfied, in which case the λ_i's were modified and further plots made. This ad-hoc process was repeated as necessary until a suitable choice was found.

Fig. 2.2 shows a series of plots for the fifth order DIMSEM. Because they have been reduced from their original large size and rendered as monochrome contour plots instead of interactively viewable, colour, 3-dimensional surfaces, it is not likely that much quantitative information can be derived from them. They are included nevertheless to illustrate the parameter-selection process. The first three rows are contour plots with λ_1 on the vertical axis and λ_2 on the horizontal axis. In the first and second rows, $\lambda_1, \lambda_2 \in [0, 5]$, while in the third row (a magnified view of the second row), $\lambda_1, \lambda_2 \in [0, 1]$. For this method, T_1, T_2, and T_3 (first row) show only small variation, which is fortunate because keeping maxcf reasonable while achieving L-stability is quite restrictive on λ_1 and λ_2. The dark areas in the maxcf plots are steep increases in maxcf due to singularities in the method coefficients corresponding to a confluence of two λ_i's. The method is L-stable at the intersection of the shaded regions in the second and third plots in the third row. The value $\lambda_2 = 0.29$ was chosen as the line passing through the L-stable region as well as through an area of small maxcf. Both plots in the last row have $\lambda_1 \in [0, 1]$ on the horizontal axis, while λ_2 is fixed at 0.29. The T_i's are the essentially constant lines in the plot on the left, while maxcf is the curved line with spikes. The bottom right plot shows the interval of λ_1 for which the method is L-stable. We chose $\lambda_1 = 0.37$ to obtain the smallest maxcf while achieving L-stability.

order	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	T_1	T_2	T_3	T_4	T_5	T_6	maxcf
2	.49	$\frac{1}{51}$					.33	.33	.26	.12	.045	.020	2
3	.25	.43	$\frac{79}{87}$				.44	.51	.44	.27	.14	.13	5
4	1.3	.27	1.0	$\frac{1337}{2710}$			.46	.87	.95	.80	.92		8
5	.37	.29	.25	.50	$\frac{654}{4981}$		.50	1.1	1.4	1.3			85
6	.71	.095	.25	.50	.75	$\frac{5086}{15879}$	.53	1.4	2.1				33

Table 2.1: Choice of Free Parameters and Error Coefficients for the DIMSEM Class.

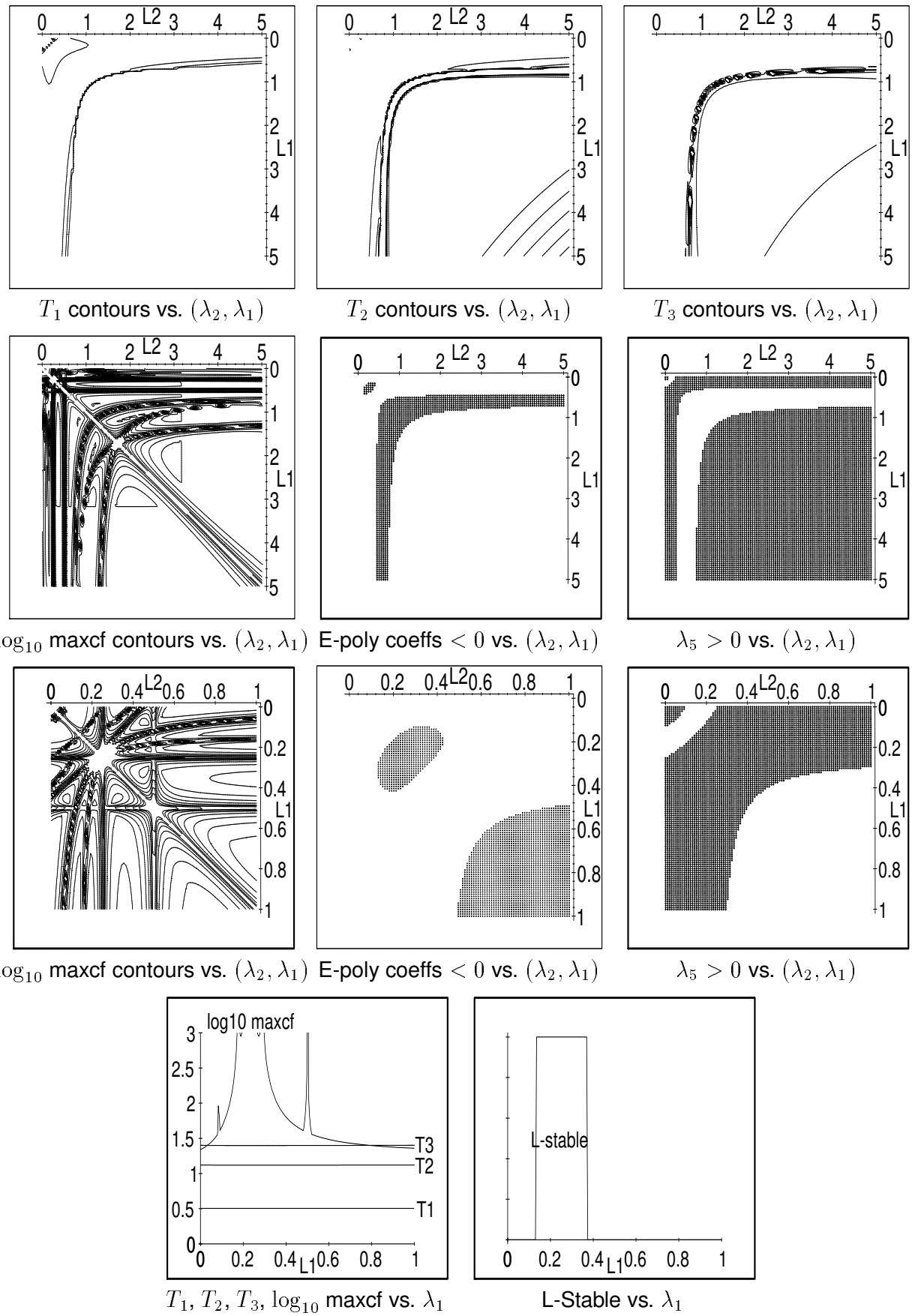
The chosen values for the free parameters, along with the resulting error coefficients and maxcf (2.47) are given in Table 2.1. The λ_i's shown are exact, while the T_i's and maxcf are rounded. (Some of the T_i's are omitted because they were too expensive to compute.)

The error coefficients of the L-stable DIMSEMs were then compared with those of two other classes of A-stable methods, the MRKs of Burrage and Chipman [6] and the Gauss and Radau-IA IRKs (see, for example, [10]). The MRKs are implicit with a lower-triangular B_1 matrix, while the Gauss/Radau formulas are fully implicit Runge-Kutta methods. The Gauss formulas have order $2s$ and are used for orders 2, 4 and 6, while the Radau-IA formulas have order $2s - 1$ and are used for orders 3 and 5. In Burrage and Chipman's methods, the choices of the free parameters used were those given on page 410 of their paper, which were used in their numerical testing. The values are shown in Table 2.2. Most of the error coefficients of the DIMSEMs are an improvement over those of the MRKs, and the ones which are not are only slightly worse. The improvements in T_2 and T_3 are quite significant. Both the DIMSEMs and MRKs have much larger error coefficients than the corresponding Gauss or Radau IRKs, however.

Table 2.3 compares the error coefficients of the DIMSEMs and the Backward Differentiation Formulas (BDFs) [20, p. 311]. (These formulas are used by the integrator LSODE, to which we compare the performance of the DIMSEMs in Chapter 3.) The BDF error coefficients are generally smaller, though comparable, to those of the DIMSEMs. The BDFs of order 3 and up are not A-stable however.

2.8 Choice of Free Parameters for the LDMs

LDMs of order $r - 1$ with $r = s$ are described in Section 2.5. Like the DIMSEMs derived in the previous chapter, the specific LDMs given in this chapter are based on a multivalue-type true-value function. The free parameters λ and T_{rr} are both set to 1, giving methods which



T_1 contours vs. (λ_2, λ_1) T_2 contours vs. (λ_2, λ_1) T_3 contours vs. (λ_2, λ_1)

$\log_{10}$ maxcf contours vs. (λ_2, λ_1) E-poly coeffs < 0 vs. (λ_2, λ_1) $\lambda_5 > 0$ vs. (λ_2, λ_1)

$\log_{10}$ maxcf contours vs. (λ_2, λ_1) E-poly coeffs < 0 vs. (λ_2, λ_1) $\lambda_5 > 0$ vs. (λ_2, λ_1)

$T_1, T_2, T_3, \log_{10}$ maxcf vs. λ_1 L-Stable vs. λ_1

Figure 2.2: Graphs for Choosing the Free Parameters for the Fifth Order DIMSEM.

	DIMSEMs					Burrage's MRKs					Gauss/Radau-IA IRKs				
order	r	s	T_1	T_2	T_3	r	s	T_1	T_2	T_3	r	s	T_1	T_2	T_3
2	3	2	.33	.33	.26	2	1	.33	.33	.23	1	1	.083	.17	.18
3	4	3	.44	.51	.44	2	2	.37	1.1	2.1	1	2	.014	.031	.037
4	5	4	.46	.87	.95	3	2	.45	1.4	2.8	1	2	1.3E-3	2.8E-3	2.9E-3
5	6	5	.50	1.1	1.4	4	3	1.4	9.7	39	1	3	1.4E-4	3.0E-4	3.3E-4
6	7	6	.53	1.4	2.1	5	4	.49	5.3	30	1	3	9.9E-6	2.0E-5	7.9E-5

Table 2.2: Comparison of Error Coefficients of DIMSEMs, Burrage and Chipman's MRKs, and Gauss/Radau-IA IRKs.

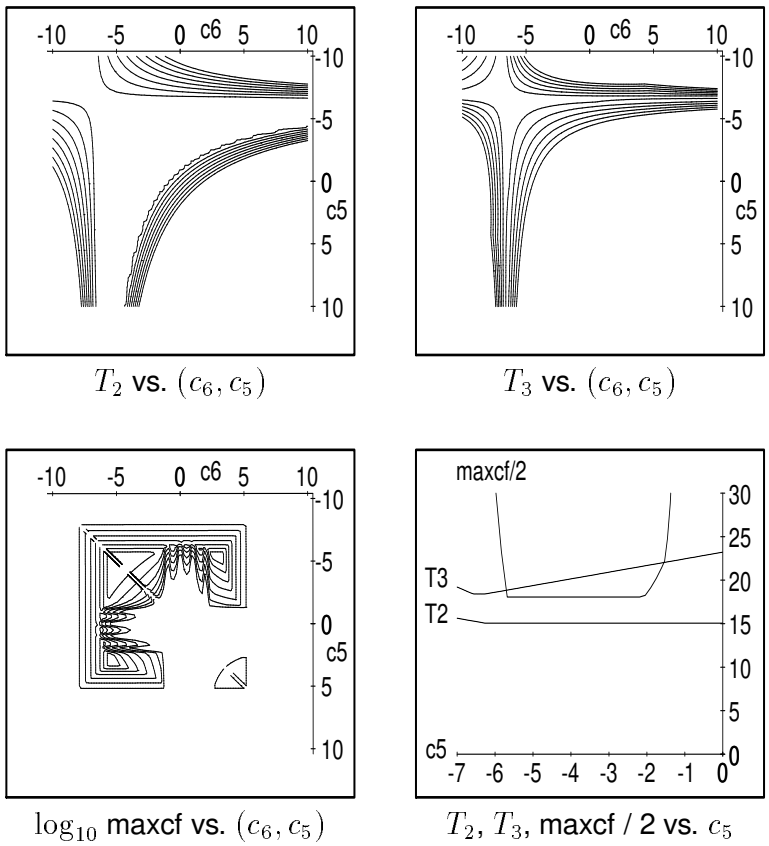
	DIMSEMs					BDFs				
order	r	s	T_1	T_2	T_3	r	s	T_1	T_2	T_3
2	3	2	.33	.33	.26	2	1	.22	.43	.47
3	4	3	.44	.51	.44	3	1	.14	.32	.41
4	5	4	.46	.87	.95	4	1	.096	.27	.41
5	6	5	.50	1.1	1.4	5	1	.073	.24	.42
6	7	6	.53	1.4	2.1	6	1	.058	.22	.44

Table 2.3: Comparison of Error Coefficients of DIMSEMs and BDFs.

are L-stable for all values of $c_1, \ldots, c_s$. It turns out that the principal error coefficients, and hence T_1, are fixed for each method, independent of the free parameters. The selection of the c_i's was performed in a manner similar to that of the λ_i's for the DIMSEMs (Section 2.7.1), except that only T_2, T_3, and the maxcf function (2.47) had to be considered due to the fixed T_1 and inherent L-stability.

To illustrate the process, Fig. 2.3 shows a series of plots for the fifth order LDM. The first three are contour plots with c_5 on the vertical axis and c_6 on the horizontal axis, covering the interval $[-10, 10]$. The contours in T_2 and T_3 show hills rising from a central valley. The line $c_6 = -5.7$ was chosen as passing near the minima of the T_i's and maxcf. The final plot has $c_5 \in [-7, 0]$ on the horizontal axis with $c_6 = -5.7$. The U-shaped function is maxcf/2 and the others are T_2 and T_3. We chose $c_5 = -5.67$ to minimise (roughly) T_2 and T_3 in the plateau region of maxcf.

The results of the optimisation are shown in Table 2.4. The values given for the c_i's are exact, while those for the T_i's are rounded. (Again, some of the T_i's are omitted because they were too expensive to compute.) Observe that the error coefficients of these LDMs are large in comparison to the other methods considered. The size of the coefficient matrix



T_2 vs. (c_6, c_5) T_3 vs. (c_6, c_5)

$\log_{10}$ maxcf vs. (c_6, c_5) T_2, T_3, maxcf / 2 vs. c_5

Figure 2.3: Graphs for Choosing the Free Parameters of the Fifth Order LDM.

entries, maxcf, also seems to increase more quickly with the order of the method for LDMs than for DIMSEMs. This might limit the maximum order of practical methods of this class, unless a way is discovered to reduce this coefficient growth. Nevertheless, the LDM methods found perform well in numerical testing.

In the interest of reducing the error coefficients, the c_i's were allowed to range outside the usual interval $[0, 1]$. This would have to be considered in a practical implementation, where problems with singularities outside the interval of integration may be encountered.

order	c_1	c_2	c_3	c_4	c_5	c_6	c_7	T_1	T_2	T_3	T_4	T_5	T_6	maxcf
2	0	1	-1.6					.83	.63	.49	.43	.37	.39	2
3	0	1	-2.26	-1.2				1.8	.92	1.36	.96	.65	.57	5
4	0	1	-1	-2.85	-3.81			4.2	5.6	5.3	3.7	2.6		10
5	0	1	-1	2	-5.67	-5.7		10	15	19	15			36
6	0	1	-1	2	-2	-6.66	-3	25	50	68				224

Table 2.4: Choice of Free Parameters and Error Coefficients for the LDM Class.

2.9 Fixed-Stepsize Numerical Testing

Preliminary fixed-stepsize numerical testing was performed to verify the accuracy and stability of the methods derived in the previous chapters and guard against errors in the theoretical work.

The integrations were started with the exact solution to the problem, and both local and global discretisation errors were measured. The test problems were integrated several times, halving the stepsize each time, and the observed (local or global) order $\tilde{p}$ was computed from each pair of consecutive runs using the formula

$$\tilde{p} = \log_2(e_{2h}/e_h),$$

where e_h is the (local or global) discretisation error corresponding to stepsize h. The maximum over all n of $||l_n||_\infty$, where l_n is defined in (1.10), was reported as the local discretisation error. This measurement of the local discretisation error was used only to verify agreement with the theory, and is not possible in a practical solver as it requires an analytic solution.

Two types of test problem were used. The first is a nonlinear problem chosen for the presence of all the elementary differentials $D(t)$ in the Butcher series, as well as the availability of an analytic solution. It is

$$y' = \omega \sin y,$$

which for the initial condition $y(x_0) = y_0$ satisfies

$$y(x) = 2 \arctan\left(e^{\omega(x-x_0)} \tan\frac{y_0}{2}\right).$$

Initial condition $y(0) = 1$ and $\omega = -1$ were used. The problem was integrated over the intervals $[0, 1]$ and $[0, 100]$, with stepsizes 2^k, $k = -11, \ldots, 1$, (approximately 5×10^{-4} to 2).

On this problem, all the DIMSEMs and LDMs produce the expected local order r and at least the expected global order $r - 1$. (Although the order 5 and 6 DIMSEMs show a global order equal to the local order, this order increase does not occur in general.) Over the interval $[0, 100]$, the endpoint global errors stay small, demonstrating stability.

The second test problem is that of Prothero and Robinson [27],

$$y' = g'(x) + \omega \left(y - g(x) \right),$$

where g is a given function. With the initial condition $y(0) = g(0)$, it has the solution $y(x) = g(x)$, but can be made arbitrarily stiff by choosing $\omega \ll 0$. We used $g(x) = \sin(x)$ and $\omega = -10^6$, making the problem stiff but non-dissipative, allowing the computation of meaningful apparent orders. The problem was integrated over the interval $[0, 10]$, with stepsizes 2^k, $k = -14, \ldots, 0$, (approximately 6×10^{-5} to 1). This problem is used to investigate the phenomenon of order-reduction, which tends to make methods perform with less than their asymptotic order when larger stepsizes are used. There is evidence that the observed order of a GLM can be expected to approximate its stage order [6].

On this problem, all the DIMSEMs and LDMs produce the expected local order r, and global order r also, one higher than achieved in general. There is no evidence of order reduction, which agrees with the theory since the order and stage order of the methods are equal.

Chapter 3

Variable-Stepsize Implementation and Testing

A variable-stepsize implementation of our methods is desirable to permit performance testing under realistic conditions, and to allow comparison with existing methods. This chapter develops the components of the variable-stepsize implementation, and gives test results on sequential and parallel machines.

3.1 Stepsize Change Scheme

We have until now assumed a fixed-stepsize formulation for our GLMs, that is,

$$\begin{aligned} Y_n &= A_1 y_n + h_n B_1 F(Y_n), \\ y_{n+1} &= A_2 y_n + h_n B_2 F(Y_n), \end{aligned} \tag{3.1}$$

with $h_n = h$ for all n. If we desire h_n to vary, the formula (3.1) no longer directly applies, as the order conditions were derived under the assumption of a fixed stepsize. Rather than re-deriving the coefficient matrices to depend on a variable stepsize, we will use an interpolant to adjust the external stage vector prior to taking a step with a new stepsize. Assuming a multistep-type true value function, y_n, produced using stepsize h_{n-1}, satisfies

$$y_{n,i} \approx y(x_n + d_i h_{n-1}), \quad i = 1, \ldots, r. \tag{3.2}$$

However, to compute y_{n+1} using (3.1), we require that

$$y_{n,i} \approx y(x_n + d_i h_n), \quad i = 1, \ldots, r.$$

We therefore interpolate the data values associated with y_n, evaluate this interpolant at $x_n + d_i h_n$, $i = 1, \ldots, r$, producing a new external stage vector $\bar{y}_n$ satisfying

$$\bar{y}_{n,i} \approx y(x_n + d_i h_n), \quad i = 1, \ldots, r, \tag{3.3}$$

and then replace y_n with $\bar{y}_n$ in (3.1). (A corresponding transformation exists for Nordsieck-type GLMs.) It remains to specify the interpolant.

Since the DIMSEMs are locally order r, we derive a polynomial interpolant of degree $r-1$, hence of order r. (There is another case, discussed later, in which an order $r+1$ interpolant is desirable.) Given y_n satisfying (3.2), we define a polynomial p of degree $r-1$ such that

$$p(x_n + d_i h_{n-1}) = y_{n,i}, \quad i = 1, \ldots, r.$$

A useful representation of p is

$$p(x_n + \tau h_{n-1}) = \sum_{i=1}^{r} l_i(x_n + \tau h_{n-1}) y_{n,i},$$

where l_i, $i = 1, \ldots, r$, are the Lagrange basis functions

$$\begin{aligned} l_i(x_n + \tau h_{n-1}) &= \frac{\prod_{j \neq i}(x_n + \tau h_{n-1} - (x_n + d_j h_{n-1}))}{\prod_{j \neq i}(x_n + d_i h_{n-1} - (x_n + d_j h_{n-1}))} \\ &= \frac{\prod_{j \neq i}(\tau - d_j)}{\prod_{j \neq i}(d_i - d_j)}. \end{aligned}$$

Then

$$\bar{y}_{n,i} = p(x_n + \tau_i h_{n-1})$$

satisfies (3.3) for

$$\tau_i = \omega_n d_i, \quad d_i h_n = \tau_i h_{n-1}, \text{ and } \omega_n = \frac{h_n}{h_{n-1}}.$$

Therefore we can write our variable-stepsize GLM as

$$\bar{y}_n = T(\omega_n) y_n, \tag{3.4}$$

$$Y_n = A_1 \bar{y}_n + h_n B_1 F(Y_n), \tag{3.5}$$

$$y_{n+1} \quad = \quad A_2\bar{y}_n + h_n B_2 F(Y_n), \tag{3.6}$$

where

$$T_{ij}(\omega_n) = \frac{\prod_{k\neq j}(\omega_n d_i - d_k)}{\prod_{k\neq j}(d_j - d_k)}. \tag{3.7}$$

3.2 Stability for Variable-Stepsize Methods

It is possible for arbitrary frequent stepsize changes to make a method asymptotically unstable, even though the underlying formula is stable in fixed-stepsize mode. It is therefore natural to restrict either the magnitude or the frequency of stepsize changes in order to guarantee stability. We now investigate each of these alternatives, beginning with the restriction on ω.

As $h_n \to 0$ in the variable-stepsize GLM (3.4-3.6), it turns out that stability on the stepsize sequence

$$h_n = \omega h_{n-1}, \quad n = 1, 2, \ldots,$$

follows if the spectral radius $\rho(A_2 T(\omega)) < 1$, provided a boundedness condition on f is assumed. (This can be seen from the proof of Theorem 6 below, with the modification that (3.23) is replaced by $(A_2 T(\omega))^{n-k}$, whose norm is bounded because $\rho(A_2 T(\omega)) < 1$.) Bounds on such ω may be computed, and values for ω^* such that the above condition is satisfied for all $\omega \in [0, \omega^*]$ are given in Table 3.1 for the DIMSEMs. (These values were arrived at by analytically verifying an estimate obtained by plotting the eigenvalues of $A_2 T(\omega)$ in Maple.) We see that as the order of the method increases, the allowable stepsize change ratio decreases. Unfortunately, the analysis above does not prove that the method is stable if

$$h_n = \omega_n h_{n-1}, \quad n = 1, 2, \ldots, \tag{3.8}$$

where $\omega_n \leq \omega^*$ for all n, which would be a more practical result. Note however that the restriction on ω_n in (3.8) is at least as stringent as ω^* in Table 3.1.

We now consider restricting the frequency rather than the magnitude of stepsize changes. In this case it is possible to prove a result guaranteeing stability.

Theorem 6 *Suppose the problem satisfies $|f(u)| \leq L|u|$ for all u, the variable-stepsize DIMSEM (3.4-3.6) has order at least one and the stepsize-change ratio $\omega_n = h_n/h_{n-1}$ is bounded above. Then the method is asymptotically stable provided at least $r-1$ steps are taken with the same stepsize before each stepsize change, where r is the number of*

order	ω^*
2	1.618
3	1.40
4	1.24
5	1.20
6	1.11

Table 3.1: Stability-Imposed ω Bounds for DIMSEMs

external stages.

Remark. The uniform bound on f is imposed to simplify the proof, but it is quite restrictive, excluding for example the stable problem $u' = -u^3$. A similar assumption (uniform Lipschitz continuity) is made by Skeel [29] when proving a corresponding result (Lemma 3.5) for fixed-stepsize methods. However, as pointed out in [20, p. 395], it is only necessary to assume the condition in a neighbourhood of the solution, in which case the assumption becomes reasonable. The cited references do not deal with this rigorously, though, nor does the proof given here.

Remark. It is not necessary to bound ω_n away from zero.

Proof. Let x_0 and x_N be fixed (although N will vary). Let M be the number of stepsize changes in stepping from x_0 to x_N,

$$h_i^* = \omega_i^* h_{i-1}^*, \quad i = 1, \ldots, M, \tag{3.9}$$

and

$$h_{m_i} = \cdots = h_{m_{i+1}-1} = h_i^*, \quad i = 0, \ldots, M-1, \tag{3.10}$$

where

$$\begin{aligned} m_0 &= 0, \\ m_{i+1} - m_i &\geq r - 1, \quad i = 0, \ldots, M-1. \end{aligned} \tag{3.11}$$

That is, the stepsize is changed at most once every r steps. Now let $h_i \to 0$ for $i = 0, \ldots, N-1$ and let $N \to \infty$. We need to show that $\|y_n\|$ stays bounded independent of n for all $n \leq N$.

We start by finding an expression for y_n. Equation (3.6) can be written as

$$y_{n+1} = S_n y_n + h_n \Psi_n(y_n),$$

where

$$\begin{aligned} S_n &= A_2 T(\omega_n), && (3.12)\\ \omega_n &= \begin{cases} \omega_i^* & \text{if } n = m_i \quad \text{for some } i, \\ 1 & \text{if } n \neq m_i, \quad i = 1, \ldots, M, \end{cases} && (3.13)\\ \Psi_n(y_n) &= B_2 F(Y_n). && (3.14) \end{aligned}$$

Then

$$\begin{aligned} y_1 &= S_0 y_0 + h_0 \Psi_0(y_0), \\ y_2 &= S_1 S_0 y_0 + h_0 S_1 \Psi_0(y_0) + h_1 \Psi_1(y_1), \\ &\vdots \\ y_n &= S_{n-1} \cdots S_0 y_0 + \sum_{k=0}^{n-1} h_k S_{n-1} \cdots S_{k+1} \Psi_k(y_k), \quad n = 1, \ldots, N. \end{aligned} \tag{3.15}$$

Now if we could find constants K_1 and K_2 such that

$$\|S_n \cdots S_k\| \leq K_1, \quad \forall k \leq n, \tag{3.16}$$

and

$$\|\Psi_k(y_k)\| \leq K_2 \|y_k\|, \quad \forall k, \tag{3.17}$$

we would have from (3.15),

$$\|y_n\| \leq K_1 \|y_0\| + K_1 K_2 \sum_{k=0}^{n-1} h_k \|y_k\|, \quad n = 1, \ldots, N. \tag{3.18}$$

Let

$$a_0 = \|y_0\| \tag{3.19}$$

and

$$a_n = K_1 a_0 + K_1 K_2 \sum_{k=0}^{n-1} h_k a_k, \quad n = 1, \ldots, N. \tag{3.20}$$

If we assume by induction that $\|y_k\| \leq a_k$ for $k = 0, \ldots, n-1$, then by (3.18) and (3.20)

$$\begin{aligned} \|y_n\| &\leq K_1 a_0 + K_1 K_2 \sum_{k=0}^{n-1} h_k a_k \\ &= a_n, \quad n = 1, \ldots, N, \end{aligned}$$

and thus

$$\|y_n\| \leq a_n, \quad n = 0, \ldots, N.$$

It then suffices to show a_n is bounded independent of n for asymptotic stability. From (3.20),

$$\begin{aligned} a_n &= K_1 a_0 + K_1 K_2 \sum_{k=0}^{n-1} h_k a_k, \\ a_{n-1} &= K_1 a_0 + K_1 K_2 \sum_{k=0}^{n-2} h_k a_k, \end{aligned}$$

so that

$$a_n - a_{n-1} = K_1 K_2 h_{n-1} a_{n-1},$$

and then

$$\begin{aligned} a_n &= a_{n-1} + K_1 K_2 h_{n-1} a_{n-1} \\ &= (1 + K_1 K_2 h_{n-1}) a_{n-1}. \end{aligned}$$

Solving the recurrence gives

$$a_n = a_0 \prod_{k=0}^{n-1} (1 + K_1 K_2 h_k). \tag{3.21}$$

Let

$$R = x_N - x_0 = \sum_{k=0}^{N-1} h_k \geq \sum_{k=0}^{n-1} h_k, \quad n \leq N. \tag{3.22}$$

Then R is independent of N since x_N is fixed. Now observe that

$$1 + x \leq 1 + x + \sum_{i=2}^{\infty} \frac{x^i}{i!} = e^x \quad \forall x \geq 0.$$

This, together with (3.21) and (3.22), implies

$$\begin{aligned} a_n &\leq a_0 \prod_{k=0}^{n-1} e^{K_1 K_2 h_k} \\ &= a_0 e^{\sum_{k=0}^{n-1} K_1 K_2 h_k} \\ &= a_0 e^{K_1 K_2 \sum_{k=0}^{n-1} h_k} \\ &= a_0 e^{K_1 K_2 R} \\ &= A, \end{aligned}$$

say. Hence $\|y_n\| \leq a_n \leq A$ for all n, and so the method is asymptotically stable. It remains to establish (3.16) and (3.17).

First consider (3.16). Define p and q in terms of k and n respectively by

$$m_p \leq k < m_{p+1}$$

and

$$m_q \leq n < m_{q+1},$$

where m_p and m_q are defined in (3.10). Let

$$\begin{aligned} L_{n,k} &= S_n \cdots S_k, \quad k \leq n \\ &= A_2 T(\omega_n) \cdots A_2 T(\omega_k) \quad \text{by (3.12)} \\ &= A_2 T(\omega_n) \cdots A_2 T(\omega_{m_q}) \\ &\quad A_2 T(\omega_{m_q - 1}) \cdots A_2 T(\omega_{m_{q-1}}) \\ &\quad \vdots \\ &\quad A_2 T(\omega_{m_{p+2}-1}) \cdots A_2 T(\omega_{m_{p+1}}) \\ &\quad A_2 T(\omega_{m_{p+1}-1}) \cdots A_2 T(\omega_k) \quad \text{by (3.13)} \\ &= A_2^{n-m_q+1} T(\omega_q^*) \\ &\quad A_2^{(m_q-1)-m_{q-1}+1} T(\omega_{q-1}^*) \\ &\quad \vdots \\ &\quad A_2^{(m_{p+2}-1)-m_{p+1}+1} T(\omega_{p+1}^*) \\ &\quad A_2^{(m_{p+1}-1)-k+1} T(\omega_k) \\ &= A_2^{n-m_q+1} T_q A_2^{\bar{m}_{q-1}} T_{q-1} \cdots A_2^{\bar{m}_{p+1}} T_{p+1} A_2^{m_{p+1}-k} T(\omega_k) \end{aligned}$$

$$= \quad U_n A_2^{\bar{m}_q - 1} T_{q-1} \cdots A_2^{\bar{m}_{p+1}} T_{p+1} V_k, \tag{3.23}$$

where

$$U_n = A_2^{n-m_q+1} T(\omega_q^*), \tag{3.24}$$

$$V_k = A_2^{m_{p+1}-k} T(\omega_k), \tag{3.25}$$

$$T_i = T(\omega_i^*), \quad i = p+1, \ldots, q-1, \tag{3.26}$$

$$\bar{m}_i = m_i - m_{i-1} \geq r-1, \quad i = p+1, \ldots, q-1. \tag{3.27}$$

Now the Jordan canonical form J of A_2 may be written

$$S^{-1} J S = A_2,$$

where

$$J = \text{diag}(J_i), \tag{3.28}$$

$$J_i = \gamma_i I + E_i, \quad E_i^{r_i} = 0, \quad i = 1, \ldots, l, \tag{3.29}$$

and $r_i \leq r-1$.

Although not necessary for this proof, the following conjecture may be of interest.

Conjecture 1 *For a DIMSEM, A_2 has only two Jordan blocks. That is, $J_1 = 1$ and $J_2 \in \mathbb{R}^{(r-1)\times(r-1)}$ is all zeroes except for a superdiagonal of 1's.*

Now $A_2 = S^{-1} J S$ implies

$$\begin{aligned} A_2^k &= (S^{-1}JS)(S^{-1}JS)\cdots(S^{-1}JS) \\ &= S^{-1} J^k S, \quad k \geq 0. \end{aligned} \tag{3.30}$$

From (3.28),

$$J^k = \begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_l \end{bmatrix}^k = \begin{bmatrix} J_1^k & & \\ & \ddots & \\ & & J_l^k \end{bmatrix}, \quad k \geq 0.$$

For a DIMSEM, the single-eigenvalue property implies $M(0) = A_2$ has $r-1$ zero eigenvalues, so we may without loss of generality choose S such that in (3.29),

$$\gamma_j = 0, \quad j = 2, \ldots, r.$$

We now show that $J_1 = 1$. Consider applying the DIMSEM to the problem $u' = \mu u$, $u(0) = 1$. Since the method has order at least one, (2.14) gives the local discretisation error as

$$l_n = M(z)\hat{y}_n - e^z \hat{y}_n = O(z),$$

where $z = h\mu$. Letting $z \to 0$ and using $M(0) = A_2$ gives

$$A_2 \hat{y}_n = \hat{y}_n.$$

By (2.46), $q(\phi) \neq 0$ and we have by (2.12) that $\hat{y}_n = q(\phi)u(x_n) + O(z)$, thus $\hat{y}_n \neq 0$ and A_2 has an eigenvalue 1 with corresponding eigenvector $\hat{y}_n$. Then, since all the other eigenvalues of A_2 are zero, we have $J_1 = 1$ as desired. Thus

$$J_1^{\bar{m}_i} = 1, \quad i = p+1, \ldots, q-1,$$

and, for $j > 1$, since $\bar{m}_i \geq r - 1$,

$$J_j^{\bar{m}_i} = (\gamma_j I + E_j)^{\bar{m}_i} = E_j^{\bar{m}_i} = 0, \quad i = p+1, \ldots, q-1.$$

Thus

$$J^{\bar{m}_i} = \begin{bmatrix} 1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{bmatrix} = E, \quad \text{say, for } i = p+1, \ldots, q-1.$$

Using this in (3.30), we have

$$A_2^{\bar{m}_i} = S^{-1} E S, \quad i = p+1, \ldots, q-1,$$

and so from (3.23),

$$\begin{aligned} L_{n,k} &= U_n (S^{-1}ES) T_{q-1} (S^{-1}ES) T_{q-2} \cdots (S^{-1}ES) T_{p+1} V_k \\ &= U_n S^{-1} E \bar{T}_{q-1} E \bar{T}_{q-2} \cdots E \bar{T}_{p+1} S V_k \\ &= U_n S^{-1} \tilde{L}_{n,k} S V_k, \end{aligned} \tag{3.31}$$

where

$$\bar{T}_i = S T_i S^{-1}, \quad i = p+1, \ldots, q-1, \tag{3.32}$$

and

$$\tilde{L}_{n,k} = E\bar{T}_{q-1}E\bar{T}_{q-2}\cdots E\bar{T}_{p+1}.$$

Consider the components of $\tilde{L}_{n,k}$.

$$E\bar{T}_i = \begin{bmatrix} 1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{bmatrix} \bar{T}_i = \begin{bmatrix} (\bar{T}_i)_{\text{row 1}} \\ 0 \end{bmatrix},$$

so that

$$E\bar{T}_{q-1}E\bar{T}_{q-2} = \begin{bmatrix} (\bar{T}_{q-1})_{\text{row 1}} \\ 0 \end{bmatrix} \begin{bmatrix} (\bar{T}_{q-2})_{\text{row 1}} \\ 0 \end{bmatrix} = (\bar{T}_{q-1})_{11} \begin{bmatrix} (\bar{T}_{q-2})_{\text{row 1}} \\ 0 \end{bmatrix},$$

and finally,

$$\begin{aligned} \tilde{L}_{n,k} &= (\bar{T}_{q-1})_{11}\cdots(\bar{T}_{p+2})_{11} \begin{bmatrix} (\bar{T}_{p+1})_{\text{row 1}} \\ 0 \end{bmatrix} \\ &= (\bar{T}_{q-1})_{11}\cdots(\bar{T}_{p+2})_{11}E\bar{T}_{p+1}. \end{aligned} \tag{3.33}$$

We now investigate $(\bar{T}_i)_{11}$. From (3.32),

$$\bar{T}_i = ST_iS^{-1}.$$

Let $u = S_{\text{row 1}}$ and $v = S^{-1}_{\text{col 1}}$, so that

$$(\bar{T}_i)_{11} = uT_iv. \tag{3.34}$$

We now show that the components of v are all the same.

Since the method has order at least one, the order condition $B(w)$ in (1.13) is necessarily satisfied for $w = 0$, which implies that $A_2q_0 = q_0$. By (2.46), however, $q_0 = e$, where $e = (1,\ldots,1)^T$. That is, e is an eigenvector of A_2 corresponding to the eigenvalue 1. But the eigenvalue 1 has Jordan block J_1 of size 1, so the eigenvector is unique up to a scalar multiple. Now

$$A_2 = S^{-1}JS \implies A_2S^{-1} = S^{-1}J$$

$$\begin{aligned}\implies \quad A_2 v = A_2 S^{-1}_{\text{col 1}} &= (A_2 S^{-1})_{\text{col 1}} = (S^{-1}J)_{\text{col 1}} = S^{-1}J_{\text{col 1}} \\ &= S^{-1}\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = S^{-1}_{\text{col 1}} = v,\end{aligned}$$

so that v is also an eigenvector of A_2 corresponding to the eigenvalue 1, and hence

$$v = ae$$

for some scalar a. In addition, $T_i e = e$ for all i, since any polynomial of degree $r-1$ interpolating the constant function 1 at r points must evaluate to 1 everywhere.

Continuing now from (3.34), we have

$$\begin{aligned}(\bar{T}_i)_{11} &= uT_i v = uT_i ae = uaT_i e = uae = uv \\ &= S_{\text{row 1}} S^{-1}_{\text{col 1}} = (SS^{-1})_{11} = I_{11} = 1.\end{aligned}$$

Hence, from (3.31) and (3.33),

$$L_{n,k} = U_n S^{-1} \tilde{L}_{n,k} S V_k = U_n S^{-1} E \bar{T}_{p+1} S V_k = U_n S^{-1} E S T_{p+1} V_k.$$

We now bound the norm of each of these matrices. $S^{-1}ES$ is bounded since it is independent of n and k. From (3.7), the entries of $T(\omega_i)$ are polynomials in ω_i, so $\|T(\omega_i)\|$ is bounded since ω_i is. It follows that $\|T_{p+1}\|$ is bounded also, since $T_{p+1} = T(\omega_i)$ for some i. From (3.24) we have

$$U_n = A_2^{n-m_q+1} T(\omega_q^*),$$

in which $\|T(\omega_q^*)\|$ is bounded by the above argument, and $\|A_2^{n-m_q+1}\|$ is also since A_2 has $r-1$ eigenvalues 0 and one eigenvalue 1, making it power bounded. From (3.25) we have

$$V_k = A_2^{m_{p+1}-k} T(\omega_k),$$

and $\|V_k\|$ is bounded in a manner similar to $\|U_n\|$. Therefore $\|L_{n,k}\|$ is bounded independent of n and k, completing the proof of (3.16).

It remains to prove (3.17). From (3.5) and the assumption that $\|f(y)\| \leq L\|y\|$, it follows

that

$$\|Y_k\| \leq \|A_1 T(\omega_k)\| \|y_k\| + h_k \|B_1\| \, L \, \|Y_k\|,$$

hence

$$(1 - h_k L \|B_1\|) \, \|Y_k\| \leq \|A_1 T(\omega_k)\| \|y_k\|.$$

For h_k sufficiently small, $h_k L \|B_1\| < 1$, so

$$\|Y_k\| \leq \frac{\|A_1 T(\omega_k)\|}{1 - h_k L \|B_1\|} \|y_k\|.$$

Then from (3.14) we have

$$\begin{aligned} \|\Psi_k(y_k)\| &\leq \|B_2\| \, L \, \|Y_k\| \\ &\leq \frac{\|B_2\| \, L \, \|A_1\| \|T(\omega_k)\|}{1 - h_k L \|B_1\|} \|y_k\|. \end{aligned}$$

As before, $\|T(\omega_k)\|$ is bounded, and the denominator of the fraction above approaches 1 as $h_k \to 0$ for all k. Therefore, there exists K_2 such that

$$\|\Psi_k(y_k)\| \leq K_2 \|y_k\| \quad \forall k,$$

and the proof is complete. □

Given a choice between limiting ω to the values in Table 3.1 or using the technique of Theorem 6, the theorem appears superior because it guarantees stability and the restriction seems less obtrusive. The practical consequences of each of the techniques are described later.

3.3 Local Error Estimators

So far we have developed a means to allow the method to vary its stepsize, assuming it knows what stepsize to use. This choice is made by a stepsize controller, based on feedback from a local error estimator, the details of which we consider next.

The construction of error estimators for DIMSEMs was more difficult than we expected. Although we found estimators that enabled us to conduct variable-stepsize testing, their robustness is not ideal and more work remains to be done in this area.

3.3.1 DIMSEM Principal Error Coefficients

It turns out that, for a DIMSEM, the principal term of the B-series expansion of the local discretisation error is independent of the particular tree. This result, which is formalised in the following theorem, reduces the task of estimating the local discretisation error to estimating the r-th derivative of the solution.

Theorem 7 *Consider a multistep-type GLM with order and stage order p, satisfying simplifying assumption $A(p+1)$. Let $e(t)$ be the vector of local error coefficients (1.11) for the tree t. Then $e(t)$ is the same for all trees t of order $p+1$.*

Remark. In particular, our DIMSEMs of order p satisfy the conditions of the theorem.

Before the theorem is proven, we need to introduce some more Butcher-series results, beyond those given in Section 1.3. We first analyse the error committed on one step of our GLM,

$$\begin{aligned} Y_n &= A_1 y_n + h B_1 F(Y_n) \in \mathbb{R}^s \\ y_{n+1} &= A_2 y_n + h B_2 F(Y_n) \in \mathbb{R}^r, \end{aligned}$$

using the theory of Butcher series. Let $\hat{y}_k$ be the vector of exact values to which the external stage vector y_k is an approximation, for every step k. By appropriate choice of the true-value function, $q : T \to \mathbb{R}^r$, we can expand $\hat{y}_k$ in a Butcher series around x_k,

$$\hat{y}_k = B(q, u(x_k)) \quad \forall k, \tag{3.35}$$

where $u : \mathbb{R} \to \mathbb{R}$ is the true solution to the IVP. Let $y_n = \hat{y}_n$, so that the method starts the n-th step with exact information. If our method is exact then we would expect $y_{n+1} = \hat{y}_{n+1}$, but as this is not typically the case we consider the difference

$$l_n = y_{n+1} - \hat{y}_{n+1} \tag{3.36}$$

to be the local discretisation error on the n-th step. For suitable $e : T \to R$ this can be written in B-series form as $l_n = B(e, u(x_n))$. The remainder of this section is devoted to the computation of e, which is the error coefficient function.

With a suitable choice of functions $Y : T \to \mathbb{R}^s$ and $y : T \to \mathbb{R}^r$, the internal stages Y_n and external stages y_{n+1} of our method can be expressed as Butcher series expanded

around x_n as follows:

$$Y_n = B(Y, u(x_n)),$$

$$y_{n+1} = B(y, u(x_n)). \tag{3.37}$$

Using the properties of Butcher series, Y and y can be found to obey

$$Y(t) = A_1 q(t) + \rho(t) B_1 \prod_{i=1}^{m} Y(t_i) \tag{3.38}$$

and

$$y(t) = A_2 q(t) + \rho(t) B_2 \prod_{i=1}^{m} Y(t_i), \tag{3.39}$$

where

$$t = [t_1, \ldots, t_m]$$

and $\rho(t)$ is the order of the tree t. It follows easily from (1.8) that two B-series expanded about the same point can be added, that is,

$$B(a, v) + B(b, v) = B(a + b, v),$$

where $(a + b)(t) = a(t) + b(t)$ for all $t \in T$. This can be applied to (3.36) if we make use of the relationship

$$u(x_{n+1}) = B(1, u(x_n)), \quad 1(t) = 1 \quad \forall t \in T,$$

which also follows from (1.8). From (3.35) and (3.36), then, we have

$$\begin{aligned} l_n &= B(y, u(x_n)) - B(q, u(x_{n+1})) \\ &= B(y, u(x_n)) - B(q, B(1, u(x_n)) \\ &= B(y, u(x_n)) - B(\bar{q}, u(x_n)), \end{aligned} \tag{3.40}$$

for an appropriate $\bar{q}$. By a theorem of Hairer and Wanner [21] on the composition of Butcher series, it follows that

$$\bar{q}(t) = \sum_{s \subset t} \beta(s \subset t) q(s),$$

where

$$\beta(s \subset t) = \begin{pmatrix} \rho(t) \\ \rho(s) \end{pmatrix} \frac{\alpha(s \subset t)}{\alpha(t)},$$

$\alpha(t)$ is the number of ways of monotonically labeling tree t, and $\alpha(s \subset t)$ is the number of ways of monotonically labeling the tree-subtree pair $s \subset t$ with subtree s labeled first. (It is necessary to distinguish between tree-subtree pairs with the same tree and subtree, but having the subtree occur at different positions within the tree. For this purpose the notation $s \subset t$ for the tree-subtree pair, introduced in [21], is customary. It denotes a particular subtree s of t, sharing the same root.)

Continuing from (3.40), we have

$$\begin{aligned} l_n &= B(y, u(x_n) - B(\bar{q}, u(x_n)) \\ &= B(y - \bar{q}, u(x_n)) \\ &= B(e, u(x_n)), \end{aligned} \tag{3.41}$$

where

$$e(t) = y(t) - \bar{q}(t) \in \mathbb{R}^r \tag{3.42}$$

is the vector of error coefficients of the method for the tree t. This concludes the background, and we proceed to the proof of Theorem 7.

Proof. The proof consists of four parts:

1. Show that $Y(t) = c^{\rho(t)}$, $\rho(t) \leq p$, where 0^0 is taken as 1.
2. Show that $y(t)$ depends only on $\rho(t)$ for all trees with $\rho(t) \leq p + 1$.
3. Show that $\bar{q}(t) = (d + e)^{\rho(t)}$, where $e = (1, \ldots, 1)^T$.
4. Show that $e(t)$ depends only on $\rho(t)$ for all trees with $\rho(t) \leq p + 1$.

Part 1: Let the method start the step with $y_n = \hat{y}_n$, the exact values. Then stage order p implies that

$$Y_{n,i} = u(x_n + c_i h) + O(h^{p+1}), \quad i = 1, \ldots, s.$$

But

$$Y_n = B(Y, u(x_n)),$$

so that Y must satisfy

$$Y(t) = c^{\rho(t)}, \quad \rho(t) \leq p.$$

Part 2:

$$y(t) = A_1 q(t) + \rho(t) B_1 \prod_{i=1}^{k} Y(t_i), \quad t = [t_1, \ldots, t_k],$$

by (3.39). Let $\rho(t) \leq p+1$. Then $\rho(t_i) \leq p$, and by Part 1, $Y(t_i) = c^{\rho(t_i)}$. Therefore

$$\begin{aligned} y(t) &= A_1 q(t) + \rho(t) B_1 \prod_{i=1}^{k} c^{\rho(t_i)} \\ &= A_1 q(t) + \rho(t) B_1 c^{\sum_{i=1}^{k} \rho(t_i)} \\ &= A_1 q(t) + \rho(t) B_1 c^{\rho(t)-1}, \end{aligned}$$

which depends only on $\rho(t)$, since $q(t)$ does by $A(p+1)$.

Part3:

$$\bar{q}(t) = \sum_{s \subset t} \left(\begin{array}{c} \rho(t) \\ \rho(s) \end{array} \right) \frac{\alpha(s \subset t)}{\alpha(t)} q(s)$$

by (3.39), where

$$q(s) = d^{\rho(s)}.$$

Let $n = \rho(t)$. Then

$$\begin{aligned} \bar{q}(t) &= \sum_{k=0}^{n} \sum_{\substack{s \subset t \\ \rho(s)=k}} \left(\begin{array}{c} n \\ k \end{array} \right) \frac{\alpha(s \subset t)}{\alpha(t)} d^k \\ &= \sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) d^k \sum_{\substack{s \subset t \\ \rho(s)=k}} \frac{\alpha(s \subset t)}{\alpha(t)} \\ &= \sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) d^k \frac{1}{\alpha(t)} \sum_{\substack{s \subset t \\ \rho(s)=k}} \alpha(s \subset t) \\ &= \sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) d^k, \end{aligned} \tag{3.43}$$

since

$$\sum_{\substack{s \subset t \\ \rho(s)=k}} \alpha(s \subset t) = \alpha(t).$$

But (3.43) = $(d+e)^n$, where $e = (1, \ldots, 1)^T$, proving Part 3.

Part 4: Since $e(t) = y(t) - \bar{q}(t)$, this follows immediately from Parts 2 and 3, completing the proof of the theorem. □

3.3.2 Difference-Type Local Error Estimate

Theorem 7 provides the basis for one type of local error estimate. The local discretisation error vector l_n satisfies

$$\begin{aligned} l_n &= B(e, u(x_n)) \\ &= \sum_{t \in T} e(t) \frac{h^{\rho(t)}}{\rho(t)!} D(t)(u(x_n)). \end{aligned}$$

Consider a DIMSEM with local discretisation error of order $p+1$. Then $e(t) = 0$ for all trees with $\rho(t) \leq p$. However, by Theorem 7, there is a vector e_{p+1} satisfying $e(t) = e_{p+1}$ for all trees t of order $p+1$. Thus

$$\begin{aligned} l_n &= \sum_{\rho(t) \geq p+1} e(t) \frac{h^{\rho(t)}}{\rho(t)!} D(t)(u(x_n)) \\ &= \sum_{\rho(t) = p+1} e(t) \frac{h^{\rho(t)}}{\rho(t)!} D(t)(u(x_n)) + O(h^{p+2}) \\ &= e_{p+1} \frac{h^{p+1}}{(p+1)!} \sum_{\rho(t) = p+1} D(t)(u(x_n)) + O(h^{p+2}) \\ &= e_{p+1} \frac{h^{p+1}}{(p+1)!} u^{(p+1)}(x_n) + O(h^{p+2}). \end{aligned} \tag{3.44}$$

Since e_{p+1} can be computed for any particular method using (3.42), (3.44) gives an estimate of the local discretisation error, provided that $u^{(p+1)}(x_n)$ can be estimated. This may be done using a difference approximation, which we consider next.

The k-th order difference $\Delta^k u_n$ is defined for the function u by

$$\Delta^k u_n = \sum_{i=0}^{k} (-1)^i \begin{pmatrix} k \\ i \end{pmatrix} u(x_{n+k-i}), \tag{3.45}$$

where $x_j = x_0 + jh$, $j = n, \ldots, n+k$, and h is some fixed spacing. If $u^{(k)}(x)$ exists and is continuous for $x \in [x_n, x_{n+k}]$, then

$$\Delta^k u_n = h^k u^{(k)}(\xi), \quad \xi \in [x_n, x_{n+k}],$$

and so $\Delta^k u_n$ can be used to obtain an approximation to $u^{(k)}(x_n)$. Suppose we have just completed a step, generating y_{n+1} from $\bar{y}_n$ using (3.6). For our local error estimate (3.44) we require an estimate of $u^{(r)}(x_{n+1})$. In order to apply (3.45), $r+1$ data points are needed,

but y_{n+1} contains only r values. One way to supply the extra point is to use the interpolated previous external stage vector $\bar{y}_n$ together with y_{n+1}. ($\bar{y}_n$ is used instead of y_n because (3.45) requires the data points to be equally spaced.) From (3.2) and (3.3), the stages used in the difference scheme can be seen to approximate

$$\begin{aligned} y_{n+1,1} &\approx u(x_{n+1} + d_1 h_n), \\ &\vdots \\ y_{n+1,r} &\approx u(x_{n+1} + d_r h_n), \\ \bar{y}_{n,r} &\approx u(x_n + d_r h_n), \end{aligned}$$

which are equidistant because $x_{n+1} = x_n + h_n$ and $d_i = r - i$, $i = 1, \ldots, r$. Using this in (3.45), we arrive at the difference approximation

$$h_n^r u^{(r)}(x_{n+1}) \approx D_n^{(r)} = \sum_{i=1}^{r} (-1)^i \begin{pmatrix} r \\ i \end{pmatrix} y_{n+1,r+1-i} + \bar{y}_{n,r}, \tag{3.46}$$

and therefore the local error estimate is

$$l_n \approx \frac{e_r}{r!} \sum_{i=1}^{r} (-1)^i \begin{pmatrix} r \\ i \end{pmatrix} y_{n+1,r+1-i} + \bar{y}_{n,r}. \tag{3.47}$$

Unfortunately this estimate did not work as well in practice as we expected, because of problems with the underlying difference estimate. For example, in testing a fixed-stepsize version of the order-2 DIMSEM on the problem $u' = \lambda u$, we observed that

$$\frac{D_n^{(r)}}{h^r u^{(r)}(x_{n+1})} \longrightarrow \frac{1}{2} \text{ as } h \to 0, \tag{3.48}$$

while from (3.46) we would expect it to converge to 1. If the exact values of u are substituted for the $y_{n,i}$'s and $\bar{y}_{n,r}$ in (3.46), then the left side of (3.48) does converges to 1 as expected. This still holds if exact values perturbed by h^p, simulating the global error, are used. One attempt at an explanation is to look at the effect of the different global error coefficients for each $y_{n,i}$, $i = 1, \ldots, r$. Consider the global error expansion

$$y_{n,i} = y(x_n + (i - r)h) + g_i h^p y^{(p)}(x_n + (i - r)h) + O(h^{p+1}), \quad i = 1, \ldots, r,$$

where the g_i's are the principal global error coefficients, and $p = r - 1$ is the order of the

method. Suppose $r = 2$, for example. Then we use as an approximation to $y'(x_n)$ the first-order difference

$$\begin{aligned}
\frac{y_{n,2} - y_{n,1}}{h} &= \frac{[y(x_n) + g_1 h y'(x_n) + O(h^2)] - [y(x_n - h) + g_2 h y'(x_n - h) + O(h^2)]}{h} \\
&= \frac{y(x_n) - y(x_n - h)}{h} + \frac{g_1 h y'(x_n) - g_1 h y'(x_n - h)}{h} + \frac{(g_1 - g_2) h y'(x_n - h)}{h} \\
&\quad + O(h) \\
&= [y'(x_n) + O(h)] + \left[g_1 h y''(x_n) + O(h^2)\right] + (g_1 - g_2) y'(x_n - h) + O(h) \\
&= \begin{cases} y'(x_n) + O(h), & \text{if } g_1 = g_2 \\ y'(x_n) + O(1), & \text{if } g_1 \neq g_2. \end{cases}
\end{aligned}$$

If $g_1 \neq g_2$, which is typically the case with DIMSEMs, the difference approximation to $y'(x_n)$ is not even asymptotically correct. Nevertheless, this analysis does not explain why the modified difference scheme, which is described next, does appear to work.

In the successful replacement for (3.46), we compute two difference estimates, each using only data from the same step, that is,

$$h_n^{r-1} u^{(r-1)}(x_{n+1}) \approx D_n^{(r-1)} = \sum_{i=1}^{r} (-1)^i \begin{pmatrix} r-1 \\ i \end{pmatrix} y_{n+1,r+1-i}, \tag{3.49}$$

$$h_n^{r-1} u^{(r-1)}(x_n) \approx D_{n-1}^{(r-1)} = \sum_{i=1}^{r} (-1)^i \begin{pmatrix} r-1 \\ i \end{pmatrix} \bar{y}_{n,r+1-i}. \tag{3.50}$$

Then we subtract them to give

$$h_n^r u^{(r)}(x_{n+1}) \approx D_n^{(r)} = \sum_{i=1}^{r} (-1)^i \begin{pmatrix} r-1 \\ i \end{pmatrix} (y_{n+1,r+1-i} - \bar{y}_{n,r+1-i}), \tag{3.51}$$

and therefore the local error estimate

$$l_n \approx \frac{e_r}{r!} \sum_{i=1}^{r} (-1)^i \begin{pmatrix} r-1 \\ i \end{pmatrix} (y_{n+1,r+1-i} - \bar{y}_{n,r+1-i}). \tag{3.52}$$

(It is worth noting that, when an attempt was made to do local extrapolation with this estimate on a fixed-stepsize DIMSEM (that is, replace y_{n+1} with $y_{n+1} - l_n$ at each step), the order of the method did not increase as would be expected with a one-step method.)

However, when the test in (3.48) is applied to (3.52), the expected result (convergence to 1) is obtained. For this reason, (3.47) was discarded and (3.52) was retained for comparison

to other candidate error estimators, to be described in the next section, for use in our variable-stepsize code.

The local error estimator (3.52) can be extended to provide estimates for a hypothetical step taken by DIMSEMs one order higher and one order lower than the current method. Although a variable-order implementation based on these estimates appeared viable, preliminary testing exposed reliability problems with the higher-order estimate, especially for the higher-order methods (Section 3.3.5). For this reason it was decided to postpone a variable-order implementation until more attention can be given to local error estimation.

3.3.3 Reference Solution-Type Local Error Estimates

Another class of error estimates uses a local reference solution to which the external stages are compared after each step is taken. If the reference solution is locally one order higher than the method, then the difference provides an estimate of the local discretisation error in the method. Since our DIMSEMs are locally order r, we require a reference solution of order $r+1$. This may be obtained from a polynomial interpolant p_n of degree r based on the $r+1$ data values $\bar{y}_{n,i}$, $i=1,\ldots,r$, and $f(Y_{n,i})$ for any i, since all the c_i's are zero. That is,

$$p_n(x_n + d_i h_n) = \bar{y}_{n,i}, \quad , i = 1,\ldots,r,$$

and

$$p_n'(x_n) = f(Y_{n,1}),$$

say. Then the estimate $\mathit{lest}_n \approx l_n$ is given by

$$\begin{aligned}
\mathit{lest}_{n,i} &= y_{n+1,i} - p_n(x_{n+1} + d_i h_n) \\
&= y_{n+1,i} - p_n(x_n + (d_i+1)h_n) \\
&= \begin{cases} y_{n+1,i} - p_n(x_n + d_{i+1}h_n), & i = 1,\ldots,r-1 \\ y_{n+1,i} - p_n(x_{n+1}), & i = r \end{cases} \\
&= \begin{cases} y_{n+1,i} - \bar{y}_{n,i+1}, & i = 1,\ldots,r-1 \\ y_{n+1,i} - p_n(x_{n+1}), & i = r. \end{cases}
\end{aligned} \tag{3.53}$$

This estimate we refer to as the interpolatory estimate. Since it uses an order $r+1$ interpolant based on $\bar{y}_n$ we modified (3.4) to use an order $r+1$ interpolant also, by making it additionally depend on $f(Y_{n-1,1})$. An alternative solution would be to base the interpolant p_n in (3.53) directly on y_n instead of $\bar{y}_n$.

Since p_n is only used for the r-th component of *lest*$_n$, and the first $r-1$ components depend only on the external stages themselves, a modified version of (3.53) would be to compute only

$$\textit{lest}_{n,i}, \quad i = 1, \ldots, r-1, \tag{3.54}$$

and avoid forming p_n altogether. Despite the lack of rigorous justification behind this idea, we felt it is at least worth testing due to its simplicity. We refer to this as the quick estimate.

3.3.4 Essential Local Discretisation Error

The local discretisation error vector l_n has r components, one for each external stage, as does *lest*$_n$. Each component has the same dimension as the problem. We need to reduce the r-vector to a single value in order to control the stepsize. One way to do this is simply to take the maximum norm. Another way is to compute the essential local discretisation error [15, 20, 29] which is defined as

$$\textit{ess}_n = El_n, \tag{3.55}$$

where

$$\begin{aligned} E &= \lim_{k\to\infty} A_2^k \\ &= S^{-1}DS, \end{aligned}$$

$$D = \begin{bmatrix} 1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{bmatrix},$$

and $J = SA_2S^{-1}$ is the Jordan canonical form of A_2. Thus *ess*$_n$ is the projection of l_n onto the component of A_2 corresponding to the eigenvalue 1, and turns out to be the part of the local discretisation error which is propagated in the global error [29]. Since all the rows of E are identical for DIMSEMs, all the r components of *ess*$_n$ are identical also, and it is only necessary to take one for use by the stepsize controller. (We didn't prove this in general, but it does hold for all the DIMSEMs derived in this thesis.)

3.3.5 Testing and Selection of Local Error Estimators

To decide which of the three error estimates described in the previous section to use in our variable-stepsize implementation, fixed-stepsize testing was performed. Each of the estimates was compared on each step to two types of local error. The first is the local discretisation error, l_n, defined in (3.36) in accordance with Skeel [29]. This error compares the true values $\hat{y}_{n+1}$, derived from the exact solution of the IVP, to the y_{n+1} produced by the method in a reference step starting from $\hat{y}_n$. This was used in the testing because it is the quantity related by Skeel's results to the global error, and so would be expected, if controlled to a given tolerance, to exercise proportional control upon the global error. However, it is unreasonable to expect a local error estimate based on data obtained during an integration to track l_n with any accuracy, as the computed solution may drift from the true solution. The second type of local error is the numerical local error lnum_n. This is defined in a manner similar to l_n, but instead of using the exact solution as the basis for starting the reference step, the solution to the local problem passing through $y_{n,r}$ is used. Comparison to lnum_n is a more reasonable test of a local error estimator. Note that neither the local discretisation error nor the numerical local error are computable in a practical integration, as, first, they require knowledge of the analytical solution of the problem, and second, they require taking each step twice, doubling the cost of the method. For this reason, the testing was performed using the problem $y' = \lambda y$, $y(0) = 1$, for which the exact solution as well as the local solution are known. The problem was integrated from 0 to 10 with $h = 0.1$ and several values for λ. The results in Table 3.2 give the average numerical local error,

$$\frac{1}{N} \sum_{n=0}^{N-1} \| \mathit{lnum}_n \|,$$

as well as the average relative discrepancy values

$$\frac{1}{N} \sum_{n=0}^{N-1} \frac{\| \mathit{lnum}_n - \mathit{lest}_n \|}{\| \mathit{lnum}_n \|},$$

where lest_n is defined as in (3.52), (3.53), and (3.54), for the columns labeled difference, interpolatory and quick, respectively. The lowest discrepancies for each run are shown in bold, except where the average numerical local error is near roundoff level, for then the relative discrepancies are not reliable. (The discrepancies decrease with λ until the error reaches around 10^{-15}, at which point they start to increase.) The greatest number of lowest discrepancies is 21 for the difference estimator, followed by 2 for the interpolatory and 1 for

the quick estimator. Therefore the difference estimator was chosen for use in subsequent variable-stepsize testing.

3.4 Preliminary Testing of Variable-Stepsize Method

Having settled on a local error estimator, we next performed some variable-stepsize testing to determine the details of the stepsize control scheme. The stepsize was adjusted in response to the local error estimate using what is known as the locally optimal controller. This involves first computing the locally optimal stepsize for the step just taken,

$$h_n^* = \left(\frac{tol}{lest_n} \right)^{\frac{1}{p+1}},$$

and then obtaining the stepsize for the next step from

$$h_{n+1} = \mu h_n^*,$$

where $\mu = 0.9$ is a safety factor intended to reduce the number of failed steps.

To determine the effect of varying the stepsize on the reliability of the local error estimate, $lest_n$ (3.52) was monitored while

1. keeping the stepsize fixed,
2. varying the stepsize every step, but restricting ω by the limits shown in Table 3.1,
3. varying the stepsize every r steps, after $r-1$ constant steps.

In cases 2 and 3, the numerical local error $lnum_n$ was used instead of an estimate to control the stepsize, to avoid introducing the further complication of possible stepsize anomalies due to questionable estimators. The test problems $u' = \lambda u$, $\lambda = 1, -1, -10, -1000$, were used. Examples of the results are shown in Tables 3.3-3.4, 3.5-3.6 and 3.7-3.8, which represent cases 1, 2 and 3 respectively for DIMSEM4 and $\lambda = -1$.

The values shown in columns 3 to 7 are defined by

$$R_n^{(k)} = \frac{D_n^{(k)}}{h_n^k u^{(k)}(x_n)}, \quad k = r-1, r, r+1,$$

which should all approach 1 for small h_n. Both the fixed- and variable-stepsize values for $R_{n+1}^{(r-1)}$ and $R_n^{(r-1)}$ do this very well. The fixed-stepsize values (Case 1) for $R_{n+1}^{(r)}$ and $R_n^{(r)}$

order	λ	avg $\|\|lnum_n\|\|$	avg $\|\|lnum_n - lest_n\|\| / \|\|lnum_n\|\|$ difference	interpolatory	quick
2	–1.0E+02	3.9E+15	1.0E+00	1.0E+00	1.0E+00
2	–1.0E+01	2.2E–02	4.8E–01	**2.3E–01**	5.9E–01
2	–1.0E+00	3.9E–05	**1.4E–02**	4.9E–01	2.0E+00
2	–1.0E–01	2.1E–07	**8.6E–03**	4.8E–01	2.0E+00
2	–1.0E–02	3.2E–10	**8.4E–03**	4.8E–01	2.0E+00
2	–1.0E–04	3.0E–16	3.6E–01	9.4E–01	2.4E+00
2	1.0E–04	2.6E–16	5.1E–01	8.1E–01	2.9E+00
2	1.0E–01	5.7E–07	**8.4E–03**	4.8E–01	2.0E+00
2	1.0E+00	6.4E–01	**1.1E–02**	4.8E–01	2.0E+00
3	–1.0E+02	1.2E+24	1.0E+00	1.0E+00	1.0E+00
3	–1.0E+01	4.3E–02	3.1E–02	**2.2E–02**	4.5E+00
3	–1.0E+00	5.2E–06	**4.5E–02**	1.8E–01	3.3E+00
3	–1.0E–01	2.8E–09	**1.5E–02**	2.4E–01	3.0E+00
3	–1.0E–02	4.2E–13	**1.2E–02**	2.4E–01	3.0E+00
3	–1.0E–04	1.7E–16	7.1E–01	3.8E–01	3.5E+00
3	1.0E–04	2.0E–16	4.9E–01	2.8E–01	2.1E+00
3	1.0E–01	7.4E–09	**1.5E–02**	2.5E–01	3.0E+00
3	1.0E+00	8.2E–02	**4.5E–02**	3.1E–01	2.7E+00
4	–1.0E+02	6.0E+32	1.0E+00	1.0E+00	1.0E+00
4	–1.0E+01	1.7E–01	7.2E–01	7.4E–01	7.2E–01
4	–1.0E+00	5.8E–07	**3.2E–02**	1.3E–01	3.3E+00
4	–1.0E–01	3.0E–11	**1.7E–02**	1.2E–01	3.9E+00
4	–1.0E–02	4.0E–16	1.4E+00	1.1E+00	7.7E+00
4	–1.0E–04	2.9E–16	1.6E+00	1.3E+00	1.0E+01
4	1.0E–04	2.8E–16	2.0E+00	1.8E+00	1.2E+01
4	1.0E–01	7.7E–11	**1.6E–02**	1.1E–01	4.1E+00
4	1.0E+00	7.9E–03	**1.6E–02**	5.3E–02	4.9E+00
5	–1.0E+02	2.5E+41	1.0E+00	1.0E+00	1.0E+00
5	–1.0E+01	5.5E–01	**7.5E–01**	7.9E–01	7.9E–01
5	–1.0E+00	6.6E–08	**5.2E–02**	3.2E–01	5.2E+00
5	–1.0E–01	3.3E–13	**2.6E–02**	3.5E–01	5.0E+00
5	–1.0E–02	7.1E–16	2.8E+00	1.7E+00	1.8E+01
5	–1.0E–04	7.8E–16	2.1E+00	1.3E+00	1.6E+01
5	1.0E–04	1.2E–15	1.7E+00	1.0E+00	1.2E+01
5	1.0E–01	8.4E–13	**2.4E–02**	3.5E–01	5.0E+00
5	1.0E+00	8.5E–04	**4.4E–02**	3.8E–01	4.7E+00
6	–1.0E+02	1.4E+50	1.0E+00	1.0E+00	1.0E+00
6	–1.0E+01	2.2E+00	8.8E–01	9.1E–01	**3.8E–01**
6	–1.0E+00	7.3E–09	**6.4E–02**	3.8E–01	5.8E+00
6	–1.0E–01	2.7E–15	1.4E+00	8.1E–01	1.4E+01
6	–1.0E–02	1.0E–15	6.6E+00	3.9E+00	5.4E+01
6	–1.0E–04	1.0E–15	8.5E+00	4.8E+00	6.5E+01
6	1.0E–04	1.4E–15	6.3E+00	3.6E+00	4.9E+01
6	1.0E–01	8.4E–15	1.1E+00	7.0E–01	1.1E+01
6	1.0E+00	8.5E–05	**5.2E–02**	4.4E–01	6.2E+00

Table 3.2: Comparison of Local Error Estimate to Numerical Local Error for DIMSEMs on $u'(x) = \lambda u(x)$, $h = 0.1$, $x \in [0, 10]$. (Best in bold.)

behave well until roundoff error becomes a problem starting at $h_n = 10^{-3}$. The fixed-stepsize values for $R_{n+1}^{(r+1)}$ also behave well down to $h_n = 5.62 \times 10^{-3}$. The variable-stepsize values (Case 2) for $R_{n+1}^{(r)}$, $R_n^{(r)}$ and $R_{n+1}^{(r+1)}$ are erratic throughout the stepsize range. In Case 3, $R_{n+1}^{(r+1)}$ is still erratic, although $R_{n+1}^{(r)}$ and $R_n^{(r)}$ behave much better.

It is because of the poor behaviour of $R_{n+1}^{(r+1)}$ in the variable-stepsize tests that we chose not to attempt a variable-order implementation, as the estimate of the error in the next-higher-order method is likely to be unsatisfactory. The best estimate for the current-order method was achieved in Case 3, in which the stepsize was changed at most every r steps. Therefore this approach was selected for our final implementation.

Even with the adoption of this strategy, we encountered situations where the stepsize controller became unstable with DIMSEM6. A typical sequence was that a retry of a failed step would have a large estimate. In response, the stepsize was reduced substantially. This caused the estimate to become tiny, as a result of which the stepsize was increased drastically, higher than the original value. Inspection of the external stage values did not suggest that the method itself was becoming unstable. Rather the error estimate appeared sometimes to become unreliable after a failed step, as the initial large estimate did not correspond to a significant increase in the numerical local error.

Two actions were taken to prevent this behaviour. First, the usual technique of limiting the stepsize-change ratio to some heuristic maximum and minimum was applied (see Section 3.6). This is a good safeguard against a variety of problems with error estimates and stepsize controllers. Second, the method was made to strictly satisfy the requirement that at least $r - 1$ steps be taken with the same stepsize before a change is allowed, even on failed steps. (That is, a failed step causes the method to back up to the most recent point where a stepsize change is allowed.) Although this is expensive, it improves the reliability of the error estimate, and eliminates the possibility of stepsize change-induced instability in the method.

Finally, the technique of controlling the essential estimate $E\,\mathit{lest}_n$ (3.55) was tested, but since no significant improvement in behaviour was observed over using $||\mathit{lest}_n||_\infty$, the max-norm was used instead.

3.5 Sequential Variable-Stepsize Testing

The variable-stepsize DIMSEM implementation was initially tested on a sequential machine to verify that it was working correctly and to determine whether there was any potential for

x_n	h_n	$R_{n+1}^{(r-1)}$	$R_n^{(r-1)}$	$R_{n+1}^{(r)}$	$R_n^{(r)}$	$R_{n+1}^{(r+1)}$
1.00E+00	1.00E+00	7.92E+00	8.63E+00	7.94E-01	-3.70E+18	-3.74E+18
5.62E-01	5.62E-01	3.01E+00	3.21E+00	4.12E-01	1.53E+09	2.76E+09
1.12E+00	5.62E-01	1.89E+00	3.01E+00	6.04E+00	5.77E+07	1.80E+08
1.69E+00	5.62E-01	2.81E+00	1.89E+00	9.13E-01	6.04E+00	1.72E+01
3.16E-01	3.16E-01	1.82E+00	1.89E+00	2.95E-01	-5.99E+00	-2.01E+01
6.32E-01	3.16E-01	1.49E+00	1.82E+00	3.20E+00	6.05E+06	2.63E+07
9.49E-01	3.16E-01	1.70E+00	1.49E+00	1.06E+00	3.20E+00	1.05E+01
1.26E+00	3.16E-01	1.64E+00	1.70E+00	2.20E+00	1.06E+00	-2.36E+00
5.33E-01	1.78E-01	1.33E+00	1.26E+00	9.50E-01	2.27E+00	9.90E+00
7.11E-01	1.78E-01	1.31E+00	1.33E+00	1.57E+00	9.50E-01	-2.43E+00
8.89E-01	1.78E-01	1.31E+00	1.31E+00	1.44E+00	1.57E+00	2.45E+00
1.07E+00	1.78E-01	1.31E+00	1.31E+00	1.44E+00	1.44E+00	1.57E+00
7.00E-01	1.00E-01	1.16E+00	1.16E+00	1.22E+00	1.22E+00	1.29E+00
8.00E-01	1.00E-01	1.16E+00	1.16E+00	1.22E+00	1.22E+00	1.29E+00
9.00E-01	1.00E-01	1.16E+00	1.16E+00	1.22E+00	1.22E+00	1.29E+00
1.00E+00	1.00E-01	1.16E+00	1.16E+00	1.22E+00	1.22E+00	1.29E+00
7.31E-01	5.62E-02	1.09E+00	1.09E+00	1.12E+00	1.12E+00	1.15E+00
7.87E-01	5.62E-02	1.09E+00	1.09E+00	1.12E+00	1.12E+00	1.15E+00
8.44E-01	5.62E-02	1.09E+00	1.09E+00	1.12E+00	1.12E+00	1.15E+00
9.00E-01	5.62E-02	1.09E+00	1.09E+00	1.12E+00	1.12E+00	1.15E+00
7.59E-01	3.16E-02	1.05E+00	1.05E+00	1.07E+00	1.07E+00	1.08E+00
7.91E-01	3.16E-02	1.05E+00	1.05E+00	1.07E+00	1.07E+00	1.08E+00
8.22E-01	3.16E-02	1.05E+00	1.05E+00	1.07E+00	1.07E+00	1.08E+00
8.54E-01	3.16E-02	1.05E+00	1.05E+00	1.07E+00	1.07E+00	1.08E+00
7.65E-01	1.78E-02	1.03E+00	1.03E+00	1.04E+00	1.04E+00	1.05E+00
7.82E-01	1.78E-02	1.03E+00	1.03E+00	1.04E+00	1.04E+00	1.05E+00
8.00E-01	1.78E-02	1.03E+00	1.03E+00	1.04E+00	1.04E+00	1.05E+00
8.18E-01	1.78E-02	1.03E+00	1.03E+00	1.04E+00	1.04E+00	1.05E+00

Table 3.3: Test of Difference Estimators $D_n^{(k)}$. $R_n^{(k)} = D_n^{(k)}/(h_n^k u^{(k)}(x_n))$. Case 1: Fixed-stepsize.

x_n	h_n	$R_{n+1}^{(r-1)}$	$R_n^{(r-1)}$	$R_{n+1}^{(r)}$	$R_n^{(r)}$	$R_{n+1}^{(r+1)}$
7.80E-01	1.00E-02	1.02E+00	1.02E+00	1.02E+00	1.02E+00	1.03E+00
7.90E-01	1.00E-02	1.02E+00	1.02E+00	1.02E+00	1.02E+00	1.02E+00
8.00E-01	1.00E-02	1.02E+00	1.02E+00	1.02E+00	1.02E+00	1.02E+00
8.10E-01	1.00E-02	1.02E+00	1.02E+00	1.02E+00	1.02E+00	1.03E+00
7.93E-01	5.62E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	1.04E+00
7.99E-01	5.62E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	1.05E+00
8.04E-01	5.62E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	9.56E-01
8.10E-01	5.62E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	1.09E+00
7.94E-01	3.16E-03	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.47E+00
7.97E-01	3.16E-03	1.00E+00	1.00E+00	1.01E+00	1.00E+00	-2.70E+00
8.00E-01	3.16E-03	1.00E+00	1.00E+00	1.00E+00	1.01E+00	4.80E+00
8.03E-01	3.16E-03	1.00E+00	1.00E+00	1.00E+00	1.00E+00	8.65E-01
7.97E-01	1.78E-03	1.00E+00	1.00E+00	1.04E+00	1.09E+00	3.11E+01
7.98E-01	1.78E-03	1.00E+00	1.00E+00	1.00E+00	1.04E+00	1.95E+01
8.00E-01	1.78E-03	1.00E+00	1.00E+00	9.23E-01	1.00E+00	4.68E+01
8.02E-01	1.78E-03	1.00E+00	1.00E+00	8.83E-01	9.23E-01	2.34E+01
7.98E-01	1.00E-03	1.00E+00	1.00E+00	1.23E+00	-1.72E+00	-2.96E+03
7.99E-01	1.00E-03	1.00E+00	1.00E+00	3.82E+00	1.23E+00	-2.59E+03
8.00E-01	1.00E-03	1.00E+00	1.00E+00	-1.23E-01	3.82E+00	3.95E+03
8.01E-01	1.00E-03	1.00E+00	1.00E+00	1.73E+00	-1.23E-01	-1.85E+03
7.99E-01	5.62E-04	9.96E-01	1.00E+00	8.77E+00	-1.10E+01	-3.51E+04
8.00E-01	5.62E-04	1.00E+00	9.96E-01	-1.32E+01	8.77E+00	3.90E+04
8.00E-01	5.62E-04	1.04E+00	1.00E+00	-4.83E+01	-1.32E+01	6.25E+04
8.01E-01	5.62E-04	1.01E+00	1.04E+00	3.96E+01	-4.83E+01	-1.56E+05
7.99E-01	3.16E-04	8.52E-01	8.76E-01	3.12E+02	1.09E+03	2.47E+06
8.00E-01	3.16E-04	9.26E-01	8.52E-01	-4.29E+02	3.12E+02	2.35E+06
8.00E-01	3.16E-04	1.05E+00	9.26E-01	-2.34E+02	-4.29E+02	-6.18E+05
8.00E-01	3.16E-04	1.25E+00	1.05E+00	-5.08E+02	-2.34E+02	8.65E+05

Table 3.4: Test of Difference Estimators $D_n^{(k)}$. $R_n^{(k)} = D_n^{(k)}/(h_n^k u^{(k)}(x_n))$. Case 1: Fixed-stepsize. (Cont'd.)

x_n	h_n	$R_{n+1}^{(r-1)}$	$R_n^{(r-1)}$	$R_{n+1}^{(r)}$	$R_n^{(r)}$	$R_{n+1}^{(r+1)}$
6.41E-01	1.32E-01	1.24E+00	1.19E+00	6.21E-01	4.79E-01	-6.64E-01
8.05E-01	1.64E-01	1.31E+00	1.24E+00	6.56E-01	5.01E-01	-5.15E-01
1.00E+00	1.95E-01	1.38E+00	1.31E+00	8.31E-01	5.51E-01	-9.31E-01
6.41E-01	1.32E-01	1.24E+00	1.19E+00	6.21E-01	4.79E-01	-6.64E-01
8.05E-01	1.64E-01	1.31E+00	1.24E+00	6.56E-01	5.01E-01	-5.15E-01
1.00E+00	1.95E-01	1.38E+00	1.31E+00	8.31E-01	5.51E-01	-9.31E-01
6.41E-01	1.32E-01	1.24E+00	1.19E+00	6.21E-01	4.79E-01	-6.64E-01
8.05E-01	1.64E-01	1.31E+00	1.24E+00	6.56E-01	5.01E-01	-5.15E-01
1.00E+00	1.95E-01	1.38E+00	1.31E+00	8.31E-01	5.51E-01	-9.31E-01
6.41E-01	1.32E-01	1.24E+00	1.19E+00	6.21E-01	4.79E-01	-6.64E-01
8.05E-01	1.64E-01	1.31E+00	1.24E+00	6.56E-01	5.01E-01	-5.15E-01
1.00E+00	1.95E-01	1.38E+00	1.31E+00	8.31E-01	5.51E-01	-9.31E-01
6.41E-01	1.32E-01	1.24E+00	1.19E+00	6.21E-01	4.79E-01	-6.64E-01
8.05E-01	1.64E-01	1.31E+00	1.24E+00	6.56E-01	5.01E-01	-5.15E-01
1.00E+00	1.95E-01	1.38E+00	1.31E+00	8.31E-01	5.51E-01	-9.31E-01
6.35E-01	1.26E-01	1.23E+00	1.19E+00	7.42E-01	5.01E-01	-1.46E+00
7.31E-01	9.57E-02	9.34E-01	1.23E+00	4.83E+00	9.81E-01	-3.88E+01
8.18E-01	8.72E-02	1.45E+00	9.34E-01	-4.89E+00	5.30E+00	1.23E+02
9.07E-01	8.86E-02	1.01E+00	1.45E+00	6.53E+00	-4.81E+00	-1.33E+02
7.47E-01	7.72E-02	1.14E+00	1.12E+00	6.54E-01	6.01E-01	-1.92E-01
8.29E-01	8.26E-02	1.14E+00	1.14E+00	1.11E+00	6.11E-01	-5.44E+00
9.05E-01	7.55E-02	1.10E+00	1.14E+00	1.83E+00	1.21E+00	-6.74E+00
7.37E-01	5.19E-02	1.09E+00	1.07E+00	5.33E-01	3.98E-01	-2.28E+00
7.80E-01	4.23E-02	1.02E+00	1.09E+00	3.12E+00	6.55E-01	-5.74E+01
8.18E-01	3.87E-02	1.12E+00	1.02E+00	-1.44E+00	3.41E+00	1.29E+02
8.64E-01	4.55E-02	1.08E+00	1.12E+00	1.68E+00	-1.22E+00	-6.49E+01

Table 3.5: Test of Difference Estimators $D_n^{(k)}$. $R_n^{(k)} = D_n^{(k)}/(h_n^k u^{(k)}(x_n))$. Case 2: Variable-stepsize, $\omega \leq \omega^*$.

x_n	h_n	$R_{n+1}^{(r-1)}$	$R_n^{(r-1)}$	$R_{n+1}^{(r)}$	$R_n^{(r)}$	$R_{n+1}^{(r+1)}$
7.52E-01	2.68E-02	1.04E+00	1.03E+00	7.32E-01	2.01E+00	4.97E+01
7.85E-01	3.25E-02	1.06E+00	1.04E+00	5.47E-01	6.04E-01	2.26E+00
8.15E-01	3.08E-02	1.04E+00	1.06E+00	1.62E+00	5.77E-01	-3.34E+01
8.44E-01	2.80E-02	1.04E+00	1.04E+00	1.00E+00	1.78E+00	2.98E+01
7.79E-01	1.65E-02	1.03E+00	1.03E+00	8.40E-01	-1.28E+00	-1.30E+02
8.00E-01	2.01E-02	1.03E+00	1.03E+00	6.71E-01	6.88E-01	1.39E+00
8.21E-01	2.11E-02	1.03E+00	1.03E+00	1.06E+00	6.40E-01	-1.95E+01
7.80E-01	1.23E-02	1.02E+00	1.02E+00	9.57E-01	9.43E-01	-2.28E-01
7.92E-01	1.24E-02	1.02E+00	1.02E+00	1.06E+00	9.53E-01	-7.44E+00
8.04E-01	1.22E-02	1.02E+00	1.02E+00	1.08E+00	1.08E+00	5.82E-01
8.17E-01	1.21E-02	1.02E+00	1.02E+00	9.87E-01	1.08E+00	9.09E+00
7.89E-01	7.71E-03	1.01E+00	1.01E+00	9.87E-01	1.02E+00	5.43E+00
7.96E-01	7.78E-03	1.01E+00	1.01E+00	9.96E-01	9.78E-01	-1.30E+00
8.04E-01	7.77E-03	1.01E+00	1.01E+00	1.03E+00	9.96E-01	-3.61E+00
8.12E-01	7.73E-03	1.01E+00	1.01E+00	1.03E+00	1.04E+00	2.33E+00
7.93E-01	4.89E-03	1.01E+00	1.01E+00	1.01E+00	1.00E+00	-4.36E-01
7.98E-01	4.89E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	1.02E+00
8.03E-01	4.90E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	1.66E+00
8.08E-01	4.90E-03	1.01E+00	1.01E+00	1.00E+00	1.00E+00	1.02E+00
7.97E-01	2.63E-03	1.01E+00	1.00E+00	-7.84E-02	2.49E+00	9.81E+02
8.00E-01	3.26E-03	1.01E+00	1.01E+00	7.69E-01	-6.42E-02	-2.56E+02
8.03E-01	2.91E-03	1.00E+00	1.01E+00	2.18E+00	8.51E-01	-4.55E+02
7.99E-01	5.48E-04	1.02E+00	1.02E+00	-1.49E+01	-4.13E+02	-7.26E+05
8.00E-01	6.80E-04	1.03E+00	1.03E+00	2.55E+00	-3.48E+01	-5.50E+04
8.01E-01	8.43E-04	1.02E+00	1.03E+00	1.80E+01	-4.06E+01	-6.95E+04
8.02E-01	1.05E-03	1.01E+00	1.02E+00	1.21E+01	9.30E+00	-2.65E+03

Table 3.6: Test of Difference Estimators $D_n^{(k)}$. $R_n^{(k)} = D_n^{(k)}/(h_n^k u^{(k)}(x_n))$. Case 2: Variable-stepsize, $\omega \leq \omega^*$. (Cont'd.)

x_n	h_n	$R_{n+1}^{(r-1)}$	$R_n^{(r-1)}$	$R_{n+1}^{(r)}$	$R_n^{(r)}$	$R_{n+1}^{(r+1)}$
1.00E+00	9.60E-01	1.03E+00	1.02E+00	1.34E-07	1.20E-02	1.27E-02
6.89E-01	6.49E-01	1.03E+00	1.02E+00	-6.83E-07	1.78E-02	2.77E-02
1.00E+00	3.11E-01	1.48E+00	1.03E+00	1.54E+00	-3.33E-06	-4.93E+00
4.49E-01	4.09E-01	1.03E+00	1.02E+00	-5.35E-08	2.82E-02	6.97E-02
8.58E-01	4.09E-01	1.52E+00	1.03E+00	6.69E-02	-5.35E-08	-1.63E-01
1.00E+00	1.42E-01	-8.91E+00	1.52E+00	7.91E+01	1.93E-01	-5.57E+02
2.98E-01	2.58E-01	1.03E+00	1.02E+00	3.63E-06	4.47E-02	1.75E-01
5.56E-01	2.58E-01	1.31E+00	1.03E+00	7.03E-02	3.63E-06	-2.72E-01
8.15E-01	2.58E-01	1.48E+00	1.31E+00	8.34E-01	7.03E-02	-2.88E+00
1.00E+00	1.85E-01	7.36E-01	1.48E+00	6.36E+00	1.16E+00	-2.62E+01
5.29E-01	1.63E-01	1.28E+00	1.19E+00	7.74E-01	8.15E-02	-4.16E+00
6.92E-01	1.63E-01	1.28E+00	1.28E+00	1.36E+00	7.74E-01	-2.73E+00
8.66E-01	1.74E-01	1.33E+00	1.28E+00	1.06E+00	1.27E+00	2.49E+00
1.00E+00	1.34E-01	9.38E-01	1.33E+00	4.79E+00	1.38E+00	-2.35E+01
6.68E-01	1.08E-01	1.17E+00	1.19E+00	1.39E+00	9.89E-01	-2.67E+00
7.77E-01	1.08E-01	1.18E+00	1.17E+00	1.17E+00	1.39E+00	3.54E+00
8.85E-01	1.08E-01	1.18E+00	1.18E+00	1.27E+00	1.17E+00	3.35E-01
1.00E+00	1.15E-01	1.20E+00	1.18E+00	9.88E-01	1.19E+00	2.99E+00
7.13E-01	7.12E-02	1.11E+00	1.12E+00	1.27E+00	9.41E-01	-3.71E+00
7.84E-01	7.12E-02	1.11E+00	1.11E+00	1.09E+00	1.27E+00	3.86E+00
8.56E-01	7.12E-02	1.11E+00	1.11E+00	1.17E+00	1.09E+00	6.86E-02
9.31E-01	7.52E-02	1.13E+00	1.11E+00	9.26E-01	1.11E+00	3.49E+00
7.31E-01	4.54E-02	1.07E+00	1.07E+00	1.11E+00	1.05E+00	-4.46E-02
7.78E-01	4.71E-02	1.08E+00	1.07E+00	9.51E-01	1.07E+00	3.52E+00
8.25E-01	4.71E-02	1.07E+00	1.08E+00	1.18E+00	9.51E-01	-3.98E+00
8.72E-01	4.71E-02	1.07E+00	1.07E+00	1.06E+00	1.18E+00	3.92E+00

Table 3.7: Test of Difference Estimators $D_n^{(k)}$. $R_n^{(k)} = D_n^{(k)}/(h_n^k u^{(k)}(x_n))$. Case 3: Variable-stepsize, $r-1$ constant steps before change.

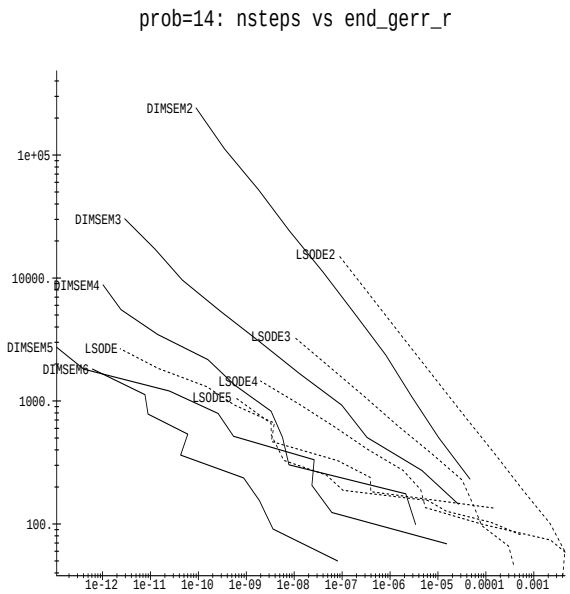
x_n	h_n	$R_{n+1}^{(r-1)}$	$R_n^{(r-1)}$	$R_{n+1}^{(r)}$	$R_n^{(r)}$	$R_{n+1}^{(r+1)}$
7.61E-01	2.98E-02	1.05E+00	1.05E+00	1.12E+00	9.67E-01	-4.04E+00
7.91E-01	2.98E-02	1.05E+00	1.05E+00	1.03E+00	1.12E+00	3.90E+00
8.21E-01	2.98E-02	1.05E+00	1.05E+00	1.07E+00	1.03E+00	-6.71E-02
8.52E-01	3.05E-02	1.05E+00	1.05E+00	9.66E-01	1.04E+00	3.54E+00
7.65E-01	1.91E-02	1.03E+00	1.03E+00	9.78E-01	1.03E+00	3.56E+00
7.84E-01	1.91E-02	1.03E+00	1.03E+00	1.07E+00	9.78E-01	-4.07E+00
8.03E-01	1.91E-02	1.03E+00	1.03E+00	1.02E+00	1.07E+00	3.88E+00
8.23E-01	1.91E-02	1.03E+00	1.03E+00	1.04E+00	1.02E+00	-9.07E-02
7.83E-01	1.21E-02	1.02E+00	1.02E+00	1.05E+00	9.86E-01	-4.09E+00
7.95E-01	1.21E-02	1.02E+00	1.02E+00	1.01E+00	1.05E+00	3.88E+00
8.07E-01	1.21E-02	1.02E+00	1.02E+00	1.03E+00	1.01E+00	-1.05E-01
8.20E-01	1.23E-02	1.02E+00	1.02E+00	9.86E-01	1.02E+00	3.57E+00
7.88E-01	7.72E-03	1.01E+00	1.01E+00	9.91E-01	1.01E+00	3.58E+00
7.96E-01	7.72E-03	1.01E+00	1.01E+00	1.03E+00	9.91E-01	-4.13E+00
8.03E-01	7.72E-03	1.01E+00	1.01E+00	1.01E+00	1.03E+00	3.90E+00
8.11E-01	7.72E-03	1.01E+00	1.01E+00	1.02E+00	1.01E+00	-1.37E-01
7.90E-01	4.87E-03	1.01E+00	1.01E+00	1.01E+00	1.01E+00	2.19E-01
7.95E-01	4.89E-03	1.01E+00	1.01E+00	9.93E-01	1.01E+00	3.69E+00
8.00E-01	4.89E-03	1.01E+00	1.01E+00	1.02E+00	9.93E-01	-4.57E+00
8.05E-01	4.89E-03	1.01E+00	1.01E+00	1.01E+00	1.02E+00	3.98E+00
7.94E-01	3.08E-03	1.00E+00	1.00E+00	1.01E+00	1.01E+00	1.15E+00
7.97E-01	3.09E-03	1.00E+00	1.00E+00	9.85E-01	1.00E+00	7.47E+00
8.00E-01	3.09E-03	1.00E+00	1.00E+00	1.01E+00	9.85E-01	-5.66E+00
8.03E-01	3.09E-03	1.00E+00	1.00E+00	1.01E+00	1.01E+00	4.26E-01

Table 3.8: Test of Difference Estimators $D_n^{(k)}$. $R_n^{(k)} = D_n^{(k)}/(h_n^k u^{(k)}(x_n))$. Case 3: Variable-stepsize, $r-1$ constant steps before change. (Cont'd.)

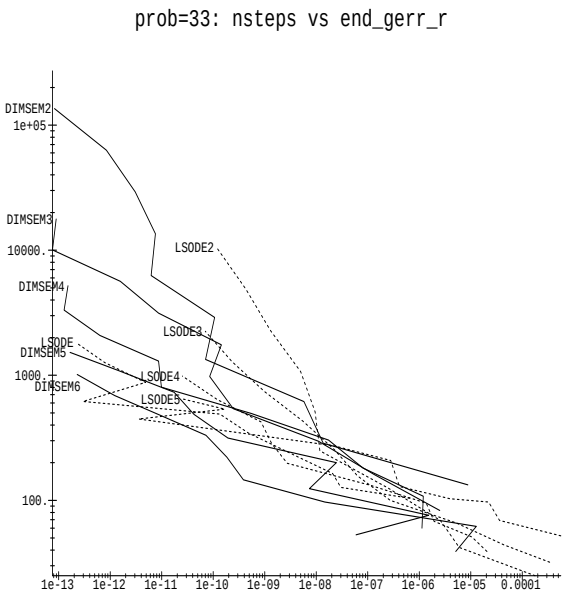
performance gains on a parallel machine. LSODE [25] with the BDF option was chosen as a good-quality integrator for comparison purposes. Since LSODE is a variable-order code and our DIMSEM implementation is fixed order, the LSODE code was modified to accept an order parameter, which prevents order change after the initial startup process reaches the desired order. We refer to the order-p fixed-order modification of LSODE as LSODEp. The scaled stiff DETEST [18] test problem set was used. An efficiency comparison was made on the basis of the number of steps used to achieve a given endpoint global error. (Even though monitoring the maximum global error might be more appropriate for stiff problems, the endpoint global error is easier to compute. DETEST provides precomputed true solutions for the endpoint, whereas to compute maximum global error, a high-accuracy integration of the problem would have to be performed with a reliable method. Despite the use of endpoint global error, limiting accuracy being reached by the end of the integration was not a problem in the testing.) The number of steps taken is a reasonable quantity to compare directly between DIMSEMp and LSODEp, because a BDF formula requires one new function evaluation per nonlinear solver iteration on each step, while a DIMSEM requires s, all of which can be done in parallel. For a DIMSEM to have a chance of being competitive with LSODE, it must therefore take fewer steps for the same accuracy, assuming that the same number of nonlinear solver iterations are taken per step by the two methods. (The DIMSEMs might be expected to require fewer iterations, as $y_{n,r}$ provides a good initial guess for each $Y_{n,i}$, $i = 1, \ldots, s$.) Then, in addition, the parallel speedup must be sufficient and the heuristics and number of iterations per step comparable, points which are considered in the next section.

The problems were run with absolute local error tolerances of 10^{-k}, $k = 4, \ldots, 13$, and efficiency plots of the number of steps versus the endpoint global error $\|y_{N,r} - u(x_N)\|$ were made. Examples of these plots are shown in Fig. 3.1. These plots were used to verify reasonable proportionality behaviour, and generally to detect any abnormalities in the methods. The plots are mainly as expected, except in a few cases where the DIMSEMs of order 5 and 6 continue to take a large number of steps despite the accuracy being relaxed. The reason for this is not known, although it may be due to some deficiency of the local error estimator.

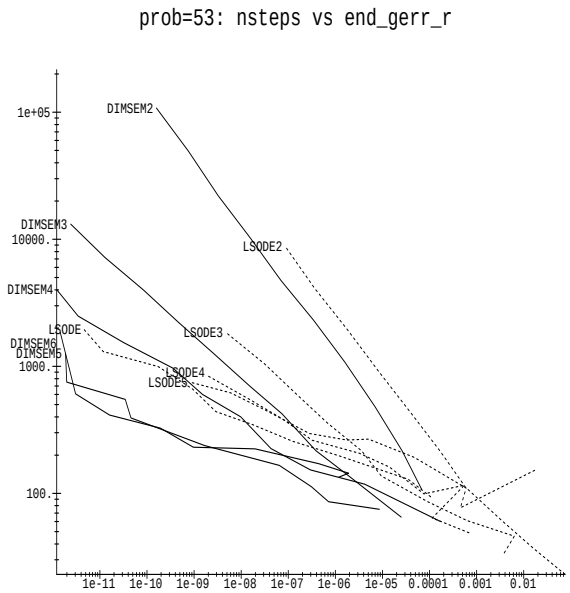
To facilitate efficiency comparisons, the data is presented in Figs. 3.2 to 3.5 in the concise form introduced by Sharp [28]. (Because LSODE provides BDFs up to order 5 only, there is no table for order 6.) The DETEST problem number is shown on the vertical axis (1n = problem An, 2n = problem Bn, etc.), the $\log_{10}$ endpoint global error is shown on the horizontal



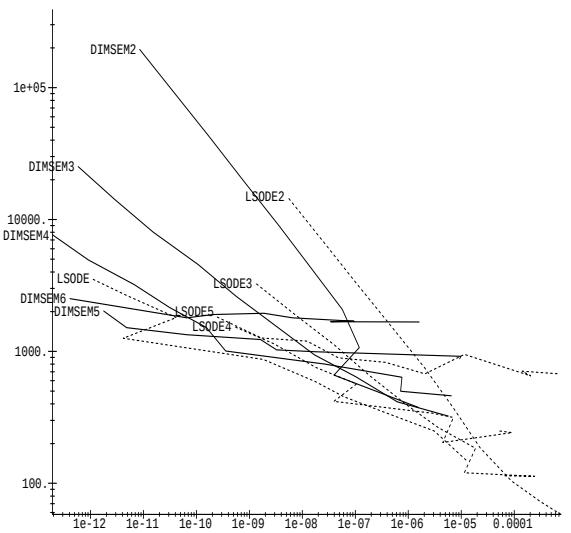
Problem A4



Problem C3



Problem E3



Problem F1

Figure 3.1: Number of Steps vs. Endpoint Global Error for DIMSEM and Fixed-Order LSODE on Selected Scaled Stiff DETEST Problems. (end_gerr_r $= \|y_{n,r} - u(x_n)\|$.)

order	DIMSEM better	LSODE better	% DIMSEM better
2	65	33	66
3	88	28	76
4	93	32	74
5	116	18	87

Table 3.9: Number of Scaled Stiff DETEST Problem Runs for which DIMSEMp Took Fewer Steps Than LSODEp and Vice Versa, for the Same Accuracy

axis, and the numbers in the body of the table are percentage superiority figures defined by

$$\frac{D - L}{\max(D, L)} \times 100,$$

where D and L are the number of steps taken by DIMSEMp and LSODEp for the order p shown at the top of the table. Therefore the numbers range from -100 to +100, and a negative value indicates DIMSEMp has taken fewer steps, the magnitude of the number reflecting the relative superiority of the better-performing method. (The values of D and L for a given error 10^n was determined by linear interpolation from the two test runs producing the nearest errors.) The results are summarised in Table 3.9, which shows the DIMSEMp's to take fewer steps than the corresponding LSODEp's on most of the problems and accuracies. Section 3.7 investigates whether this potential translates into better real-time performance when the DIMSEMs are run in parallel.

3.6 Variable-Stepsize Implementation Details

This section contains details of the final variable-stepsize fixed-order DIMSEM implementation.

The starting values $y_{0,i}$, $i = 1, \ldots, r$, were computed by (variable-order) LSODE to the same tolerance specified for the DIMSEM. $y_{0,1}$ was arranged to coincide with the initial condition, and LSODE integrated successively from $y_{0,i}$ to $y_{0,i+1}$, for $i = 1, \ldots, r-1$. For a sufficiently lengthy integration, the cost of the starting method is small compared to the overall cost, so the effect on performance is small. In a more realistic implementation, a scheme similar to that used by LSODE could be adopted. For example, take $y_{0,1}$ from the initial condition, compute $y_{0,2}$ and $y_{0,3}$ using a BDF or Runge-Kutta method, and then compute y_1 with DIMSEM2, y_2 with DIMSEM3, etc. This would allow the parallel DIMSEMs

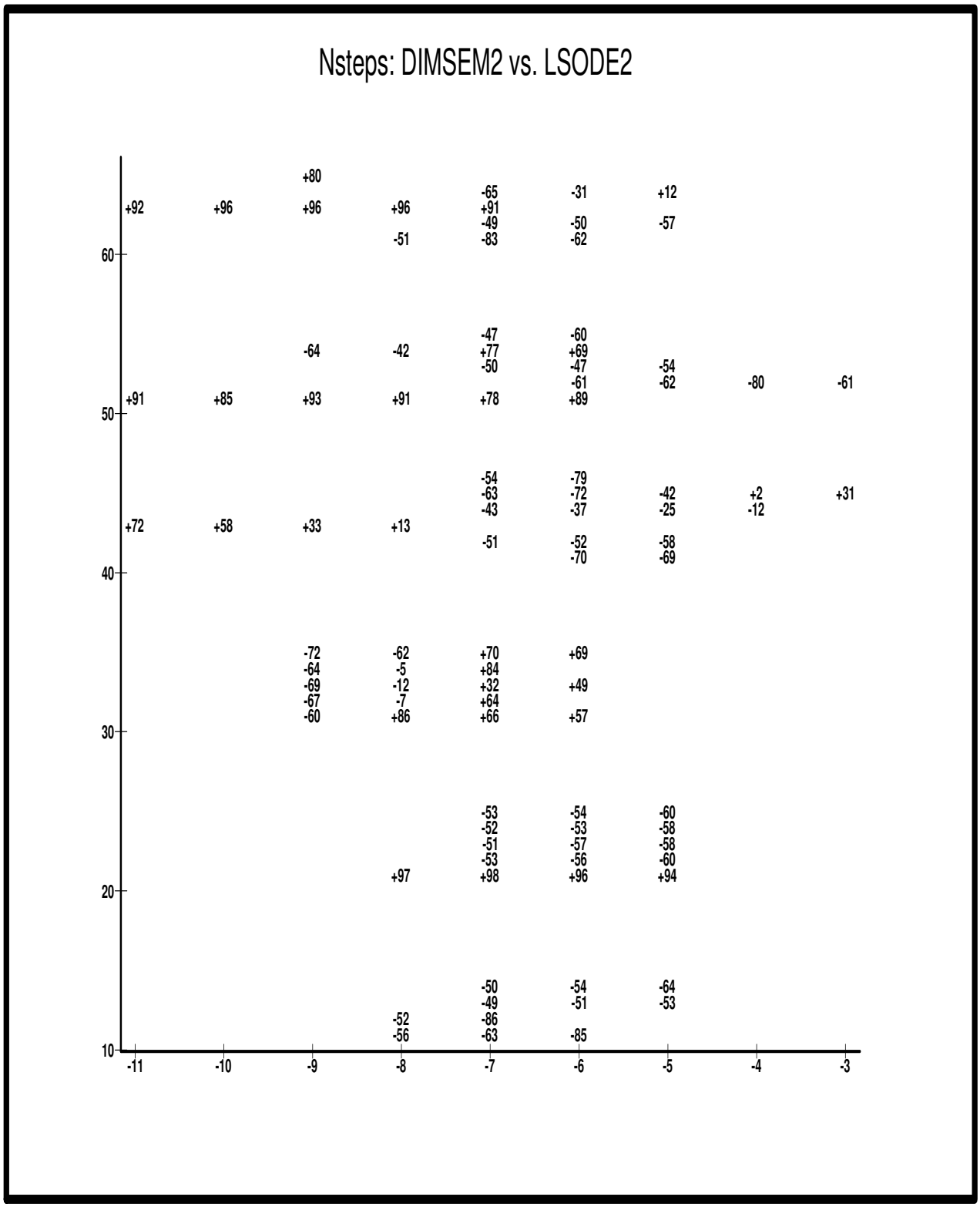


Figure 3.2: Percent Fewer (-) or More (+) Steps Taken by DIMSEM2 vs. LSODE2: Scaled Stiff DETEST Problem Number vs. $\log_{10}$ Endpoint Global Error

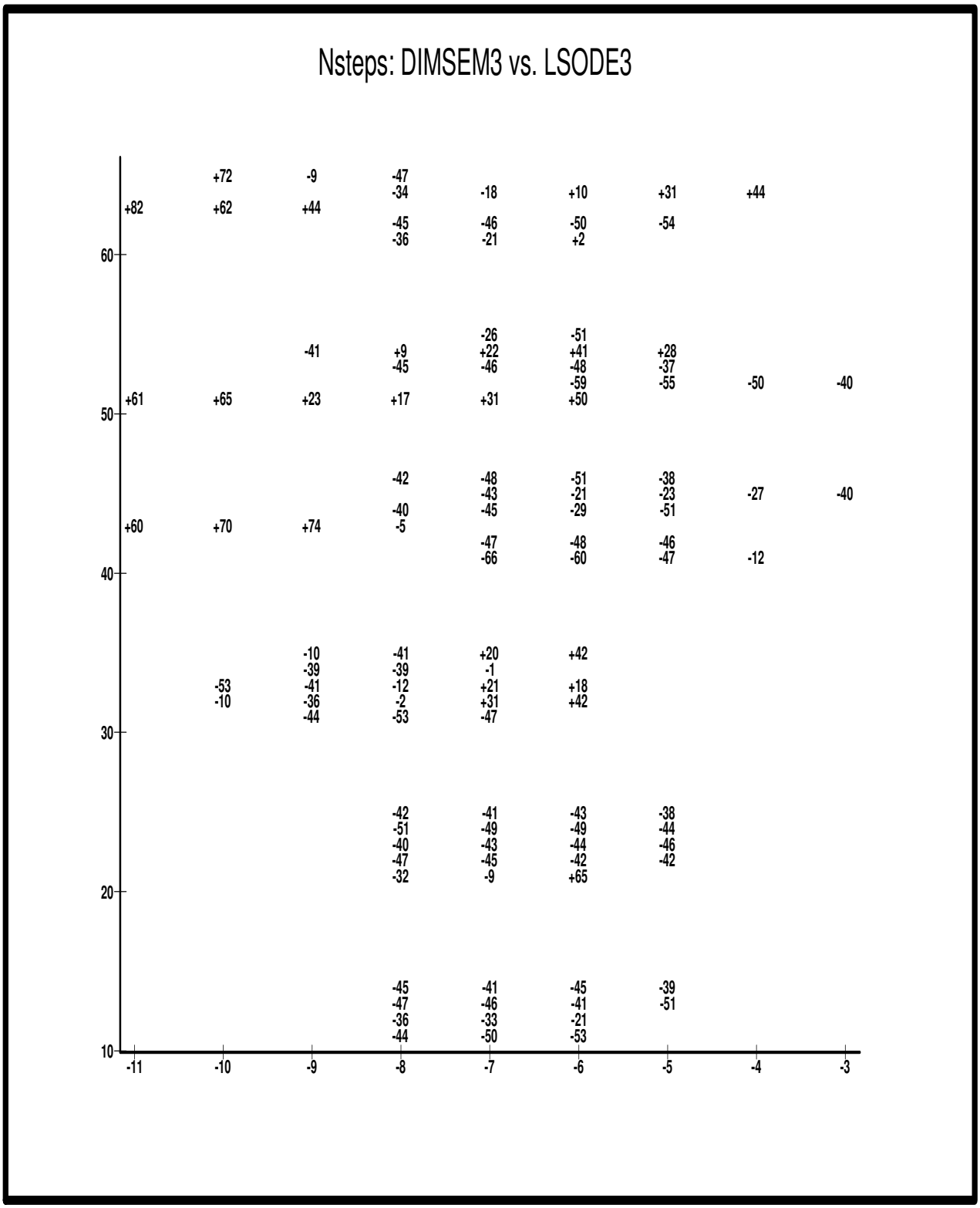


Figure 3.3: Percent Fewer (-) or More (+) Steps Taken by DIMSEM3 vs. LSODE3: Scaled Stiff DETEST Problem Number vs. $\log_{10}$ Endpoint Global Error

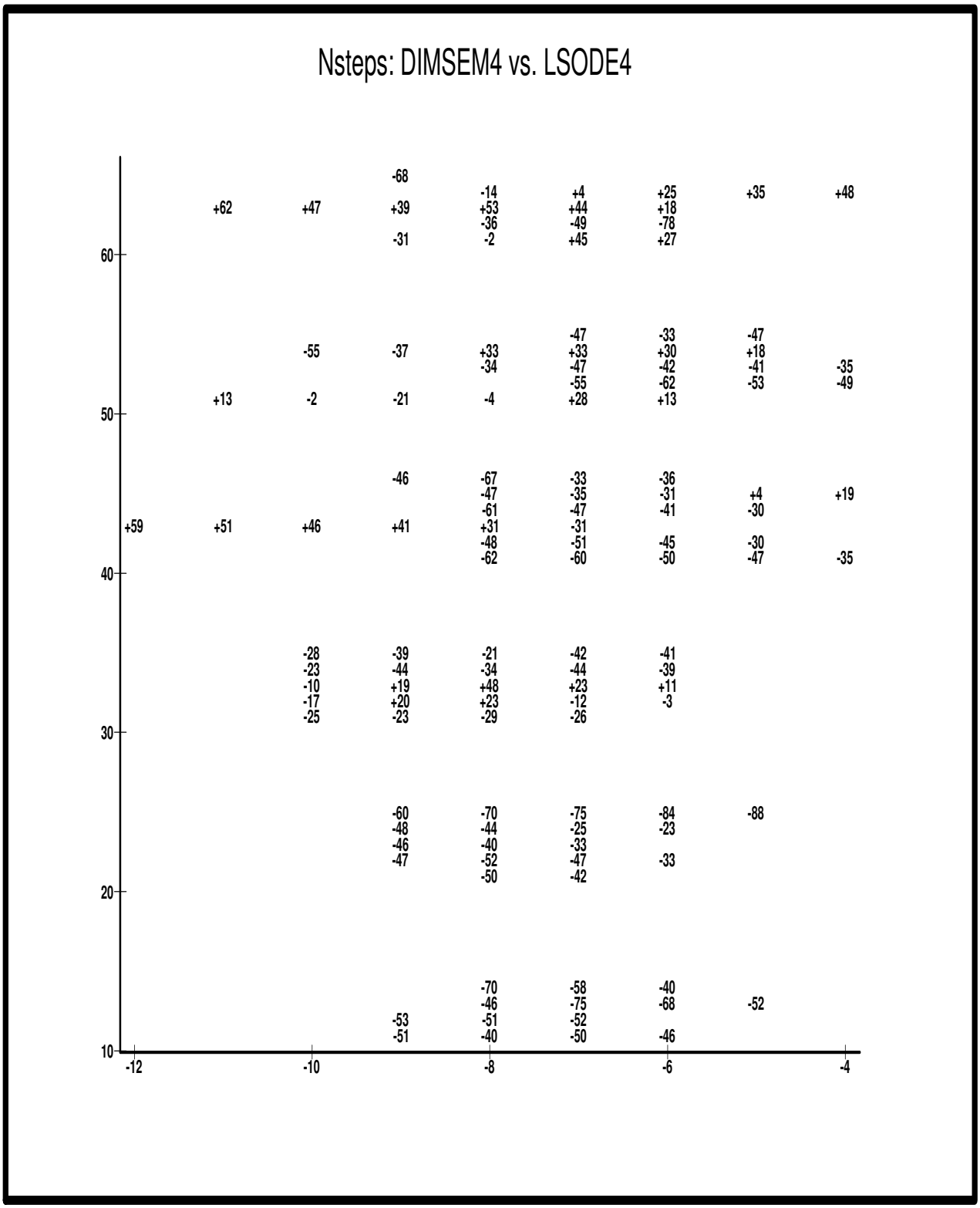


Figure 3.4: Percent Fewer (-) or More (+) Steps Taken by DIMSEM4 vs. LSODE4: Scaled Stiff DETEST Problem Number vs. $\log_{10}$ Endpoint Global Error

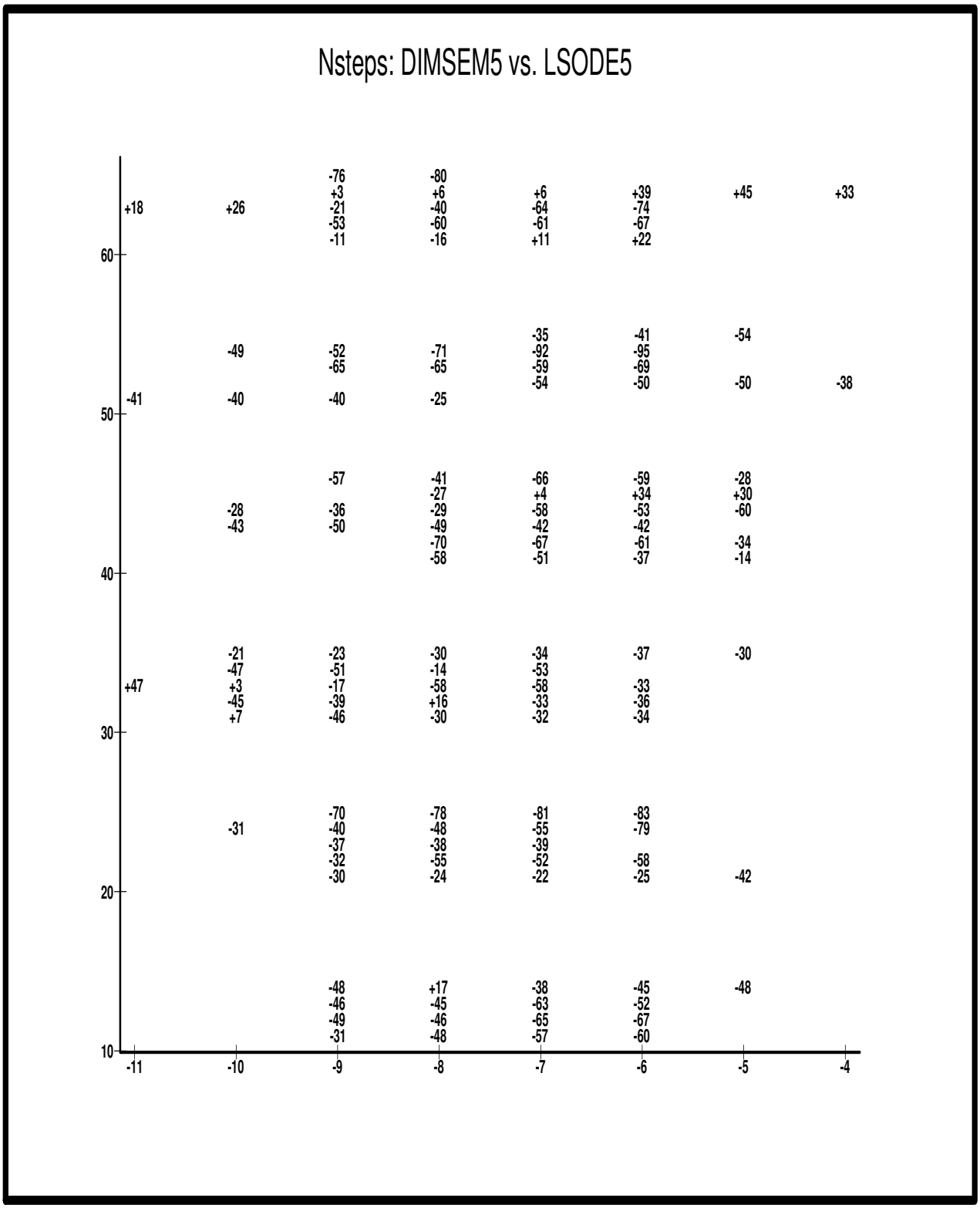


Figure 3.5: Percent Fewer (-) or More (+) Steps Taken by DIMSEM5 vs. LSODE5: Scaled Stiff DETEST Problem Number vs. $\log_{10}$ Endpoint Global Error

to be used sooner.

To satisfy the requirement, employed to assure stability, that $r-1$ steps be taken with the same stepsize before a stepsize change is allowed, it is not necessarily possible simply to re-take a failed step. Instead, a failed step causes the method to back up to the most recent point where a stepsize change would be allowed under the stability requirement. Although this strict approach is not necessary for all problems, it was not clear how to determine exactly when it is necessary, and so it was used at all times in the test code. Failed steps therefore incur a higher cost than in methods that can re-take only the step that failed. Although this is not obvious from the test results reported in the next section, reduced performance of the prototype implementation can be expected on integrations with a large proportion of failed steps.

The convergence criterion for the Newton iteration was that the maximum norm of the update vector be less than the local error tolerance. Because this criterion would often cause two iterations to be taken on linear problems (which are solved exactly in one Newton iteration, up to roundoff), the method was told if the problem is linear, in which case it was made to use exactly one iteration. LSODE uses a more sophisticated convergence criterion which almost always terminates after one corrector iteration on linear problems, without a priori knowledge of linearity.

The solution to the problem at the end of the integration was taken as $y_{n,r}$, the most advanced component of the external stage vector.

The heuristics used in the variable-stepsize DIMSEM code were adopted from VODE [2] for convenience. (In some cases, the comments in the VODE code differ slightly from the FORTRAN statements; in such cases we used the FORTRAN statements.) Although it is possible that some performance gain might result from tuning the parameters to specifically suit the DIMSEM code, this was not investigated, and the VODE parameters were used directly. The DIMSEM heuristics are summarised below.

- Minimum stepsize-change ratio = 0.1
- $$\text{Maximum stepsize-change ratio} = \begin{cases} 10^4 & \text{on 1st step} \\ 10 & \text{on 2nd and later accepted steps} \\ 1 & \text{on 1st retry of failed step} \\ 0.2 & \text{on 2nd retry of failed step} \end{cases}$$
- On 2nd-7th retry of a failed step, use $h \leftarrow h/10$. Give up after 7 retries.

- Declare Newton convergence failure after 3 failed iterations.
- On Newton convergence failure, use $h \leftarrow h/4$ and retry step. Give up after 10 retries.
- Increase h only if it increases by a factor of at least 1.5.
- Update and factorise Newton matrix:
  - on 1st step,
  - after Newton failure,
  - at least every 20 steps,
  - if stepsize has changed by $>$ 30% since last update.
- Update Jacobian if

  (Newton failure on this step
  OR Jacobian > 50 steps old)
  AND (No Newton failure with old Jacobian on this step
  OR stepsize has changed < 20 % since last update of Newton matrix)
- If Jacobian evaluation or Newton matrix factorisation is done on a step, it is done only before first Newton iteration, and not after each iteration.

3.7 Parallel Performance Testing

3.7.1 Parallel Machine

The parallel testing was done on a Silicon Graphics Challenge L computer, with eight 150 MHz R4400 processors, 512 MB of memory, and running IRIX 5.3. The programs were written in FORTRAN 77, and used the native SGI parallel directives. This machine was chosen for parallel performance testing because of its simple, shared-memory architecture, and consequent freedom from performance-distorting architectural features such as non-homogeneous communication paths and memory access times. The only consideration needed in this regard was to separate simultaneously-written memory by at least 64 bytes, in order to avoid cache conflicts. Although it was not possible to ensure exclusive access to the processors during performance testing, the tests were run when the system load

was near zero, giving sufficiently consistent results. Timing was done with the FORTRAN function dtime, based on a clock with microsecond resolution.

Although the primary source of parallelism is in the simultaneous Newton iterations, the overhead incurred in starting a parallel process was found to be sufficiently small that the various smaller matrix-vector operations, for example, the computation of the external stages, could also be effectively parallelised.

3.7.2 Test Problem

We desired a large, non-linear, real-world test problem with expensive function and Jacobian evaluations, for speedup can be expected to suffer if too little time is spent in the major parallel section of the program (see (3.60)). For this reason, a parallel method such as our DIMSEM implementation is only useful on sufficiently expensive problems.

The problem chosen was Problem #8 in [1], an expression of the graph-theoretic travelling salesman problem as an ODE via a neural-network approach. The ODE is

$$\begin{aligned} u'_{ij}(t) &= -\frac{1}{\tau}u_{ij}(t) - A\sum_{\substack{k=1\\k\neq j}}^{n} V_{ik}(t) - B\sum_{\substack{k=1\\k\neq i}}^{n} V_{kj}(t) - C\left(\sum_{k=1}^{n}\sum_{l=1}^{n} V_{kl}(t) - p\right) \\ &\quad -D\sum_{k=1}^{n} d_{ik}(V_{k,j+1}(t) + V_{k,j-1}(t)), \quad i,j = 1,\ldots,n, \end{aligned} \tag{3.56}$$

where d_{ik} is the distance between cities i and k (placed equidistantly around the unit circle),

$$\begin{aligned} V_{ij}(t) &= \frac{1}{2}\left(1 + \tanh\frac{u_{ij}(t)}{u_0}\right), \\ n &= 10, \\ A &= 500, \\ B &= 500, \\ C &= 200, \\ D &= 500, \\ u_0 &= 0.02, \\ p &= 15, \\ \tau &= 1, \end{aligned}$$

and the initial condition is

$$u_{ij}(0) = a + \delta_{ij}, \tag{3.57}$$

where the δ_{ij}'s are uniformly randomly chosen in $[-0.1u_0, 0.1u_0]$, and a satisfies

$$\frac{1}{2}\left(1 + \tanh\frac{a}{u_0}\right) = 0.1.$$

(The random values used were identical for all the test runs.)

The dimension of the problem is 100. The problem is stiff, with a full Jacobian having an eigenvalue of maximum magnitude of approximately -3.8×10^5 at the initial point $t = 0$. All the eigenvalues are real. The range of integration was from 0 to 1. A true solution for comparison purposes was computed by LSODE with a maximum absolute error tolerance of 10^{-13}, the most stringent value for which the integration was successful in double precision with the default heuristics (including a maximum of 10 convergence failures per step).

3.7.3 DIMSEM Speedup Testing

Some simple measures of parallel performance are speedup and efficiency. Speedup can be defined in various ways, depending on the purpose of the measurement. To isolate the effect of the quality of the programming and utilisation of the parallel machine, the algorithm can be compared to itself via

$$S = \frac{\text{elapsed time for the parallel algorithm on 1 processor}}{\text{elapsed time for the parallel algorithm on } n \text{ processors.}} \tag{3.58}$$

The maximum such speedup is (usually) n, and the efficiency $E = S/n$ indicates what proportion of the maximum has been attained. To determine the practical value of the parallel algorithm, speedup can be defined instead as

$$S^* = \frac{\text{elapsed time for a good sequential algorithm}}{\text{elapsed time for the parallel algorithm on } n \text{ processors.}} \tag{3.59}$$

We use (3.58) here, and consider (3.59) in the next section.

The first test explored the speedups attained by each DIMSEMp on s processors compared to DIMSEMp on 1 processor, solving problem (3.56) with a maximum absolute error tolerance of 10^{-4}. The speedups were compared to the bound provided by Amdahl's law, which states that

$$S \leq A = \frac{P + Q}{P/n + Q}, \tag{3.60}$$

where P is the time spent in the parallel section of the code when it is run on a single processor, Q is the time spent in the sequential section, and n is the number of processors that can be applied to the parallel section.

The computation of the initial external stages $y_{0,i},\ i = 2, \ldots, r$, for the DIMSEMs was done using LSODE, simply because it is convenient ($y_{0,1}$ comes directly from the initial condition). A more sophisticated implementation might use LSODE to compute $y_{0,2}$ and $y_{0,3}$, and then use DIMSEMp to compute $y_{0,p+2}$ for $p \geq 2$, allowing parallel computation to take place during the startup phase. In the test implementation, the approach taken however was to keep the timing statistics for the LSODE startup phase separate from that for the rest of the integration. Simply omitting the startup times then provides a reasonable approximation to the situation encountered in a long integration, in which the startup is a negligible proportion of the whole.

The results in Table 3.10 are from a run on one processor, and show the LSODE startup time, the time spent in all other sequential sections, and the time spent in the parallel sections, in columns 2-4 respectively. The remaining sequential sections are sufficiently small that the Amdahl speedup bounds are practically optimal.

The results in Table 3.11 are from runs on both one and s processors. The sequential LSODE startup phase is included in the timings. The efficiencies range from 76% to 87%, significantly less than optimal due to the non-parallelisable startup. However, when the startup time is excluded in Table 3.12, the efficiencies are quite good, ranging from 87% to 94%. This asymptotic level of performance can be expected to be approached as the integration becomes longer.

Besides the startup phase, another possible source of lost speedup is poor load balance resulting when the Newton iterations for each internal stage, each assigned to its own processor, converge after different numbers of iterations. On this particular problem, however, the stages converged in the same number of iterations on almost every step, and only rarely were the stages split between one and two, or two and three, iterations.

It is worth explaining why the times reported in Tables 3.10-3.12 increase with the order of the method. There are two reasons for this. First, these integrations were run at a fixed local-error tolerance of 10^{-4}, and the higher-order methods are achieving tighter global errors. Second, 10^{-4} is a sufficiently lax tolerance for this problem that the lower-order methods are more efficient. (This phenomenon could be taken advantage of by a variable-order scheme.) A comparison of the performance of the various DIMSEMs over a range of tolerances is provided in the next section, where the higher-order methods are shown to be

order	startup time	sequential time	parallel time	Amdahl speedup	Amdahl efficiency
2	1.46	0.07	13.47	1.99	0.99
3	2.17	0.06	21.07	2.98	0.99
4	2.74	0.10	41.67	3.97	0.99
5	3.16	0.12	53.36	4.96	0.99
6	3.73	0.17	86.52	5.94	0.99

Table 3.10: DIMSEM Amdahl's Law Performance (ignoring LSODE startup phase). Problem (3.56), $TOL = 10^{-4}$.

order	time on 1 processor	time on s processors	speedup	efficiency
2	15.00	8.64	1.74	0.87
3	23.30	9.94	2.34	0.78
4	44.50	14.73	3.02	0.76
5	56.64	14.86	3.81	0.76
6	90.43	19.78	4.57	0.76

Table 3.11: DIMSEM Parallel Speedups (including LSODE startup phase). Problem (3.56), $TOL = 10^{-4}$.

more efficient at stringent tolerances.

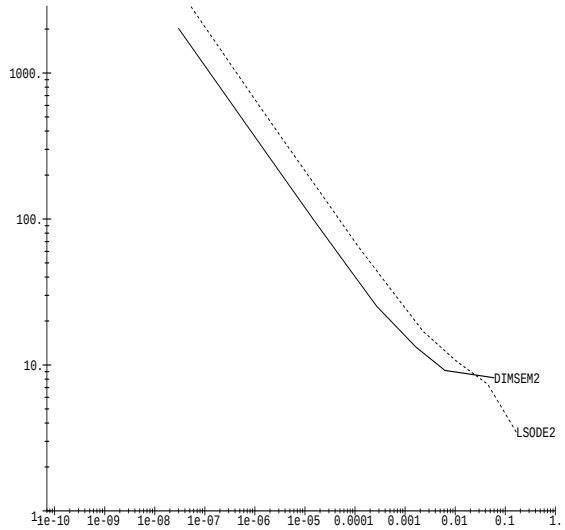
3.7.4 Comparison of Parallel DIMSEMs with Fixed-Order LSODE

In this section we investigate the effectiveness of our DIMSEM implementation by comparing it to the fixed-order version of LSODE. The problem (3.56) was integrated from $t = 0$ to 1, as in the previous section, but for a range of tolerances 10^{-k}, $k = 1, \ldots, 11$. Since the runs at stringent tolerances took considerable CPU time, which was expensive on the parallel machine, they were done instead on a sequential machine and the elapsed times were corrected by the speedups in Table 3.12. To verify the accuracy of this indirect procedure, DIMSEM4 and LSODE4 were used to solve the problem on the parallel machine for a reduced range of tolerances, and good agreement with the corrected sequential timings was obtained. The timing for the DIMSEMs includes the LSODE startup phase. The sequential machine used was a SUN SPARCstation 20/172 with two 75 MHz processors and 256 MB of memory, running SunOS 5.4. As for the parallel testing, timing was done with the FORTRAN function dtime, based on a clock with microsecond resolution.

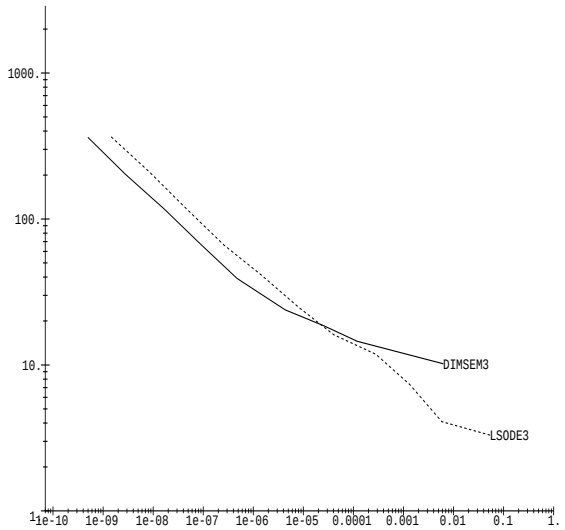
order	time on 1 processor	time on s processors	speedup	efficiency
2	13.54	7.17	1.89	0.94
3	21.12	7.76	2.72	0.91
4	41.77	11.98	3.49	0.87
5	53.47	11.66	4.59	0.92
6	86.70	16.02	5.41	0.90

Table 3.12: DIMSEM Parallel Speedups (ignoring LSODE startup phase). Problem (3.56), $TOL = 10^{-4}$.

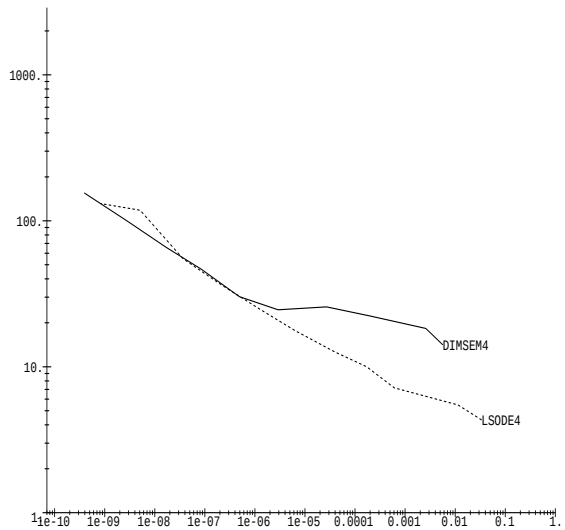
The results are shown in Fig. 3.6. In each case, the performance of the DIMSEM decreases at more relaxed tolerances, and decreases more with increasing order. This suggests a fixed overhead in the methods, and may have to do with the use of LSODE as a starting method for the DIMSEMs. At the most relaxed tolerances, LSODEp always outperforms DIMSEMp. For sufficiently stringent tolerances, however, DIMSEMp outperforms LSODEp by a factor of 30% to 50%, demonstrating the efficacy of the DIMSEM methods.



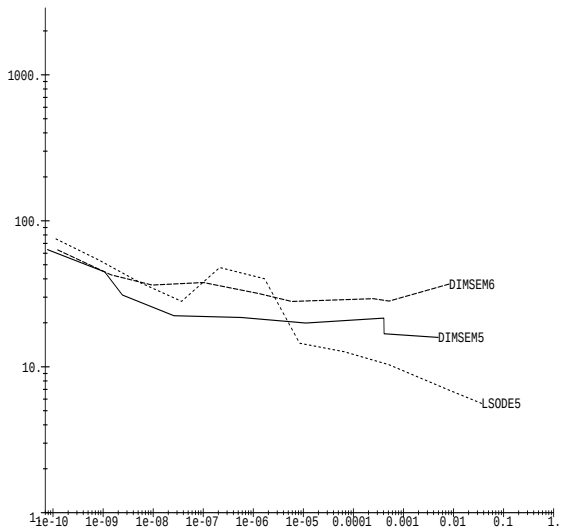
LSODE2 and DIMSEM2



LSODE3 and DIMSEM3



DIMSEM4 and LSODE4



DIMSEM5, DIMSEM6 and LSODE5

Figure 3.6: Elapsed Time (sec) vs. Endpoint Global Error for Parallel DIMSEM and Fixed-Order LSODE on Problem (3.56)

Chapter 4

Conclusions and Future Work

In this thesis, we have introduced two new classes of methods which show promise for the solution of stiff ODEs on parallel computers. Both the LDM and DIMSEM classes contain methods of all orders ≥ 2. L-stable methods of orders 2–6 from each of these classes have been derived, and have behaved satisfactorily in fixed-stepsize numerical testing. A fixed-order, variable-stepsize implementation of the DIMSEM class has been developed, and testing performed on sequential and parallel machines compared the performance of the DIMSEMs to a fixed-order version of LSODE. While the DIMSEM implementation is not a production code, it has demonstrated the potential of DIMSEMs by showing them to be competitive with fixed-order LSODE on a practical test problem. In addition, the L-stability of the methods implies that they can be expected to outperform multistep methods on problems with eigenvalues near the imaginary axis.

There is much further research to do on DIMSEMs. It is possible that the DIMSEM class developed in this work has suffered from the restriction, made early on for the sake of simplicity when solving the order and single-eigenvalue conditions, that the internal abscissas $c_1, \ldots, c_s$ are all zero. An interesting possibility for future research would be to attempt to derive DIMSEMs for which at least some of the c_i's are free parameters, and use them to reduce the error coefficients, making the methods more efficient. This would likely require a new approach to proving Theorems 4 and 5 relating to the single-eigenvalue conditions. The external abscissas d_i's are also possible free parameters. Further generalisations might be to use the Nordsieck type, or some other type, of true-value function in place of the multistep-type true-value function employed here. Also, the nonlinear stability of the methods should be investigated. A production code employing both variable stepsize and variable order is a reasonable goal, which would require further study of robust local error estimators.

The parallel performance exhibited by the DIMSEMs in this thesis suggests that further research on such methods would be worthwhile.

Appendix A

Theorem 4

Theorem 4 *Let* $b_{ij} = (\bar{B}_2)_{ij}$. *Let*

$$s = r - 1 \geq 1,$$

$$c_i = 0, \quad i = 1, \ldots, s,$$

$$b_{rj} = \frac{1}{s!} \frac{\sum_{k=0}^{s-1} \frac{\theta_{kj}^{(s)}}{(s-k)!}}{\prod_{\substack{l=1 \\ l \neq j}}^{s} (\lambda_l - \lambda_j)} \tag{A.1}$$

$$= \frac{1}{s! \lambda_j^{s-1} p_j(\frac{1}{\lambda_j})} \sum_{k=0}^{s-1} \frac{\theta_{kj}^{(s)}}{(s-k)!}, \tag{A.2}$$

where

$$\theta_{kj}^{(s)} \text{ is the coefficient of } x^k \text{ in } p_j(x) \tag{A.3}$$

and

$$p_j(x) = \prod_{\substack{l=1 \\ l \neq j}}^{s} (\lambda_l x - 1). \tag{A.4}$$

Let

$$b_{ij} = T_{ij}(\lambda) b_{rj}, \quad i = 1, \ldots, r-1, \tag{A.5}$$

where $\lambda = (\lambda_1, \ldots, \lambda_s)$ *with* $\lambda_i, i = 1, \ldots, s$, *distinct and nonzero,*

$$T_{ij}(\lambda) = \frac{1}{(i-1)!} \sum_{l=0}^{r-i} \left[(i-1+l)! \gamma_{s-i-l+1}^{(s)} \lambda_j^l \right], \tag{A.6}$$

and

$$\gamma_k^{(s)} = \begin{cases} \sum_{1 \le i_1 < \cdots < i_k \le s} i_1 \cdots i_k, & k \ge 1, \\ 1, & k = 0. \end{cases} \tag{A.7}$$

Then the transformed stability matrix $\bar{M}(z)$ *has* $r-1$ *zero eigenvalues for all* $z \neq 1/\lambda_i$*,* $i = 1, \ldots, s$*.*

Remark. The requirement that $z \neq 1/\lambda_i$ for $i = 1, \ldots, s$ is a necessary restriction, since the factor $(I - zB_1)$ that occurs in both $M(z)$ and $\bar{M}(z)$ is singular if and only if $z = 1/\lambda_i$ for some $i = 1, \ldots, s$. Note that, if we choose $\lambda_i > 0$ for $i = 1, \ldots, s$, then $M(z)$ and $\bar{M}(z)$ are well defined and the single eigenvalue property holds for all $z \in \mathbb{C}^-$.

Proof. Write $\bar{M}_r(z)$ for $\bar{M}(z)$ to emphasize the dependence of the transformed stability matrix on r. Theorem 5 in Appendix B then gives the following expression for the characteristic polynomial of $\bar{M}_r(z)$:

$$\phi_r(w, z) = \sum_{k=0}^{r} \phi_k^{(r)}(z) w^k, \tag{A.8}$$

where

$$\phi_k^{(r)}(z) = \begin{cases} \delta_k^{(s)} + \sum_{i=1}^{r} \left[(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z) a_i(z) \right], & k = 0, \ldots, r-1, \\ 1, & k = r, \end{cases} \tag{A.9}$$

and

$$a_i(z) = \sum_{k=1}^{s} \frac{b_{ik}}{1 - \lambda_k z}. \tag{A.10}$$

(The other quantities are defined in Theorem 5.) We will show that $\phi_r(w, z)$ has all but one of its w-roots at zero for all $z \neq 1/\lambda_i$, $i = 1, \ldots, s$. This is equivalent to

$$\phi_k^{(r)}(z) = 0, \quad \text{for } k = 0, \ldots, r-2 \text{ and } z \neq 1/\lambda_i \text{ for } i = 1, \ldots, s, \tag{A.11}$$

for then

$$\begin{aligned} \phi_r(w, z) &= \phi_r^{(r)}(z) w^r + \phi_{r-1}^{(r)}(z) w^{r-1} \\ &= \left[\phi_r^{(r)}(z) w + \phi_{r-1}^{(r)}(z) \right] w^{r-1}, \end{aligned}$$

which has a root of multiplicity $r-1$ at $w = 0$.

From (A.9) and (A.10), we have

$$\phi_k^{(r)}(z) = \delta_k^{(s)} + \sum_{i=1}^{r} \left[(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z) \sum_{j=1}^{s} \frac{b_{ij}}{1 - \lambda_j z} \right]$$

$$
\begin{aligned}
&= \delta_k^{(s)} + \sum_{i=1}^{r}\sum_{j=1}^{s}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z\right)\frac{b_{ij}}{1-\lambda_j z} \\
&= \delta_k^{(s)} + \sum_{i=1}^{r}\frac{\sum_{j=1}^{s}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z\right)\prod_{\substack{l=1\\ l\neq j}}^{s}(1-\lambda_l z)}{\prod_{j=1}^{s}(1-\lambda_j z)} b_{ij} \\
&= \delta_k^{(s)} + \sum_{i=1}^{r}\frac{\sum_{j=1}^{s}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z\right)(-1)^{s-1} p_j(z)}{(-1)^s p(z)} b_{ij} \\
&= \frac{\delta_k^{(s)} p(z) - \sum_{i=1}^{r}\sum_{j=1}^{s}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z\right) p_j(z) b_{ij}}{p(z)} \\
&= \frac{q_k(z)}{p(z)}, \qquad \text{(A.12)}
\end{aligned}
$$

where

$$
\begin{aligned}
q_k(z) &= \delta_k^{(s)} p(z) - \sum_{i=1}^{r}\sum_{j=1}^{s}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z\right) p_j(z) b_{ij}, \qquad \text{(A.13)} \\
p(z) &= \prod_{j=1}^{s}(\lambda_j z - 1). \qquad \text{(A.14)}
\end{aligned}
$$

Note that $q_k(z)$ is a polynomial of degree s. To prove (A.11) then, it suffices to show $q_k(z) = 0$ at $s+1$ different z's. Then,

$$
\begin{aligned}
& q_k(z) = 0 \quad \forall z \\
\Longrightarrow \quad & \phi_k^{(r)}(z) = q_k(z)/p(z) = 0 \quad \forall z \ni p(z) \neq 0 \\
\Longrightarrow \quad & \phi_k^{(r)}(z) = 0 \quad \forall z \neq \frac{1}{\lambda_i}, \quad i = 1, \ldots, s,
\end{aligned}
$$

which is (A.11). The values we choose for z to prove $q_k(z) = 0$ are

$$
z = \frac{1}{\lambda_1}, \ldots, \frac{1}{\lambda_s}, 0, \qquad \text{(A.15)}
$$

which are distinct and well-defined because the λ_i's are distinct and nonzero. Then from (A.4) and (A.14) we have

$$
\begin{aligned}
p\left(\frac{1}{\lambda_j}\right) &= 0, \quad j = 1, \ldots, s, \\
p_j\left(\frac{1}{\lambda_m}\right) &= 0, \quad m \neq j.
\end{aligned}
$$

Consider $z = 1/\lambda_j$ first. Applying the above expressions to (A.13) gives

$$q_k\left(\frac{1}{\lambda_j}\right) = -\psi_{kj}, \tag{A.16}$$

where

$$\psi_{kj} = \sum_{i=1}^{r}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)}\frac{1}{\lambda_j}\right) p_j\left(\frac{1}{\lambda_j}\right) b_{ij}. \tag{A.17}$$

Now we have, using (A.17) and (A.5),

$$\begin{aligned}\psi_{kj} &= \sum_{i=1}^{r}\left(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)}\frac{1}{\lambda_j}\right) p_j\left(\frac{1}{\lambda_j}\right) T_{ij} b_{rj} \\ &= \frac{1}{\lambda_j} p_j\left(\frac{1}{\lambda_j}\right) b_{rj} \sum_{i=1}^{r}\left(\lambda_j \alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)}\right) T_{ij}.\end{aligned} \tag{A.18}$$

Thus

$$\psi_{kj} = 0 \Longleftrightarrow \begin{cases} \frac{1}{\lambda_j} p_j\left(\frac{1}{\lambda_j}\right) b_{rj} = 0, \text{ or} \\ \sum_{i=1}^{r}\left(\lambda_j \alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)}\right) T_{ij} = 0. \end{cases} \tag{A.19}$$

Consider the first alternative in (A.19). The first factor $1/\lambda_j \neq 0$. Also,

$$p_j\left(\frac{1}{\lambda_j}\right) = \prod_{\substack{l=1 \\ l\neq j}}^{s}\left(\frac{\lambda_l}{\lambda_j} - 1\right) \neq 0,$$

because the λ_j's are distinct. We could have $b_{rj} = 0$, but, by (A.1), this requires

$$\sum_{k=0}^{s-1} \frac{\theta_{kj}^{(s)}}{(s-k)!} = 0$$

which imposes a condition on the λ_j's (through $\theta_{kj}^{(s)}$), and we prefer to keep the λ_j's as free parameters. It turns out that the second alternative in (A.19) is always true, so we will prove

$$\sum_{i=1}^{r}\left(\lambda_j \alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)}\right) T_{ij} = 0, \quad j = 1, \ldots, s, \quad k = 0, \ldots, r-2. \tag{A.20}$$

From (A.15), we must also show $q_k(0) = 0$. Now $p(0) = (-1)^s$ and $p_j(0) = (-1)^{s-1}$, so by (A.13), $q_k(0) = 0$ is equivalent to

$$\delta_k^{(s)} + \sum_{i=1}^{r}\sum_{j=1}^{s} \alpha_{k,i}^{(s)} b_{ij} = 0, \quad k = 0, \ldots, r-2. \tag{A.21}$$

Therefore we will show (A.20) and (A.21). Consider (A.20) first. Using (B.4) gives

$$\begin{aligned}
\text{(A.20)} &\iff \sum_{i=1}^{r}\left(-(i-1)\lambda_j\beta_{k,i-1}^{(s)}T_{ij}\right)+\sum_{i=1}^{r}\beta_{k,i}^{(s)}T_{ij}=0\\
&\iff \sum_{i=1}^{r-1}(-\lambda_j iT_{i+1,j}+T_{ij})\beta_{k,i}^{(s)}+\beta_{k,r}^{(s)}=0. \qquad \text{(A.22)}
\end{aligned}$$

From (A.6) and (A.7), we have

$$\begin{aligned}
& -\lambda_j iT_{i+1,j}+T_{ij}\\
= \;& -\lambda_j i\sum_{l=0}^{r-i-1}\frac{(i+l)!}{i!}\gamma_{s-i-l}^{(s)}\lambda_j^l+\sum_{l=0}^{r-i}\frac{(i-1+l)!}{(i-1)!}\gamma_{s-i-l+1}^{(s)}\lambda_j^l\\
= \;& -\sum_{l=1}^{r-i}\frac{(i-1+l)!}{(i-1)!}\gamma_{s-i-l+1}^{(s)}\lambda_j^l+\sum_{l=0}^{r-i}\frac{(i-1+l)!}{(i-1)!}\gamma_{s-i-l+1}^{(s)}\lambda_j^l\\
= \;& \gamma_{s-i+1}^{(s)},
\end{aligned}$$

because all but the $l=0$ term cancels. Using this in (A.22) reduces what we must prove to

$$\sum_{i=1}^{r-1}\gamma_{s-i+1}^{(s)}\beta_{k,i}^{(s)}+\beta_{k,r}^{(s)}=0,\quad k=0,\ldots,r-2,$$

or

$$\sum_{i=1}^{r}\gamma_{s-i+1}^{(s)}\beta_{k,i}^{(s)}=0,\quad k=0,\ldots,r-2, \qquad \text{(A.23)}$$

since $\gamma_0^{(s)}=1$ by (A.7). Using (B.6), we rewrite (B.5) as

$$\beta_{k,i}^{(s)}=\begin{cases}(-1)^{s+1-k}\begin{pmatrix}s\\k\end{pmatrix}, & i=1,\\ (-1)^{s-k}\left[\sum_{l=0}^{s-k-1}(-1)^l\begin{pmatrix}s-l-1\\k\end{pmatrix}\Sigma_l^{i-1}\right], & i\geq 2.\end{cases} \qquad \text{(A.24)}$$

Applying (A.24) to (A.23) further reduces what we must prove to

$$\begin{aligned}
& \gamma_s^{(s)}\beta_{k,1}^{(s)}+\sum_{i=2}^{r}\gamma_{s-i+1}^{(s)}\beta_{k,i}^{(s)}=0\\
\iff \;& \gamma_s^{(s)}\beta_{k,1}^{(s)}+\sum_{i=1}^{s}\gamma_{s-i}^{(s)}\beta_{k,i+1}^{(s)}=0
\end{aligned}$$

$$\iff \quad (-1)^{s+1-k}\gamma_s^{(s)}\left(\begin{array}{c} s \\ k \end{array}\right) + (-1)^{s-k}\sum_{i=1}^{s}\gamma_{s-i}^{(s)}\sum_{l=0}^{s-k-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ k \end{array}\right)\Sigma_l^i = 0$$

$$\iff \quad \gamma_s^{(s)}\left(\begin{array}{c} s \\ k \end{array}\right) + \sum_{i=1}^{s}\gamma_{s-i}^{(s)}\sum_{l=1}^{s-k}(-1)^l\left(\begin{array}{c} s-l \\ k \end{array}\right)\Sigma_{l-1}^i = 0. \tag{A.25}$$

By Lemma 5 in Appendix E.4,

$$\Sigma_{l-1}^i = (-1)^l\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)m^i.$$

Using this in (A.25), we must show

$$\gamma_s^{(s)}\left(\begin{array}{c} s \\ k \end{array}\right) + \sum_{l=1}^{s-k}(-1)^l\left(\begin{array}{c} s-l \\ k \end{array}\right)\sum_{i=1}^{s}\gamma_{s-i}^{(s)}\Sigma_{l-1}^i = 0$$

$$\iff \quad \gamma_s^{(s)}\left(\begin{array}{c} s \\ k \end{array}\right) + \sum_{l=1}^{s-k}(-1)^l\left(\begin{array}{c} s-l \\ k \end{array}\right)\sum_{i=1}^{s}\gamma_{s-i}^{(s)}(-1)^l\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)m^i = 0$$

$$\iff \quad \gamma_s^{(s)}\left(\begin{array}{c} s \\ k \end{array}\right) + \sum_{l=1}^{s-k}\left(\begin{array}{c} s-l \\ k \end{array}\right)\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)U_m^{(s)}, \tag{A.26}$$

where

$$U_m^{(s)} = \sum_{i=1}^{s}\gamma_{s-i}^{(s)}m^i. \tag{A.27}$$

Lemma 6 in Appendix E.5 gives an expression for $U_m^{(s)}$. This together with $\gamma_s^{(s)} = s!$ from (A.7) reduces (A.26) to

$$s!\left(\begin{array}{c} s \\ k \end{array}\right) + \sum_{l=1}^{s-k}\left(\begin{array}{c} s-l \\ k \end{array}\right)\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)s!\left[\left(\begin{array}{c} m+s \\ s \end{array}\right) - 1\right] = 0$$

$$\iff \quad \left(\begin{array}{c} s \\ k \end{array}\right) + \sum_{l=1}^{s-k}\left(\begin{array}{c} s-l \\ k \end{array}\right)\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)\left[\left(\begin{array}{c} m+s \\ s \end{array}\right) - 1\right] = 0. \tag{A.28}$$

Now

$$\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right) = \sum_{m=0}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right) - 1 = -1, \quad l \neq 0, \tag{A.29}$$

by (F.3). Also,

$$\begin{aligned}\sum_{m=1}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)\left(\begin{array}{c} m+s \\ s \end{array}\right) &= \sum_{m=0}^{l}(-1)^m\left(\begin{array}{c} l \\ m \end{array}\right)\left(\begin{array}{c} m+s \\ s \end{array}\right)-1 \\ &= (-1)^l\left(\begin{array}{c} s \\ s-l \end{array}\right)-1=(-1)^l\left(\begin{array}{c} s \\ l \end{array}\right)-1, \end{aligned} \tag{A.30}$$

by (F.7). Using (A.29) and (A.30) in (A.28), we need to show

$$\left(\begin{array}{c} s \\ k \end{array}\right)+\sum_{l=1}^{s-k}\left(\begin{array}{c} s-l \\ k \end{array}\right)\left[(-1)^l\left(\begin{array}{c} s \\ l \end{array}\right)-1-(-1)\right]=0, \quad k=0,\ldots,r-2,$$

which is equivalent to Lemma 7 in Appendix E.6, completing the proof of (A.20).

Now consider (A.21). Letting

$$S_i^{(s)}=\sum_{j=1}^{s}b_{ij}, \tag{A.31}$$

and using (B.4) and (B.3), we rewrite (A.21) as

$$\begin{aligned} & -\sum_{i=1}^{r}\alpha_{k,i}^{(s)}\sum_{j=1}^{s}b_{ij}=\delta_k^{(s)}, \quad k=0,\ldots,r-2 \\ \iff & \sum_{i=2}^{r}(i-1)\beta_{k,i-1}^{(s)}\sum_{j=1}^{s}b_{ij}=\delta_k^{(s)}, \quad k=0,\ldots,r-2 \\ \iff & \sum_{i=1}^{s}i\beta_{k,i}^{(s)}S_{i+1}^{(s)}=(-1)^{s+1-k}\left(\begin{array}{c} s+1 \\ k \end{array}\right), \quad k=0,\ldots,s-1. \end{aligned} \tag{A.32}$$

Using (A.24) in (A.32), we rewrite the left side as

$$\begin{aligned} & (-1)^{s+1-k}\left(\begin{array}{c} s \\ k \end{array}\right)S_2^{(s)}+(-1)^{s-k}\sum_{i=2}^{s}i\sum_{l=0}^{s-k-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ k \end{array}\right)\Sigma_l^{i-1}S_{i+1}^{(s)} \\ = & (-1)^{s+1-k}\left(\begin{array}{c} s \\ k \end{array}\right)S_2^{(s)}+(-1)^{s-k}\sum_{l=0}^{s-k-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ k \end{array}\right)\sum_{i=2}^{s}i\Sigma_l^{i-1}S_{i+1}^{(s)} \\ = & (-1)^{s+1-k}\left(\begin{array}{c} s \\ k \end{array}\right)S_2^{(s)}+(-1)^{s-k}\sum_{l=0}^{s-k-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ k \end{array}\right)M_l^{(s)}, \end{aligned} \tag{A.33}$$

where

$$M_l^{(s)}=\sum_{i=2}^{s}i\Sigma_l^{i-1}S_{i+1}^{(s)}. \tag{A.34}$$

We now make use of Lemmas 6 and 7 in Appendices E.5 and E.6. When $s = 1$, $k = 0, \ldots, s-1$ reduces to $k = 0$, and $M_l^{(s)} = 0$. So (A.33) becomes

$$(-1)^{s+1-k} \left(\begin{array}{c} s \\ k \end{array} \right) 1 = (-1)^{s+1-k} \left(\begin{array}{c} s+1 \\ k \end{array} \right),$$

which equals the right side of (A.32), as required. When $s \geq 2$, (A.33) becomes

$$\begin{aligned}
& (-1)^{s-k} \left[- \left(\begin{array}{c} s \\ k \end{array} \right) s + \sum_{l=0}^{s-k-1} (-1)^l \left(\begin{array}{c} s-l-1 \\ k \end{array} \right) \left(\begin{array}{c} s \\ l+2 \end{array} \right) \right] \\
= \; & (-1)^{s-k} \left[- \left(\begin{array}{c} s \\ k \end{array} \right) s + \sum_{l=2}^{s-k+1} (-1)^l \left(\begin{array}{c} s-l+1 \\ k \end{array} \right) \left(\begin{array}{c} s \\ l \end{array} \right) \right] \\
= \; & (-1)^{s-k} \sum_{l=1}^{s-k+1} (-1)^l \left(\begin{array}{c} s-l+1 \\ k \end{array} \right) \left(\begin{array}{c} s \\ l \end{array} \right) \\
= \; & (-1)^{s-k+1} \sum_{l=0}^{s-k} (-1)^l \left(\begin{array}{c} s-l \\ k \end{array} \right) \left(\begin{array}{c} s \\ l+1 \end{array} \right),
\end{aligned}$$

so that (A.32) becomes

$$\sum_{l=0}^{s-k} (-1)^l \left(\begin{array}{c} s-l \\ k \end{array} \right) \left(\begin{array}{c} s \\ l+1 \end{array} \right) = \left(\begin{array}{c} s+1 \\ k \end{array} \right), \quad k = 0, \ldots, s-1.$$

Letting $N = s - k$, $0 \leq k \leq s-1 \Rightarrow 1 \leq s-k \leq s$, so Lemma 10 in Appendix E.9 suffices. This completes the proof of (A.21), and so we have proven Theorem 4. □

Appendix B

Theorem 5

Theorem 5 *Assume* $z \neq 1/\lambda_i$ *for* $i = 1, \ldots, s$ *and let*

$$\phi_r(w, z) = \textit{det}(wI - \bar{M}_r(z)),$$

where $\bar{M}_r(z) = \bar{M}(z)$ *defined in (2.36). Let*

$$s = r - 1 \geq 1,$$

$$c_i = 0, \quad i = 1, \ldots, s.$$

Then

$$\phi_r(w, z) = \sum_{k=0}^{r} \phi_k^{(r)}(z) w^k, \tag{B.1}$$

where

$$\phi_k^{(r)}(z) = \begin{cases} \delta_k^{(s)} + \sum_{i=1}^{r} \left[(\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z) a_i(z) \right], & k = 0, \ldots, r-1, \\ 1, & k = r, \end{cases} \tag{B.2}$$

$$\delta_k^{(s)} = (-1)^{s+1-k} \begin{pmatrix} s+1 \\ k \end{pmatrix}, \tag{B.3}$$

$$\alpha_{k,i}^{(s)} = \begin{cases} 0, & i = 1, \\ -(i-1)\beta_{k,i-1}^{(s)}, & i \geq 2, \end{cases} \tag{B.4}$$

$$\beta_{k,i}^{(s)} = \begin{cases} (-1)^{s+1-k} \begin{pmatrix} s \\ k \end{pmatrix}, & i = 1, \\ (-1)^{s-k} \left[\begin{pmatrix} s-1 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-1} (-1)^l \begin{pmatrix} s-l-1 \\ k \end{pmatrix} \Sigma_l^{i-1} \right], & i \geq 2, \end{cases} \tag{B.5}$$

$$a_i(z) = \sum_{k=1}^{s} \frac{b_{ik}}{1 - \lambda_k z},$$

and

$$\Sigma_l^n = \begin{cases} \sum_{1 \le i_1 < \cdots < i_l \le n-1} \begin{pmatrix} i_2 \\ i_1 \end{pmatrix} \begin{pmatrix} i_3 \\ i_2 \end{pmatrix} \cdots \begin{pmatrix} i_l \\ i_{l-1} \end{pmatrix} \begin{pmatrix} n \\ i_l \end{pmatrix}, & l \ge 1, \\ 1, & l = 0. \end{cases} \tag{B.6}$$

Remark. Again note that the requirement that $z \neq 1/\lambda_i$ for $i = 1, \ldots, s$ is a necessary restriction, since the factor $(I - zB_1)$ that occurs in both $M(z)$ and $\bar{M}(z)$ is singular if and only if $z = 1/\lambda_i$ for some $i = 1, \ldots, s$. If we choose $\lambda_i > 0$ for $i = 1, \ldots, s$, then $M(z)$ and $\bar{M}(z)$ are well defined and the single-eigenvalue property obtained by Theorem 4 holds for all $z \in \mathbb{C}^-$.

Proof. The proof is by induction on r. From (2.36), our $r \times r$ transformed stability matrix is

$$\bar{M}_r(z) = P + \bar{B}_2(I - zB_1)^{-1}V(zI - U),$$

where

$$P_{ij} = \begin{cases} \begin{pmatrix} j-1 \\ i-1 \end{pmatrix}, & j \ge i \\ 0, & j < i \end{cases},$$

$$B_1 = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_s \end{bmatrix},$$

$$s = r - 1,$$

and $V_{ij} = c_i^{j-1}$, but since all the c_i's are zero,

$$V = \begin{bmatrix} 1 & \\ \vdots & \bigcirc \\ 1 & \end{bmatrix}.$$

Then

$$V(zI-U) = \begin{bmatrix} 1 & \\ \vdots & \bigcirc \\ 1 & \end{bmatrix} \begin{bmatrix} z & -1 & & \\ & \ddots & \ddots & \\ & & \ddots & -(r-1) \\ & & & z \end{bmatrix} = \begin{bmatrix} z, & -1 & \\ \vdots & \vdots & \bigcirc \\ z, & -1 & \end{bmatrix},$$

and

$$(I-zB_1)^{-1}V(zI-U) = \begin{bmatrix} \frac{1}{1-\lambda_1 z} & & \\ & \ddots & \\ & & \frac{1}{1-\lambda_s z} \end{bmatrix} \begin{bmatrix} z, & -1 & \\ \vdots & \vdots & \bigcirc \\ z, & -1 & \end{bmatrix} = \begin{bmatrix} \frac{z}{1-\lambda_1 z}, & \frac{-1}{1-\lambda_1 z} & \\ \vdots & \vdots & \bigcirc \\ \frac{z}{1-\lambda_s z}, & \frac{-1}{1-\lambda_s z} & \end{bmatrix}.$$

Thus, letting $b_{ij} = (\bar{B}_2)_{ij}$ gives

$$\bar{M}_r(z)_{ij} = \begin{cases} P_{ij} + z\sum_{k=1}^{s} \frac{b_{ik}}{1-\lambda_k z}, & j = 1 \\ P_{ij} - \sum_{k=1}^{s} \frac{b_{ik}}{1-\lambda_k z}, & j = 2 \\ P_{ij}, & j \geq 3 \end{cases}$$

or

$$\bar{M}_r(z) = P + \begin{bmatrix} za_1(z), & -a_1(z) & \\ \vdots & \vdots & \bigcirc \\ za_r(z), & -a_r(z) & \end{bmatrix},$$

where

$$a_i(z) = \sum_{k=1}^{s} \frac{b_{ik}}{1-\lambda_k z}.$$

Then

$$\phi_r(w,z) = \mathsf{det}(wI - \bar{M}_r(z)) = (-1)^r \mathsf{det}(\bar{M}_r(z) - wI),$$

so that

$$(-1)^r \phi_r(w,z) = \mathsf{det}(\bar{M}_r(z) - wI) =$$

$$\begin{vmatrix} za_1 + \binom{0}{0} - w & -a_1 + \binom{1}{0} & \binom{2}{0} & \cdots & \binom{r-2}{0} & \binom{r-1}{0} \\ za_2 & -a_2 + \binom{1}{1} - w & \binom{2}{1} & \cdots & \binom{r-2}{1} & \binom{r-1}{1} \\ za_3 & -a_3 & \binom{2}{2} - w & \cdots & \binom{r-2}{2} & \binom{r-1}{2} \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ za_{r-1} & -a_{r-1} & \bigcirc & & \binom{r-2}{r-2} - w & \binom{r-1}{r-2} \\ za_r & -a_r & & & & \binom{r-1}{r-1} - w \end{vmatrix}, \tag{B.7}$$

an $r \times r$ determinant.

As noted earlier, the proof is by induction on $r \geq 2$. First consider the base case $r = 2$ ($s = 1$), for which (B.7) reduces to

$$\begin{aligned} (-1)^2 \phi_2(w,z) &= \begin{vmatrix} za_1 + 1 - w & -a_1 + 1 \\ za_2 & -a_2 + 1 - w \end{vmatrix} \\ &= (za_1 + 1 - w)(-a_2 + 1 - w) - (za_2)(-a_1 + 1) \\ &= (1-w)^2 + (1-w)za_1 - (1-w)a_2 - za_2 \\ &= 1 + za_1 - (1+z)a_2 + (-2 + a_2 - za_1)w + w^2. \end{aligned}$$

Therefore, from (B.1), we must show that

$$\begin{aligned} \phi_0^{(2)} &= 1 + za_1 - (1+z)a_2, \\ \phi_1^{(2)} &= -2 + a_2 - za_1, \\ \phi_2^{(2)} &= 1. \end{aligned}$$

From (B.3), (B.4), and (B.5),

$$\begin{aligned} \delta_0^{(1)} &= (-1)^2 \binom{2}{0} = 1, \\ \delta_1^{(1)} &= (-1)^1 \binom{2}{1} = -2, \end{aligned}$$

$$
\begin{aligned}
\alpha_{0,1}^{(1)} &= \alpha_{1,1}^{(1)} = 0, \\
\beta_{0,1}^{(1)} &= (-1)^2 \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1, \\
\beta_{1,1}^{(1)} &= (-1)^1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} = -1, \\
\alpha_{0,2}^{(1)} &= -\beta_{0,1}^{(1)} = -1, \\
\alpha_{1,2}^{(1)} &= -\beta_{1,1}^{(1)} = 1, \\
\beta_{0,2}^{(1)} &= (-1)^1 \left[\begin{pmatrix} 0 \\ 0 \end{pmatrix} + 0 \right] = -1, \\
\beta_{1,2}^{(1)} &= (-1)^0 \left[\begin{pmatrix} 0 \\ 1 \end{pmatrix} + 0 \right] = 0.
\end{aligned}
$$

Using these expressions in (B.2), we get

$$
\begin{aligned}
\phi_0^{(2)} &= 1 + (0 + 1z)a_1 + (-1 - 1z)a_2 = 1 + za_1 - (1 + z)a_2, \\
\phi_1^{(2)} &= -2 + (0 - 1z)a_1 + (1 + 0z)a_2 = -2 + a_2 - za_1, \\
\phi_2^{(2)} &= 1,
\end{aligned}
$$

as required.

The rest of the proof is devoted to the general case, $r \geq 3$ ($s \geq 2$). Expanding (B.7) along the last row gives

$$
(-1)^r \phi_r(w,z) =
$$

$$
(-1)^{r-1} z a_r \begin{vmatrix}
-a_1 + \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 2 \\ 0 \end{pmatrix} & \cdots & \begin{pmatrix} r-2 \\ 0 \end{pmatrix} & \begin{pmatrix} r-1 \\ 0 \end{pmatrix} \\
-a_2 + \begin{pmatrix} 1 \\ 1 \end{pmatrix} - w & \begin{pmatrix} 2 \\ 1 \end{pmatrix} & \cdots & \begin{pmatrix} r-2 \\ 1 \end{pmatrix} & \begin{pmatrix} r-1 \\ 1 \end{pmatrix} \\
-a_3 & \begin{pmatrix} 2 \\ 2 \end{pmatrix} - w & \cdots & \begin{pmatrix} r-2 \\ 2 \end{pmatrix} & \begin{pmatrix} r-1 \\ 2 \end{pmatrix} \\
\vdots & \bigcirc & \ddots & \vdots & \vdots \\
-a_{r-1} & & & \begin{pmatrix} r-2 \\ r-2 \end{pmatrix} - w & \begin{pmatrix} r-1 \\ r-2 \end{pmatrix}
\end{vmatrix}
$$

$$
+ \quad (-1)^r(-a_r)
\begin{vmatrix}
za_1 + \binom{0}{0} - w & \binom{2}{0} & \cdots & \binom{r-2}{0} & \binom{r-1}{0} \\
za_2 & \binom{2}{1} & \cdots & \binom{r-2}{1} & \binom{r-1}{1} \\
za_3 & \binom{2}{2} - w & \cdots & \binom{r-2}{2} & \binom{r-1}{2} \\
\vdots & & \ddots & \vdots & \vdots \\
za_{r-1} & \bigcirc & & \binom{r-2}{r-2} - w & \binom{r-1}{r-2}
\end{vmatrix}
$$

$$
+ \quad \left[\binom{r-1}{r-1} - w\right]
\begin{vmatrix}
za_1 + \binom{0}{0} - w & -a_1 + \binom{1}{0} & \binom{2}{0} & \cdots & \binom{r-2}{0} \\
za_2 & -a_2 + \binom{1}{1} - w & \binom{2}{1} & \cdots & \binom{r-2}{1} \\
za_3 & -a_3 & \binom{2}{2} - w & \cdots & \binom{r-2}{2} \\
\vdots & \vdots & & \ddots & \vdots \\
za_{r-1} & -a_{r-1} & \bigcirc & & \binom{r-2}{r-2} - w
\end{vmatrix}.
$$

(B.8)

Let us call the three determinants above A, B and C, respectively, so that

$$
(-1)^r\phi_r(w,z) = (-1)^{r-1}za_rA + (-1)^r(-a_r)B + \left[\binom{r-1}{r-1} - w\right]C. \tag{B.9}
$$

Now $C = (-1)^{r-1}\phi_{r-1}(w,z)$ is known by induction, and A and B differ only in the first

column. Let

$$D^{(s)} = \begin{vmatrix} \binom{2}{0} & \binom{3}{0} & \cdots & \binom{r-2}{0} & \binom{r-1}{0} \\ \binom{2}{1} & \binom{3}{1} & \cdots & \binom{r-2}{1} & \binom{r-1}{1} \\ \binom{2}{2} - w & \binom{3}{2} & \cdots & \binom{r-2}{2} & \binom{r-1}{2} \\ & & \ddots & \vdots & \vdots \\ & \bigcirc & & \binom{r-2}{r-2} - w & \binom{r-1}{r-2} \end{vmatrix},$$

and define

$$D_k^{(s)} = D^{(s)} \text{ with the } k\text{-th row crossed out.} \tag{B.10}$$

This gives an $(r-2) \times (r-2)$ determinant equal to columns 2 to $r-1$ of A and B. Expanding A and B down the first column gives

$$A = \left[-a_1 + \binom{1}{0}\right] D_1^{(s)} - \left[-a_2 + \binom{1}{1} - w\right] D_2^{(s)} - \sum_{i=3}^{r} (-1)^{(i-1)} a_i D_i^{(s)},$$

and

$$B = \left[z a_1 + \binom{0}{0} - w\right] D_1^{(s)} - z a_2 D_2^{(s)} + \sum_{i=3}^{r} (-1)^{(i-1)} z a_i D_i^{(s)}.$$

Then (B.9) becomes

$$\begin{aligned} & (-1)^r \phi_r(w,z) \\ = \; & (-1)^{r-1} z a_r A + (-1)^r (-a_r) B + \left[\binom{r-1}{r-1} - w\right] (-1)^{r-1} \phi_{r-1}(w,z) \\ = \; & (-1)^{r-1} \left\{ a_r [zA + B] + \left[\binom{r-1}{r-1} - w\right] \phi_{r-1}(w,z) \right\} \\ = \; & (-1)^{r-1} \left\{ a_r \left[z \left(-a_1 + \binom{1}{0}\right) D_1^{(s)} - z \left(-a_2 + \binom{1}{1} - w\right) D_2^{(s)} \right.\right. \\ & \left.\left. -z \sum_{i=3}^{r} (-1)^{i-1} a_i D_i^{(s)} + \left(z a_1 + \binom{0}{0} - w\right) D_1^{(s)} - z a_2 D_2^{(s)} + \sum_{i=3}^{r} (-1)^{i-1} z a_i D_i^{(s)} \right] \right. \end{aligned}$$

$$+\left[\binom{r-1}{r-1}-w\right]\phi_{r-1}(w,z)\Big\}$$
$$= (-1)^{r-1}\left\{a_r\left[(z+1-w)D_1^{(s)}+(zw-z)D_2^{(s)}\right]+(1-w)\phi_{r-1}(w,z)\right\},$$

which implies

$$\phi_r(w,z) = -a_r\left\{(z+1-w)D_1^{(s)}+(zw-z)D_2^{(s)}\right\}-(1-w)\phi_{r-1}(w,z). \qquad \text{(B.11)}$$

Our goal is to show that (B.2) = (B.11), so we need expressions for $D_1^{(s)}$ and $D_2^{(s)}$. We have from (B.10),

$$D_1^{(s)} = \begin{vmatrix} \binom{2}{1} & \binom{3}{1} & \cdots & \cdots & \binom{r-1}{1} \\ 1-w & \binom{3}{2} & \cdots & \cdots & \binom{r-1}{2} \\ & 1-w & \binom{4}{3} & \cdots & \binom{r-1}{3} \\ & & \ddots & & \vdots \\ & \bigcirc & & 1-w & \binom{r-1}{r-2} \end{vmatrix},$$

and

$$D_2^{(s)} = \begin{vmatrix} 1 & 1 & \cdots & \cdots & 1 \\ 1-w & \binom{3}{2} & \cdots & \cdots & \binom{r-1}{2} \\ & 1-w & \binom{4}{3} & \cdots & \binom{r-1}{3} \\ & & \ddots & & \vdots \\ & \bigcirc & & 1-w & \binom{r-1}{r-2} \end{vmatrix}.$$

Note that $D_1^{(s)}$ and $D_2^{(s)}$ differ only in the first row, so we will expand them along that row. To do this we need to know the $(r-3)\times(r-3)$ determinants $\bar{M}_k^{(s)}$, which are defined by

letting

$$\bar{M}^{(s)} = \begin{vmatrix} 1-w & \binom{3}{2} & \cdots & \cdots & \cdots & \binom{r-1}{2} \\ & 1-w & \binom{4}{3} & \cdots & \cdots & \binom{r-1}{3} \\ & & 1-w & \binom{5}{4} & \cdots & \binom{r-1}{4} \\ & \bigcirc & & \ddots & & \vdots \\ & & & & 1-w & \binom{r-1}{r-2} \end{vmatrix},$$

and

$$\bar{M}_k^{(s)} = \bar{M}^{(s)} \text{ with the } k\text{-th column crossed out.}$$

Then,

$$D_1^{(s)} = \sum_{k=1}^{s-1} (-1)^{k-1} \binom{k+1}{1} \bar{M}_k^{(s)}, \tag{B.12}$$

$$D_2^{(s)} = \sum_{k=1}^{s-1} (-1)^{k-1} \bar{M}_k^{(s)}. \tag{B.13}$$

Note that $\bar{M}_k^{(s)}$ is a block upper-triangular determinant, with the top-left block being itself upper triangular. Thus

$$\bar{M}_k^{(s)} = (1-w)^{k-1} M_{s-k-1}^{(s)}, \tag{B.14}$$

where for $n \geq 1$,

$$M_n^{(s)} = \begin{vmatrix} \binom{s-n+1}{s-n} & \cdots & \cdots & \binom{s}{s-n} \\ 1-w & \binom{s-n+2}{s-n+1} & \cdots & \binom{s}{s-n+1} \\ & \ddots & & \vdots \\ \bigcirc & & 1-w & \binom{s}{s-1} \end{vmatrix}, \tag{B.15}$$

an $n \times n$ determinant, and $M_0^{(s)} = 1$. Then from (B.12), (B.13), and (B.14),

$$D_1^{(s)} = \sum_{k=1}^{s-1} (-1)^{k-1}(k+1)(1-w)^{k-1} M_{s-k-1}^{(s)}, \tag{B.16}$$

$$D_2^{(s)} = \sum_{k=1}^{s-1} (-1)^{k-1}(1-w)^{k-1} M_{s-k-1}^{(s)}. \tag{B.17}$$

We now define a transformation on the arguments of the binomial coefficients in $M_n^{(s)}$, under which properties of the binomial coefficients are not invariant. Therefore, we will consider the binomial coefficient $\begin{pmatrix} i \\ j \end{pmatrix}$ as an ordered pair $\begin{bmatrix} i \\ j \end{bmatrix}$ while we are using the transformation. We begin by writing (B.15) as

$$M_n^{(s)} = \begin{vmatrix} \begin{bmatrix} s-n+1 \\ s-n \end{bmatrix} & \cdots & \cdots & \begin{bmatrix} s \\ s-n \end{bmatrix} \\ 1-w & \begin{bmatrix} s-n+2 \\ s-n+1 \end{bmatrix} & \cdots & \begin{bmatrix} s \\ s-n+1 \end{bmatrix} \\ & \ddots & & \vdots \\ \bigcirc & & 1-w & \begin{bmatrix} s \\ s-1 \end{bmatrix} \end{vmatrix}. \tag{B.18}$$

We now define the transformation $\psi_n^{(s)}$ as follows. For any expression E, $\psi_n^{(s)}(E)$ is found by replacing every occurrence of $\begin{bmatrix} i \\ j \end{bmatrix}$ according to the mapping

$$\begin{bmatrix} i \\ j \end{bmatrix} \longmapsto \begin{bmatrix} i-(s-n) \\ j-(s-n-1) \end{bmatrix} = \begin{bmatrix} i-s+n \\ j-s+n+1 \end{bmatrix}.$$

Then $\psi_n^{(s)}(M_n^{(s)}) = M_n$, where

$$M_n = \begin{vmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} & \begin{bmatrix} 2 \\ 1 \end{bmatrix} & \cdots & \cdots & \begin{bmatrix} n \\ 1 \end{bmatrix} \\ 1-w & \begin{bmatrix} 2 \\ 2 \end{bmatrix} & \cdots & \cdots & \begin{bmatrix} n \\ 2 \end{bmatrix} \\ & 1-w & \begin{bmatrix} 3 \\ 3 \end{bmatrix} & \cdots & \begin{bmatrix} n \\ 3 \end{bmatrix} \\ & & \ddots & & \vdots \\ & \bigcirc & & 1-w & \begin{bmatrix} n \\ n \end{bmatrix} \end{vmatrix}, \tag{B.19}$$

and $\psi_n^{(s)^{-1}}(M_n) = M_n^{(s)}$, where $\psi_n^{(s)^{-1}}$ replaces every occurrence of $\begin{bmatrix} i \\ j \end{bmatrix}$ according to the inverse mapping

$$\begin{bmatrix} i \\ j \end{bmatrix} \longmapsto \begin{bmatrix} s-n+i \\ s-n+j-1 \end{bmatrix}.$$

Now we can work with the simple determinant M_n and then recover $M_n^{(s)}$ later. We now find an expression for M_n using Lemma 2 in Appendix E.1. Recall we are trying to show that (B.2) = (B.11). So far, we can write (B.11) in terms of $M_n^{(s)}$ using (B.16) and (B.17). Since Lemma 2 gives us an expression for M_n, we will now express $M_n^{(s)} = \psi_n^{(s)^{-1}}(M_n)$ in terms of M_n. Looking at (E.1), we define

$$\tilde{\Sigma}_l^{s,n} = \psi_n^{(s)^{-1}}(\bar{\Sigma}_l^n) = \bar{\Sigma}_l^n \Big| \begin{bmatrix} i \\ j \end{bmatrix} \leftarrow \begin{pmatrix} s-n+i \\ s-n+j-1 \end{pmatrix}, \tag{B.20}$$

undoing the transformation and recovering the binomial coefficients from the ordered pairs. We can then write

$$M_n^{(s)} = \psi_n^{(s)^{-1}}(M_n) = \begin{cases} \sum_{k=0}^{n-1}(-1)^{n-k} w^k \sum_{l=1}^{n-k}(-1)^l \begin{pmatrix} n-l \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,n}, & n \geq 1, \\ 1, & n = 0. \end{cases} \tag{B.21}$$

The special case $M_0^{(s)}$ comes from (B.14). Now, using (B.20) and (E.2), we find that

$$\begin{aligned}
\tilde{\Sigma}_l^{s,n} &= \begin{cases} \sum_{1\le i_1<\cdots<i_{l-1}\le n-1} \binom{s-n+i_1}{s-n}\binom{s-n+i_2}{s-n+i_1}\cdots \\ \qquad \binom{s-n+i_{l-1}}{s-n+i_{l-2}}\binom{s}{s-n+i_{l-1}}, & l\ge 2 \\ \binom{s}{s-n}, & l=1 \end{cases} \\
&= \begin{cases} \sum_{s-n+1\le i_1<\cdots<i_{l-1}\le s-1} \binom{i_1}{s-n}\binom{i_2}{i_1}\cdots\binom{i_{l-1}}{i_{l-2}}\binom{s}{i_{l-1}}, & l\ge 2 \\ \binom{s}{n}, & l=1. \end{cases}
\end{aligned} \tag{B.22}$$

We now use (B.11) together with (B.1) by induction to get

$$\phi_r(w,z) = -a_r(z)\left[(z+1-w)D_1^{(s)} + (zw-z)D_2^{(s)}\right] - (1-w)\sum_{k=0}^{r-1} w^k \phi_k^{(r-1)}. \tag{B.23}$$

Our goal now is to show that (B.23) = (B.1). We do this by comparing the terms involving $a_i(z)$ in both expressions, for $i = 1,\ldots,r$, and also the terms independent of any $a_i(z)$. First, note that, since no $D_k^{(s)}$ contains any $a_i(z)$, the coefficient of $a_r(z)$ in (B.23) is

$$-\left[(z+1-w)D_1^{(s)} + (zw-z)D_2^{(s)}\right], \tag{B.24}$$

and the terms involving $a_i(z)$, $i = 1,\ldots,r-1$, must be

$$\begin{aligned}
& -(1-w)\sum_{k=0}^{r-1} w^k \phi_k^{(r-1)} \\
= \; & -\sum_{k=0}^{r-1} w^k \phi_k^{(r-1)} + \sum_{k=0}^{r-1} w^{k+1}\phi_k^{(r-1)} \\
= \; & -\sum_{k=0}^{r-1} w^k \phi_k^{(r-1)} + \sum_{k=1}^{r} w^k \phi_{k-1}^{(r-1)} \\
= \; & -\phi_0^{(r-1)} - \sum_{k=1}^{r-1} w^k(\phi_k^{(r-1)} - \phi_{k-1}^{(r-1)}) + w^r \phi_{r-1}^{(r-1)}.
\end{aligned} \tag{B.25}$$

Therefore, we must show that the coefficients of $a_i(z)$, $i = 1,\ldots,r-1$, as well as the term independent of any $a_i(z)$ in (B.25) and (B.1) agree. First, consider the term with no $a_i(z)$'s.

Using (B.2) and (B.3), we see that in (B.1) this term is

$$\begin{aligned}\sum_{k=0}^{r} w^k \delta_k^{(s)} &= \sum_{k=0}^{r} (-1)^{s+1-k} \left(\begin{array}{c} s+1 \\ k \end{array} \right) w^k \\ &= (-1)^{s+1} \sum_{k=0}^{s+1} (-1)^k \left(\begin{array}{c} s+1 \\ k \end{array} \right) w^k \end{aligned} \tag{B.26}$$

Using (B.2) and (B.3) in (B.25), we see that the term with no $a_i(z)$'s is

$$\begin{aligned} & -\left[\delta_0^{(s-1)} + \sum_{k=1}^{r-1} w^k (\delta_k^{(s-1)} - \delta_{k-1}^{(s-1)}) - w^r \delta_{r-1}^{(s-1)} \right] \\ = & -(-1)^s \left[(-1)^0 \left(\begin{array}{c} s \\ 0 \end{array} \right) + \sum_{k=1}^{s} (-1)^k \left(\left(\begin{array}{c} s \\ k \end{array} \right) + \left(\begin{array}{c} s \\ k-1 \end{array} \right) \right) w^k - (-1)^0 \left(\begin{array}{c} s \\ s \end{array} \right) w^r \right] \\ = & (-1)^{s+1} \left[\left(\begin{array}{c} s+1 \\ 0 \end{array} \right) + \sum_{k=1}^{s} (-1)^k \left(\begin{array}{c} s+1 \\ k \end{array} \right) w^k - \left(\begin{array}{c} s+1 \\ s+1 \end{array} \right) w^r \right] \\ = & (-1)^{s+1} \sum_{k=0}^{s+1} (-1)^k \left(\begin{array}{c} s+1 \\ k \end{array} \right) w^k = \text{(B.26)}, \end{aligned}$$

as required. Now let $i = 1, \ldots, r-1$ and consider the coefficients of $w^k a_i(z)$ in (B.25) and (B.1). In (B.1), we have for $k = 0, \ldots, r-1$

$$\alpha_{k,i}^{(s)} + \beta_{k,i}^{(s)} z, \tag{B.27}$$

and for $k = r$ the coefficient of $w^k a_i(z)$ is 0. In (B.25) there are 3 cases: First, when $k = 0$, the coefficient of $w^k a_i(z)$ in (B.25) is

$$-\left(\alpha_{0,i}^{(s-1)} + \beta_{0,i}^{(s-1)} z \right), \tag{B.28}$$

in which case we need to show

$$\alpha_{0,i}^{(s)} = -\alpha_{0,i}^{(s-1)}, \tag{B.29}$$

$$\beta_{0,i}^{(s)} = -\beta_{0,i}^{(s-1)}, \tag{B.30}$$

for $i = 1, \ldots, r-1$. Second, when $k = 1, \ldots, r-1$, the coefficient of $w^k a_i(z)$ in (B.25) is

$$-\left[\alpha_{k,i}^{(s-1)} + \beta_{k,i}^{(s-1)} z - \left(\alpha_{k-1,i}^{(s-1)} + \beta_{k-1,i}^{(s-1)} z \right) \right], \tag{B.31}$$

in which case we need to show

$$\alpha_{k,i}^{(s)} = -\alpha_{k,i}^{(s-1)} + \alpha_{k-1,i}^{(s-1)}, \tag{B.32}$$

$$\beta_{k,i}^{(s)} = -\beta_{k,i}^{(s-1)} + \beta_{k-1,i}^{(s-1)}, \tag{B.33}$$

for $k = 1, \ldots, r-1$ and $i = 1, \ldots, r-1$. Third, when $k = r$, the coefficient of $w^k a_i(z)$ in (B.25) is 0 as required, since $\phi_{r-1}^{(r-1)} = 1$ by (B.2). Consider the second equation (B.30) of the first case. Using (B.5), we have, for $i = 1$,

$$LS = (-1)^{s+1} \begin{pmatrix} s \\ 0 \end{pmatrix} = (-1)^{s+1} = -(-1)^s \begin{pmatrix} s-1 \\ 0 \end{pmatrix} = RS.$$

For $i > 1$,

$$\begin{aligned} LS &= (-1)^s \left[\begin{pmatrix} s-1 \\ 0 \end{pmatrix} + \sum_{l=1}^{s-1} (-1)^l \begin{pmatrix} s-l-1 \\ 0 \end{pmatrix} \Sigma_l^{i-1} \right] \\ &= (-1)^s \left[1 + \sum_{l=1}^{s-1} (-1)^l \Sigma_l^{i-1} \right], \\ RS &= -(-1)^{s-1} \left[\begin{pmatrix} s-2 \\ 0 \end{pmatrix} + \sum_{l=1}^{s-2} (-1)^l \begin{pmatrix} s-l-2 \\ 0 \end{pmatrix} \Sigma_l^{i-1} \right] \\ &= (-1)^s \left[1 + \sum_{l=1}^{s-2} (-1)^l \Sigma_l^{i-1} \right] \end{aligned}$$

and $LS = RS$ because, by (B.6), $i \leq r - 1 = s$ implies $\Sigma_{s-1}^{i-1} = 0$.

For (B.29), when $i = 1$, $LS = RS = 0$ by (B.4). When $i > 1$, using (B.4) together with (B.30), which we already proved, we get

$$LS = -(i-1)\beta_{0,i-1}^{(s)} = (i-1)\beta_{0,i-1}^{(s-1)} = RS.$$

Now consider the second equation (B.33) of the second case. Using (B.5), we have, for $i = 1$,

$$\begin{aligned} LS &= (-1)^{s+1-k} \begin{pmatrix} s \\ k \end{pmatrix}, \\ RS &= -\left[(-1)^{s-k} \begin{pmatrix} s-1 \\ k \end{pmatrix} - (-1)^{s-k+1} \begin{pmatrix} s-1 \\ k-1 \end{pmatrix} \right] \end{aligned}$$

$$
\begin{aligned}
&= (-1)^{s+1-k}\left[\begin{pmatrix} s-1 \\ k \end{pmatrix} + \begin{pmatrix} s-1 \\ k-1 \end{pmatrix}\right] \\
&= (-1)^{s+1-k}\begin{pmatrix} s \\ k \end{pmatrix} = LS.
\end{aligned}
$$

For $i > 1$,

$$
\begin{aligned}
LS &= (-1)^{s-k}\left[\begin{pmatrix} s-1 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-1}(-1)^l \begin{pmatrix} s-l-1 \\ k \end{pmatrix}\Sigma_l^{i-1}\right] \\
RS &= -\left\{(-1)^{s-1-k}\left[\begin{pmatrix} s-2 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-2}(-1)^l \begin{pmatrix} s-l-2 \\ k \end{pmatrix}\Sigma_l^{i-1}\right]\right. \\
&\quad \left. -(-1)^{s-k}\left[\begin{pmatrix} s-2 \\ k-1 \end{pmatrix} + \sum_{l=1}^{s-k-1}(-1)^l \begin{pmatrix} s-l-2 \\ k-1 \end{pmatrix}\Sigma_l^{i-1}\right]\right\} \\
&= (-1)^{s-k}\left\{\begin{pmatrix} s-2 \\ k \end{pmatrix} + \begin{pmatrix} s-2 \\ k-1 \end{pmatrix} + \sum_{l=1}^{s-k-2}(-1)^l \left[\begin{pmatrix} s-l-2 \\ k \end{pmatrix}\right.\right. \\
&\quad \left.\left. + \begin{pmatrix} s-l-2 \\ k-1 \end{pmatrix}\right]\Sigma_l^{i-1} + (-1)^{s-k-1}\begin{pmatrix} s-(s-k-1)-2 \\ k-1 \end{pmatrix}\Sigma_{s-k-1}^{i-1}\right\} \\
&= (-1)^{s-k}\left\{\begin{pmatrix} s-1 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-2}(-1)^l \begin{pmatrix} s-l-1 \\ k \end{pmatrix}\Sigma_l^{i-1}\right. \\
&\quad \left. +(-1)^{s-k-1}\begin{pmatrix} k-1 \\ k-1 \end{pmatrix}\Sigma_{s-k-1}^{i-1}\right\} \\
&= (-1)^{s-k}\left\{\begin{pmatrix} s-1 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-1}(-1)^l \begin{pmatrix} s-l-1 \\ k \end{pmatrix}\Sigma_l^{i-1}\right\} \\
&= LS.
\end{aligned}
$$

For (B.32), when $i = 1$, $LS = RS = 0$. When $i > 1$, using (B.4) together with (B.33), which we already proved, we get

$$
LS = -(i-1)\beta_{k,i-1}^{(s)} = -\left[-(i-1)\beta_{k,i-1}^{(s-1)} + (i-1)\beta_{k-1,i-1}^{(s-1)}\right] = RS.
$$

So we have established the agreement of all the terms in (B.23) and (B.1) except the $a_r(z)$ term, which we consider now.

The coefficient of $a_r(z)$ in (B.1) is

$$\sum_{k=0}^{r} w^k \left(\alpha_{k,r}^{(s)} + \beta_{k,r}^{(s)} z\right), \tag{B.34}$$

while that in (B.23) is

$$-\left[(z+1-w)D_1^{(s)} + (zw-z)D_2^{(s)}\right] = -\left[z\left(D_1^{(s)} + (w-1)D_2^{(s)}\right) + (1-w)D_1^{(s)}\right].$$

Thus we must show that

$$\sum_{k=0}^{r} w^k \alpha_{k,r}^{(s)} = -(1-w)D_1^{(s)}, \tag{B.35}$$

and

$$\sum_{k=0}^{r} w^k \beta_{k,r}^{(s)} = -\left[D_1^{(s)} + (w-1)D_2^{(s)}\right]. \tag{B.36}$$

Note first that the left sides of (B.35) and (B.36) have no terms with $k > s-1 = r-2$. This is because, from (B.5),

$$\beta_{k,r}^{(s)} = (-1)^{s-k}\left[\begin{pmatrix} s-1 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-1} \cdots\right],$$

which is zero for $k > s-1$. It then follows from (B.4) that

$$\alpha_{k,r}^{(s)} = -(r-1)\beta_{k,r-1}^{(s)} = -(r-1)(-1)^{s-k}\left[\begin{pmatrix} s-2 \\ k \end{pmatrix} + \sum_{l=1}^{s-k-2} \cdots\right],$$

which is zero for $k > s-2$. Thus we can replace (B.35) and (B.36) by

$$\sum_{k=0}^{r-2} w^k \alpha_{k,r}^{(s)} = -(1-w)D_1^{(s)}, \tag{B.37}$$

and

$$\sum_{k=0}^{r-2} w^k \beta_{k,r}^{(s)} = -\left[D_1^{(s)} + (w-1)D_2^{(s)}\right]. \tag{B.38}$$

From (B.21) we have

$$M_n^{(s)} = \begin{cases} \sum_{k=0}^{n-1}(-1)^{n-k} w^k T_{n,k}^{(s)}, & n \geq 1, \\ 1, & n = 0, \end{cases} \tag{B.39}$$

where

$$T_{n,k}^{(s)} = \sum_{l=1}^{n-k} (-1)^l \begin{pmatrix} n-l \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,n}. \tag{B.40}$$

Using this in (B.16) and (B.17), we get

$$D_1^{(s)} = \sum_{n=1}^{s-2} (n+1)(1-w)^{n-1} \sum_{k=0}^{s-n-2} (-1)^{s-k} w^k T_{s-n-1,k}^{(s)} + (-1)^s s(1-w)^{s-2}, \tag{B.41}$$

$$D_2^{(s)} = \sum_{n=1}^{s-2} (1-w)^{n-1} \sum_{k=0}^{s-n-2} (-1)^{s-k} w^k T_{s-n-1,k}^{(s)} + (-1)^s (1-w)^{s-2}. \tag{B.42}$$

Looking at (B.41) and (B.42), we see that the highest power of w in $D_1^{(s)}$ and $D_2^{(s)}$ is $s-2$. So

$$D_1^{(s)} = \sum_{k=0}^{s-2} t_k w^k \tag{B.43}$$

$$D_2^{(s)} = \sum_{k=0}^{s-2} u_k w^k \tag{B.44}$$

for suitable t_k's and u_k's. Then (B.37) and (B.38) reduce to

$$\alpha_{k,r}^{(s)} = -t_k + t_{k-1}, \quad k = 0, \ldots, r-2, \tag{B.45}$$

$$\beta_{k,r}^{(s)} = -t_k - u_{k-1} + u_k, \quad k = 0, \ldots, r-2, \tag{B.46}$$

where

$$t_{-1} = u_{-1} = t_{r-2} = u_{r-2} = 0. \tag{B.47}$$

All that remains is to prove (B.45) and (B.46) subject to (B.47).

To this end, we now find expressions for the t_k's and u_k's. Consider the expression (B.42) for $D_2^{(s)}$ first. The argument of the n-sum is

$$\begin{aligned} & (1-w)^{n-1} \sum_{k=0}^{s-n-2} (-1)^{s-k} w^k T_{s-n-1,k}^{(s)} \\ = & \sum_{l=0}^{n-1} (-1)^l \begin{pmatrix} n-1 \\ l \end{pmatrix} w^l \sum_{k=0}^{s-n-2} (-1)^{s-k} w^k T_{s-n-1,k}^{(s)} \\ = & (-1)^s \sum_{l=0}^{n-1} \sum_{k=0}^{s-n-2} (-1)^{l+k} \begin{pmatrix} n-1 \\ l \end{pmatrix} T_{s-n-1,k}^{(s)} w^{l+k} \end{aligned}$$

$$= \quad (-1)^s \sum_{k=0}^{s-n-2} \sum_{l=0}^{n-1} \alpha(k,l) w^{l+k}, \tag{B.48}$$

where

$$\alpha(k,l) = (-1)^{l+k} \left(\begin{array}{c} n-1 \\ l \end{array} \right) T^{(s)}_{s-n-1,k}.$$

We want to express (B.48) as powers of w. Consider the general case

$$\sum_{k=0}^{a} \sum_{l=0}^{b} \alpha(k,l) w^{k+l}.$$

Let $p = k + l$, so that $l = p - k$. Then $0 \leq p - k \leq b \Longrightarrow p - b \leq k \leq p$, so that

$$\sum_{k=0}^{a} \sum_{l=0}^{b} \alpha(k,l) w^{k+l} = \sum_{p=0}^{a+b} \sum_{k=\max\{0,p-b\}}^{\min\{a,p\}} \alpha(k,p-k) w^p. \tag{B.49}$$

In (B.48), $a = s-n-2$, $b = n-1$, and $a+b = s-3$. Therefore

$$\text{(B.48)} = (-1)^s \sum_{p=0}^{s-3} \sum_{k=\max\{0,p-n+1\}}^{\min\{s-n-2,p\}} (-1)^p w^p \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k},$$

and so from (B.42),

$$\begin{aligned} D_2^{(s)} &= (-1)^s \sum_{n=1}^{s-2} \sum_{p=0}^{s-3} (-1)^p w^p \sum_{k=\max\{0,p-n+1\}}^{\min\{s-n-2,p\}} \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k} \\ &\quad + (-1)^s \sum_{p=0}^{s-2} (-1)^p \left(\begin{array}{c} s-2 \\ p \end{array} \right) w^p \\ &= (-1)^s \left\{ \sum_{p=0}^{s-3} (-1)^p w^p \left[\sum_{n=1}^{s-2} \sum_{k=\max\{0,p-n+1\}}^{\min\{s-n-2,p\}} \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k} + \left(\begin{array}{c} s-2 \\ p \end{array} \right) \right] \right. \\ &\quad \left. + (-1)^s w^{s-2} \right\}. \end{aligned}$$

Thus we have

$$u_p = \begin{cases} (-1)^{s+p} \left[\sum_{n=1}^{s-2} \sum_{k=\max\{0,p-n+1\}}^{\min\{s-n-2,p\}} \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k} + \left(\begin{array}{c} s-2 \\ p \end{array} \right) \right], & p \leq s-3, \\ 1, & p = s-2. \end{cases} \tag{B.50}$$

Similarly, for $D_1^{(s)}$,

$$t_p = \begin{cases} (-1)^{s+p}\left[\sum_{n=1}^{s-2}(n+1)\sum_{k=\max\{0,p-n+1\}}^{\min\{s-n-2,p\}}\left(\begin{array}{c} n-1 \\ p-k \end{array}\right) T_{s-n-1,k}^{(s)} \right. \\ \qquad\qquad \left. + \left(\begin{array}{c} s-2 \\ p \end{array}\right) s\right], & p \le s-3, \\ s, & p = s-2. \end{cases} \tag{B.51}$$

Now we use (B.50) and (B.51) to prove (B.45) and (B.46). Consider (B.46) first. There are 3 cases: $k = 0$, $k = 1, \ldots, r-3$, and $k = r-2$. If $k = 0$, (B.46) requires

$$\beta_{0,r}^{(s)} = -t_0 + u_0. \tag{B.52}$$

If $s = 2$, the right side is $-s + 1 = -1$. From (B.5), the left side is

$$(-1)^2\left[\left(\begin{array}{c} 1 \\ 0 \end{array}\right) + (-1)^1\left(\begin{array}{c} 0 \\ 0 \end{array}\right)\Sigma_1^2\right] = 1 - \Sigma_1^2 = -1$$

by (B.6), as required. When $s \ge 3$, we see from (B.50), (B.51) and (B.40) that the right side of (B.52) is

$$\begin{aligned} -t_0 + u_0 &= -(-1)^s\left[\sum_{n=1}^{s-2} n \sum_{k=\max\{0,-n+1\}}^{\min\{s-n-2,0\}}\left(\begin{array}{c} n-1 \\ -k \end{array}\right) T_{s-n-1,k}^{(s)} + \left(\begin{array}{c} s-2 \\ 0 \end{array}\right)(s-1)\right] \\ &= -(-1)^s\left[\sum_{n=1}^{s-2} n T_{s-n-1,0}^{(s)} + (s-1)\right] \\ &= -(-1)^s\left[\sum_{n=1}^{s-2} n \sum_{l=1}^{s-n-1}(-1)^l\left(\begin{array}{c} s-n-1-l \\ 0 \end{array}\right)\tilde{\Sigma}_l^{s,s-n-1} + (s-1)\right] \\ &= -(-1)^s\left[\sum_{n=1}^{s-2} n \sum_{l=1}^{s-n-1}(-1)^l\tilde{\Sigma}_l^{s,s-n-1} + (s-1)\right]. \end{aligned}$$

Using (B.5), we see that the left side is

$$\begin{aligned} \beta_{0,r}^{(s)} &= (-1)^s\left[\left(\begin{array}{c} s-1 \\ 0 \end{array}\right) + \sum_{l=1}^{s-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ 0 \end{array}\right)\Sigma_l^{r-1}\right] \\ &= (-1)^s\left[1 + \sum_{l=1}^{s-1}(-1)^l\Sigma_l^{r-1}\right]. \end{aligned}$$

Therefore, (B.52) is equivalent to

$$\sum_{l=1}^{s-1}(-1)^l\Sigma_l^s = -s - \sum_{n=1}^{s-2} n \sum_{l=1}^{s-n-1}(-1)^l\tilde{\Sigma}_l^{s,s-n-1}$$

which in turn is equivalent to

$$\sum_{l=1}^{s-1}(-1)^l\Sigma_l^s = -s - \sum_{n=1}^{s-2}(s-n-1)\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n}. \tag{B.53}$$

Since the left side of (B.53) involves Σ_l^s and the right side involves $\tilde{\Sigma}_l^{s,n}$, we need to find a relationship between these two quantities. From (B.6),

$$\begin{aligned}
\Sigma_l^s &= \sum_{1\le i_1<\cdots<i_l\le s-1}\binom{i_2}{i_1}\binom{i_3}{i_2}\cdots\binom{i_l}{i_{l-1}}\binom{s}{i_l} \\
&= \sum_{i_1=1}^{s-l}\sum_{i_1<i_2<\cdots<i_l\le s-1}\binom{i_2}{i_1}\binom{i_3}{i_2}\cdots\binom{i_l}{i_{l-1}}\binom{s}{i_l} \\
&= \sum_{i=1}^{s-l}\sum_{i+1\le i_2<\cdots<i_l\le s-1}\binom{i_2}{i_1}\binom{i_3}{i_2}\cdots\binom{i_l}{i_{l-1}}\binom{s}{i_l} \\
&= \sum_{i=1}^{s-l}\tilde{\Sigma}_l^{s,s-i},
\end{aligned}$$

by (B.20). Thus

$$\Sigma_l^s = \sum_{n=1}^{s-l}\tilde{\Sigma}_l^{s,s-n}. \tag{B.54}$$

Using this in (B.53) reduces what we need to prove to

$$\sum_{l=1}^{s-1}(-1)^l\sum_{n=1}^{s-l}\tilde{\Sigma}_l^{s,s-n} = -s - \sum_{n=1}^{s-2}(s-n-1)\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n}. \tag{B.55}$$

Now switch the order of the sums on the left side of (B.55) to agree with those on the right side. When $l=1$, $n=1,\ldots,s-1$. When $l=s-1$, $n=1$. Thus, $l=1,\ldots,s-n$ and $n=1,\ldots,s-1$. So (B.55) is equivalent to

$$\begin{aligned}
&\sum_{n=1}^{s-1}\sum_{l=1}^{s-n}(-1)^l\tilde{\Sigma}_l^{s,s-n} = -s - \sum_{n=1}^{s-2}(s-n-1)\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n} \\
\iff\ &\sum_{n=1}^{s-1}\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n} = -s - \sum_{n=1}^{s-2}(s-n-1)\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n}
\end{aligned}$$

$$\iff \sum_{n=1}^{s-2}\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n} + \sum_{l=1}^{s-1}(-1)^l\tilde{\Sigma}_l^{s,s-1} = -s - \sum_{n=1}^{s-2}(s-n-1)\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n}$$

$$\iff \sum_{n=1}^{s-2}(s-n)\sum_{l=1}^{n}(-1)^l\tilde{\Sigma}_l^{s,n} + \sum_{l=1}^{s-1}(-1)^l\tilde{\Sigma}_l^{s,s-1} + s = 0 \tag{B.56}$$

Using Lemma 3 in Appendix E.2, we reduce the left side of (B.56) to

$$\begin{aligned}
&\sum_{n=1}^{s-2}(s-n)(-1)^n\begin{pmatrix} s \\ n \end{pmatrix} + (-1)^{s-1}\begin{pmatrix} s \\ s-1 \end{pmatrix} + s \\
= \;& s\sum_{n=1}^{s-2}(-1)^n\begin{pmatrix} s \\ n \end{pmatrix} - \sum_{n=1}^{s-2}(-1)^n n\begin{pmatrix} s \\ n \end{pmatrix} + (-1)^{s-1}s + s \\
= \;& s\left[\sum_{n=0}^{s}(-1)^n\begin{pmatrix} s \\ n \end{pmatrix} - (-1)^0\begin{pmatrix} s \\ 0 \end{pmatrix} - (-1)^{s-1}\begin{pmatrix} s \\ s-1 \end{pmatrix} - (-1)^s\begin{pmatrix} s \\ s \end{pmatrix}\right] \\
& - \left[\sum_{n=0}^{s}(-1)^n n\begin{pmatrix} s \\ n \end{pmatrix} - (-1)^0 0\begin{pmatrix} s \\ 0 \end{pmatrix} - (-1)^{s-1}(s-1)\begin{pmatrix} s \\ s-1 \end{pmatrix}\right. \\
& \qquad \left. -(-1)^s s\begin{pmatrix} s \\ s \end{pmatrix}\right] + (-1)^{s-1}s + s \\
= \;& s\left[0 - 1 + (-1)^s s - (-1)^s\right] - \left[0 - 0 + (-1)^s(s-1)s - (-1)^s s\right] - (-1)^s s + s = 0,
\end{aligned}$$

since $s \geq 2$ implies

$$\sum_{n=0}^{s}(-1)^n\begin{pmatrix} s \\ n \end{pmatrix} = 0$$

by (F.3) and

$$\sum_{n=0}^{s}(-1)^n n\begin{pmatrix} s \\ n \end{pmatrix} = 0$$

by (F.1). This completes the proof of (B.52), the $k = 0$ case of (B.46).

Now consider $k = r - 2 = s - 1$, in which case (B.46) becomes

$$\beta_{s-1,r}^{(s)} = -t_{s-1} - u_{s-2} + u_{s-1}. \tag{B.57}$$

Using (B.50) and (B.47), we see that the right side of (B.57) equals $0 - 1 + 0 = -1$, while, from (B.5), the left side equals $-(1 + 0) = -1$ also. Therefore, (B.57) holds, as required.

All that remains is to show that (B.46) holds for $k = 1, \ldots, r - 3 = s - 2$, which is equivalent to showing that

$$\beta_{k,r}^{(s)} = -t_k - u_{k-1} + u_k,$$

or, on substituting p for k,

$$\beta_{p,r}^{(s)} = -t_p - u_{p-1} + u_p, \quad \text{for } p = 1, \ldots, s-2. \tag{B.58}$$

First consider $p = s-2$. Then, using (B.51), (B.50) and (B.40), we have

$$\begin{aligned} -t_p - u_{p-1} + u_p &= -s - u_{s-3} + 1 \\ &= 1 - s + \left[\sum_{n=1}^{s-2} \begin{pmatrix} n-1 \\ n-1 \end{pmatrix} T_{s-n-1,s-n-2}^{(s)} + \begin{pmatrix} s-2 \\ s-3 \end{pmatrix} \right] \\ &= -1 + \sum_{n=1}^{s-2} T_{s-n-1,s-n-2}^{(s)} \\ &= -1 - \sum_{n=1}^{s-2} \begin{pmatrix} s-n-2 \\ s-n-2 \end{pmatrix} \tilde{\Sigma}_1^{s,s-n-1} \\ &= -1 - \sum_{n=1}^{s-2} \tilde{\Sigma}_1^{s,n}, \end{aligned} \tag{B.59}$$

where (B.59) follows from the line above it by reversing the order of the sum. We now apply (B.54) with $l = 1$ and (B.22) to get

$$\sum_{n=1}^{s-2} \tilde{\Sigma}_1^{s,n} = \Sigma_1^s - \tilde{\Sigma}_1^{s,s-1} = \Sigma_1^s - \begin{pmatrix} s \\ s-1 \end{pmatrix} = \Sigma_1^s - s.$$

Using this in (B.59) gives $-1 - (\Sigma_1^s - s)$ for the right side of (B.58). From (B.5), the left side of (B.58) is

$$\begin{aligned} \beta_{s-2,r}^{(s)} &= \begin{pmatrix} s-1 \\ s-2 \end{pmatrix} - \begin{pmatrix} s-2 \\ s-2 \end{pmatrix} \Sigma_1^{r-1} \\ &= s - 1 - \Sigma_1^s, \end{aligned}$$

as required.

It remains to show (B.58) for $p = 1, \ldots, s-3$. The next large portion of the proof is devoted to this. For the right side of (B.58), we have

$$u_p - t_p - u_{p-1} = (-1)^{s+p} \left[\sum_{n=1}^{s-2} (-n) \sum_{k=\max\{0,p-n+1\}}^{\min\{s-n-2,p\}} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} T_{s-n-1,k}^{(s)} + \begin{pmatrix} s-2 \\ p \end{pmatrix} (1-s) \right]$$

$$-(-1)^{s+p-1}\left[\sum_{n=1}^{s-2}\sum_{k=\max\{0,p-n\}}^{\min\{s-n-2,p-1\}}\left(\begin{array}{c} n-1 \\ p-k-1 \end{array}\right) T^{(s)}_{s-n-1,k} + \left(\begin{array}{c} s-2 \\ p-1 \end{array}\right)\right]. \tag{B.60}$$

We handle the min and max occurring in the sums by splitting the range of n as follows.

$$1 \le n \le p \implies \begin{cases} \max\{0, p-n\} = p-n, \\ \max\{0, p-n+1\} = p-n+1, \end{cases} \tag{B.61}$$

$$p+1 \le n \le s-2 \implies \begin{cases} \max\{0, p-n\} = 0, \\ \max\{0, p-n+1\} = 0, \end{cases} \tag{B.62}$$

$$1 \le n \le s-p-2 \implies \begin{cases} \min\{s-n-2, p\} = p, \\ \min\{s-n-2, p-1\} = p-1, \end{cases} \tag{B.63}$$

$$s-p-1 \le n \le s-2 \implies \begin{cases} \min\{s-n-2, p\} = s-n-2, \\ \min\{s-n-2, p-1\} = s-n-2. \end{cases} \tag{B.64}$$

Are any of these ranges incompatible?

$$\text{(B.61) and (B.63)} \implies \begin{cases} p \ge 1, \\ s-p-2 \ge 1 \end{cases} \implies p \le s-3 \quad \text{(which is always true)},$$

$$\text{(B.61) and (B.64)} \implies s-p-1 \le p \implies p \ge \frac{s-1}{2},$$

$$\text{(B.62) and (B.63)} \implies p+1 \le s-p-2 \implies p \le \frac{s-3}{2},$$

$$\text{(B.62) and (B.64)} \implies \begin{cases} p+1 \le s-2, \\ s-p-1 \le s-2 \end{cases} \implies p \le s-3 \quad \text{(which is always true)}.$$

Thus, when $p \le (s-3)/2$, we have (B.61) and (B.63), (B.62) and (B.63), and (B.62) and (B.64); and when $p \ge (s-3)/2$, we have (B.61) and (B.63), (B.61) and (B.64), and (B.62) and (B.64).

First, assume

$$1 \le p \le \frac{s-3}{2}. \tag{B.65}$$

In this case, (B.60) becomes

$$(-1)^{s+p}\left[\left(\begin{array}{c} s-2 \\ p \end{array}\right)(1-s) + \left(\begin{array}{c} s-2 \\ p-1 \end{array}\right)\right]$$

$$+ \quad (-1)^{s+p} \left\{ \sum_{n=1}^{p} \left[-n \sum_{k=p-n+1}^{p} \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k} + \sum_{k=p-n}^{p-1} \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) T^{(s)}_{s-n-1,k} \right] \right.$$
$$+ \quad \sum_{n=p+1}^{s-p-2} \left[-n \sum_{k=0}^{p} \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k} + \sum_{k=0}^{p-1} \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) T^{(s)}_{s-n-1,k} \right]$$
$$+ \quad \left. \sum_{n=s-p-1}^{s-2} \left[-n \sum_{k=0}^{s-n-2} \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) T^{(s)}_{s-n-1,k} + \sum_{k=0}^{s-n-2} \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) T^{(s)}_{s-n-1,k} \right] \right\}$$

By (B.40), this is equivalent to

$$(-1)^{s+p} \left\{ \left(\begin{array}{c} s-2 \\ p \end{array} \right) (1-s) + \left(\begin{array}{c} s-2 \\ p-1 \end{array} \right) \tag{B.66}$$
$$- \quad \sum_{n=1}^{p} \sum_{k=p-n+1}^{p} \sum_{l=1}^{s-n-k-1} (-1)^l n \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \tilde{\Sigma}_l^{s,s-n-1} \tag{B.67}$$
$$+ \quad \sum_{n=1}^{p} \sum_{k=p-n}^{p-1} \sum_{l=1}^{s-n-k-1} (-1)^l \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \tilde{\Sigma}_l^{s,s-n-1} \tag{B.68}$$
$$- \quad \sum_{n=p+1}^{s-p-2} \sum_{k=0}^{p} \sum_{l=1}^{s-n-k-1} (-1)^l n \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \tilde{\Sigma}_l^{s,s-n-1} \tag{B.69}$$
$$+ \quad \sum_{n=p+1}^{s-p-2} \sum_{k=0}^{p-1} \sum_{l=1}^{s-n-k-1} (-1)^l \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \tilde{\Sigma}_l^{s,s-n-1} \tag{B.70}$$
$$- \quad \sum_{n=s-p-1}^{s-2} \sum_{k=0}^{s-n-2} \sum_{l=1}^{s-n-k-1} (-1)^l n \left(\begin{array}{c} n-1 \\ p-k \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \tilde{\Sigma}_l^{s,s-n-1} \tag{B.71}$$
$$+ \quad \sum_{n=s-p-1}^{s-2} \sum_{k=0}^{s-n-2} \sum_{l=1}^{s-n-k-1} (-1)^l \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \tilde{\Sigma}_l^{s,s-n-1} \Bigg\} \tag{B.72}$$

From (B.5), the left side of (B.58) is

$$\beta^{(s)}_{p,r} = (-1)^{s+p} \left[\left(\begin{array}{c} s-1 \\ p \end{array} \right) + \sum_{l=1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-1 \\ p \end{array} \right) \Sigma^s_l \right] \tag{B.73}$$

since $r \geq 3$. To show the left and right sides of (B.58) are indeed equal, we will switch the order of the sums in (B.66) to (B.72) from n, k, l to l, k, n and then evaluate the k sum, since $\tilde{\Sigma}_l^{s,s-n-1}$ is independent of k. We first derive some general formulas for switching sums and then use them in specific instances.

First, consider the sums

$$\sum_{k=a}^{b}\sum_{l=1}^{c-k},$$

where a, b, c are free. When $k = a$, $l = 1, \ldots, c-a$. When $k = b$, $l = 1, \ldots, c-b$. Thus, for $l = 1, \ldots, c-b$, $k = a, \ldots, b$, and for $l = c-b, \ldots, c-a$, $k = a, \ldots, c-l$. Therefore,

$$\sum_{k=a}^{b}\sum_{l=1}^{c-k} = \sum_{l=1}^{c-b}\sum_{k=a}^{b} + \sum_{l=c-b+1}^{c-a}\sum_{k=a}^{c-l}. \tag{B.74}$$

Next, consider the sums

$$\sum_{n=1}^{p}\sum_{l=a-n}^{b},$$

where a, b, p are free. When $n = 1$, $l = a-1, \ldots, b$. When $n = p$, $l = a-p, \ldots, b$. Thus, for $l = a-1, \ldots, b$, $n = 1, \ldots, p$, and for $l = a-p, \ldots, a-1$, $n = a-l, \ldots, p$. Therefore,

$$\sum_{n=1}^{p}\sum_{l=a-n}^{b} = \sum_{l=a-p}^{a-1}\sum_{n=a-l}^{p} + \sum_{l=a}^{b}\sum_{n=1}^{p}. \tag{B.75}$$

Now consider the sums

$$\sum_{n=a}^{b}\sum_{l=c-n}^{d},$$

where a, b, c, d are free. When $n = a$, $l = c-a, \ldots, d$. When $n = b$, $l = c-b, \ldots, d$. Thus, for $l = c-a, \ldots, d$, $n = a, \ldots, b$, and for $l = c-b, \ldots, c-a$, $n = c-l, \ldots, b$. Therefore,

$$\sum_{n=a}^{b}\sum_{l=c-n}^{d} = \sum_{l=c-b}^{c-a}\sum_{n=c-l}^{b} + \sum_{l=c-a+1}^{d}\sum_{n=a}^{b}. \tag{B.76}$$

Finally, consider the sums

$$\sum_{n=a}^{b}\sum_{l=c-n}^{d-n},$$

where a, b, c, d are free. When $n = a$, $l = c-a, \ldots, d-a$. When $n = b$, $l = c-b, \ldots, d-b$. Here we need to consider 2 cases. First, if $c-a \leq d-b$, for $l = c-b, \ldots, c-a$, $n = c-l, \ldots, b$, for $l = c-a, \ldots, d-b$, $n = a, \ldots, b$, and for $l = d-b, \ldots, d-a$, $n = a, \ldots, d-l$. Second, if $d-b \leq c-a$, for $l = c-b, \ldots, d-b$, $n = c-l, \ldots, b$, for $l = d-b, \ldots, c-a$, $n = c-l, \ldots, d-l$, and for $l = c-a, \ldots, d-a$, $n = a, \ldots, d-l$.

Therefore, when $c-a \leq d-b$,

$$\sum_{n=a}^{b}\sum_{l=c-n}^{d-n} = \sum_{l=c-b}^{c-a}\sum_{n=c-l}^{b} + \sum_{l=c-a+1}^{d-b}\sum_{n=a}^{b} + \sum_{l=d-b+1}^{d-a}\sum_{n=a}^{d-l}, \tag{B.77}$$

and when $d-b \leq c-a$,

$$\sum_{n=a}^{b}\sum_{l=c-n}^{d-n} = \sum_{l=c-b}^{d-b}\sum_{n=c-l}^{b} + \sum_{l=d-b+1}^{c-a}\sum_{n=c-l}^{d-l} + \sum_{l=c-a+1}^{d-a}\sum_{n=a}^{d-l}. \tag{B.78}$$

This completes the general sum-reversal formulas. We now apply them.

It is convenient to work without the sum arguments, when possible, to avoid unnecessary clutter. Applying (B.74) to the k and l sums in (B.67) gives $a = p-n-1, b = p$, $c = s-n-1$, $c-b = s-n-p-1$ and $c-a = s-p-2$, so that

$$\begin{aligned} \text{(B.67)} &= -\sum_{n=1}^{p}\sum_{k=p-n+1}^{p}\sum_{l=1}^{s-n-k-1} \\ &= -\sum_{n=1}^{p}\sum_{l=1}^{s-n-p-1}\sum_{k=p-n+1}^{p} - \sum_{n=1}^{p}\sum_{l=s-n-p}^{s-p-2}\sum_{k=p-n+1}^{s-n-l-1}. \end{aligned} \tag{B.79}$$

Applying (B.74) again to the first n and l sums in (B.79) gives $a = 1$, $b = p$, $c = s-p-1$, $c-b = s-2p-1$ and $c-a = s-p-2$. Applying (B.75) to the second n and l sums in (B.79) gives $a = s-p$ and $b = s-p-2$. Therefore,

$$\begin{aligned} \text{(B.67)} &= -\sum_{l=1}^{s-2p-1}\sum_{n=1}^{p}\sum_{k=p-n+1}^{p} - \sum_{l=s-2p}^{s-p-2}\sum_{n=1}^{s-p-l-1}\sum_{k=p-n+1}^{p} \\ &\quad -\sum_{l=s-2p}^{s-p-1}\sum_{n=s-l-p}^{p}\sum_{k=p-n+1}^{s-n-l-1} - \sum_{l=s-p}^{s-p-2}\sum_{n=1}^{p}\sum_{k=p-n+1}^{s-n-l-1}. \end{aligned} \tag{B.80}$$

(Note that the last l sum is empty). Recovering the arguments from (B.67), we now evaluate the k sums appearing in (B.80). The first one is

$$\begin{aligned} \sum_{k=p-n+1}^{p}\binom{n-1}{p-k}\binom{s-n-l-1}{k} &= \sum_{k=0}^{n-1}\binom{n-1}{n-1-k}\binom{s-n-l-1}{k+p-n+1} \\ &= \binom{n-1+s-n-l-1}{n-1+p-n+1} \end{aligned}$$

$$= \begin{pmatrix} s-l-2 \\ p \end{pmatrix}, \tag{B.81}$$

by (F.6). The second k sum is

$$\begin{aligned}
&\sum_{k=p-n+1}^{s-n-l-1} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \\
&= \sum_{k=0}^{s-p-l-2} \begin{pmatrix} n-1 \\ n-1-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k+p-n+1 \end{pmatrix} \\
&= \sum_{k=0}^{s-p-l-2} \begin{pmatrix} n-1 \\ k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k+p-n+1 \end{pmatrix} \\
&= \sum_{k=0}^{N} \begin{pmatrix} x \\ k \end{pmatrix} \begin{pmatrix} N+r \\ k+r \end{pmatrix} \quad \text{for } N = s-p-l-2,\ x = n-1,\ r = p-n+1 \\
&= \sum_{k=0}^{N} \begin{pmatrix} x \\ k \end{pmatrix} \begin{pmatrix} N+r \\ N-k \end{pmatrix} \\
&= \begin{pmatrix} x+N+r \\ N \end{pmatrix} = \begin{pmatrix} x+N+r \\ x+r \end{pmatrix} = \begin{pmatrix} s-l-2 \\ p \end{pmatrix},
\end{aligned} \tag{B.82}$$

by (F.4). Then, from (B.67), (B.80), (B.81) and (B.82), we have

$$\begin{aligned}
\text{(B.67)} \quad = \quad & -\sum_{l=1}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n \tilde{\Sigma}_l^{s,s-n-1} \\
& -\sum_{l=s-2p}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \\
& -\sum_{l=s-2p}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-l-p}^{p} n \tilde{\Sigma}_l^{s,s-n-1}.
\end{aligned} \tag{B.83}$$

Applying (B.74) to the k and l sums in (B.68) gives $a = p-n$, $b = p-1$, $c = s-n-1$, $c-b = s-p-n$ and $c-a = s-p-1$, so that

$$\begin{aligned}
\text{(B.68)} \quad &= \sum_{n=1}^{p} \sum_{k=p-n}^{p-1} \sum_{l=1}^{s-n-k-1} \\
&= \sum_{n=1}^{p} \sum_{l=1}^{s-p-n} \sum_{k=p-n}^{p-1} + \sum_{n=1}^{p} \sum_{l=s-p-n-1}^{s-p-1} \sum_{k=p-n}^{s-n-l-1}.
\end{aligned} \tag{B.84}$$

Applying (B.74) again to the first n and l sums in (B.84) gives $a = 1$, $b = p$, $c = s - p$, $c - b = s - 2p$ and $c - a = s - p - 1$. Applying (B.75) to the second n and l sums in (B.84) gives $a = s - p + 1$ and $b = s - p - 1$. Therefore,

$$\begin{aligned}\text{(B.68)} \;=\; & \sum_{l=1}^{s-2p}\sum_{n=1}^{p}\sum_{k=p-n}^{p-1} + \sum_{l=s-2p+1}^{s-p-1}\sum_{n=1}^{s-p-l}\sum_{k=p-n}^{p-1} \\ & + \sum_{l=s-2p+1}^{s-p}\sum_{n=s-l-p+1}^{p}\sum_{k=p-n}^{s-n-l-1} + \sum_{l=s-p+1}^{s-p-2}. \end{aligned} \tag{B.85}$$

(Note that the last sum is empty). Recovering the arguments from (B.68), we now evaluate the k sums appearing in (B.85). The first one is

$$\begin{aligned}\sum_{k=p-n}^{p-1}\left(\begin{array}{c} n-1 \\ p-k-1 \end{array}\right)\left(\begin{array}{c} s-n-l-1 \\ k \end{array}\right) &= \sum_{k=0}^{n-1}\left(\begin{array}{c} n-1 \\ n-k-1 \end{array}\right)\left(\begin{array}{c} s-n-l-1 \\ k+p-n \end{array}\right) \\ &= \left(\begin{array}{c} n-1+s-n-l-1 \\ n-1+p-n \end{array}\right) \\ &= \left(\begin{array}{c} s-l-2 \\ p-1 \end{array}\right), \end{aligned} \tag{B.86}$$

by (F.6). The second k sum is

$$\begin{aligned} & \sum_{k=p-n}^{s-n-l-1}\left(\begin{array}{c} n-1 \\ p-k-1 \end{array}\right)\left(\begin{array}{c} s-n-l-1 \\ k \end{array}\right) \\ = \; & \sum_{k=0}^{s-p-l-1}\left(\begin{array}{c} n-1 \\ n-1-k \end{array}\right)\left(\begin{array}{c} s-n-l-1 \\ k+p-n \end{array}\right) \\ = \; & \sum_{k=0}^{s-p-l-1}\left(\begin{array}{c} n-1 \\ k \end{array}\right)\left(\begin{array}{c} s-n-l-1 \\ k+p-n \end{array}\right) \\ = \; & \sum_{k=0}^{N}\left(\begin{array}{c} x \\ k \end{array}\right)\left(\begin{array}{c} N+r \\ k+r \end{array}\right) \quad \text{for } N = s-p-l-1, x = n-1, r = p-n \\ = \; & \sum_{k=0}^{N}\left(\begin{array}{c} x \\ k \end{array}\right)\left(\begin{array}{c} N+r \\ N-k \end{array}\right) \\ = \; & \left(\begin{array}{c} x+N+r \\ N \end{array}\right) = \left(\begin{array}{c} x+N+r \\ x+r \end{array}\right) = \left(\begin{array}{c} s-l-2 \\ p-1 \end{array}\right), \end{aligned} \tag{B.87}$$

by (F.4). Then, from (B.68), (B.85), (B.86) and (B.87), we have

$$\begin{aligned}
\text{(B.68)} &= \sum_{l=1}^{s-2p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=s-2p+1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=s-2p+1}^{s-p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-l-p+1}^{p} \tilde{\Sigma}_l^{s,s-n-1}. \qquad \text{(B.88)}
\end{aligned}$$

Applying (B.74) to the k and l sums in (B.71) gives $a = 0$, $b = s-n-2$, $c = s-n-1$, $c-b = 1$ and $c-a = s-n-1$, so that

$$\begin{aligned}
\text{(B.71)} &= -\sum_{n=s-p-1}^{s-2} \sum_{k=0}^{s-n-2} \sum_{l=1}^{s-n-k-1} \\
&= -\sum_{n=s-p-1}^{s-2} \sum_{l=1}^{s-n-1} \sum_{k=0}^{s-n-l-1}. \qquad \text{(B.89)}
\end{aligned}$$

Applying (B.74) again to the n and l sums in (B.89) gives $a = s-p-1, b = s-2$, $c = s-1$, $c-b = 1$ and $c-a = p$. Therefore,

$$\text{(B.71)} = -\sum_{l=1}^{p} \sum_{n=s-p-1}^{s-l-1} \sum_{k=0}^{s-n-l-1} \qquad \text{(B.90)}$$

Recovering the arguments from (B.71), we now evaluate the k sum appearing in (B.90).

$$\begin{aligned}
& \sum_{k=0}^{s-n-l-1} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \\
&= \sum_{k=0}^{N} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} N \\ k \end{pmatrix} \quad \text{for } N = s-n-l-1 \\
&= \sum_{k=0}^{N} \begin{pmatrix} n-1 \\ n-1-p+k \end{pmatrix} \begin{pmatrix} N \\ k \end{pmatrix} \\
&= \begin{pmatrix} N+n-1 \\ N+n-1-p \end{pmatrix} \\
&= \begin{pmatrix} N+n-1 \\ p \end{pmatrix}
\end{aligned}$$

$$= \begin{pmatrix} s-l-2 \\ p \end{pmatrix}, \tag{B.91}$$

by (F.6). Then, from (B.71), (B.90) and (B.91), we have

$$\text{(B.71)} = -\sum_{l=1}^{p}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1}. \tag{B.92}$$

For (B.72), the same sums as for (B.71) appear, so that (B.90) applies. Recovering the arguments from (B.72), we now evaluate the k sum appearing in (B.90).

$$\begin{aligned}
&\sum_{k=0}^{s-n-l-1} \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \\
&= \sum_{k=0}^{N} \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} \begin{pmatrix} N \\ k \end{pmatrix} \\
&= \sum_{k=0}^{N} \begin{pmatrix} n-1 \\ n-p+k \end{pmatrix} \begin{pmatrix} N \\ k \end{pmatrix} \quad \text{for } N = s-n-l-1 \\
&= \begin{pmatrix} N+n-1 \\ N+n-p \end{pmatrix} \\
&= \begin{pmatrix} N+n-1 \\ p-1 \end{pmatrix} \\
&= \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix},
\end{aligned} \tag{B.93}$$

by (F.6). Then, from (B.72), (B.90) and (B.93), we have

$$\text{(B.72)} = \sum_{l=1}^{p}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-p-1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1}. \tag{B.94}$$

Now consider (B.69), which is somewhat more involved. Applying (B.74) to the k and l sums in (B.69) gives $a = 0$, $b = p$, $c = s-n-1$, $c-b = s-n-p-1$ and $c-a = s-n-1$, so that

$$\text{(B.69)} = -\sum_{n=p+1}^{s-p-2} \sum_{k=0}^{p} \sum_{l=1}^{s-n-k-1}$$

$$= -\sum_{n=p+1}^{s-p-2}\sum_{l=1}^{s-p-n-1}\sum_{k=0}^{p} - \sum_{n=p+1}^{s-p-2}\sum_{l=s-p-n}^{s-n-1}\sum_{k=0}^{s-n-l-1}. \tag{B.95}$$

Applying (B.74) again to the first n and l sums in (B.95) gives $a = p+1$, $b = s-p-2$, $c = s-p-1$, $c-b = 1$ and $c-a = s-2p-2$, so that

$$\sum_{n=p+1}^{s-p-2}\sum_{l=1}^{s-p-n-1} = \sum_{l=1}^{s-2p-2}\sum_{n=p+1}^{s-p-l-1}. \tag{B.96}$$

For the second n and l sums in (B.95), we must consider whether (B.77) or (B.78) applies. We have $a = p+1$, $b = s-p-2$, $c = s-p$, and $d = s-1$, hence

$$c-a = s-2p-1, \quad c-b = 2, \quad d-a = s-p-2, \quad d-b = p+1. \tag{B.97}$$

(B.77) applies when

$$\begin{aligned} c-a \le d-b &\iff s-2p-1 \le p+1 \\ &\iff p \ge \frac{s-2}{3}, \end{aligned} \tag{B.98}$$

and (B.78) applies when

$$\begin{aligned} d-b \le c-a &\iff s-2p-1 \ge p+1 \\ &\iff p \le \frac{s-2}{3}. \end{aligned} \tag{B.99}$$

We are currently assuming $1 \le p \le (s-3)/2$ from (B.65), which implies $s \ge 5$. However,

$$\frac{s-3}{2} \ge \frac{s-2}{3} \iff 3s-9 \ge 2s-4 \iff s \ge 5,$$

so that

$$\frac{s-3}{2} \ge \frac{s-2}{3}.$$

This means, for $1 \le p \le (s-3)/2$, both cases (B.98) and (B.99) occur. So we need to consider two sub-cases:

$$1 \le p \le \frac{s-2}{3}, \tag{B.100}$$

$$\frac{s-2}{3} \le p \le \frac{s-3}{2}, \tag{B.101}$$

for which (B.78) and (B.77) apply, respectively.

Consider (B.100) first. Using (B.97) in (B.78), we find that the second n and l sums in (B.95) become

$$\sum_{n=p+1}^{s-p-2}\sum_{l=s-p-n}^{s-n-1} = \sum_{l=2}^{p}\sum_{n=s-p-l}^{s-p-2} + \sum_{l=p+1}^{s-2p-1}\sum_{n=s-p-l}^{s-l-1} + \sum_{l=s-2p}^{s-p-2}\sum_{n=p+1}^{s-l-1}. \tag{B.102}$$

Now consider (B.101). Using (B.97) in (B.77), we find that the second n and l sums in (B.95) become

$$\sum_{n=p+1}^{s-p-2}\sum_{l=s-p-n}^{s-n-1} = \sum_{l=2}^{s-2p-1}\sum_{n=s-p-l}^{s-p-2} + \sum_{l=s-2p}^{p}\sum_{n=p+1}^{s-p-2} + \sum_{l=p+1}^{s-p-2}\sum_{n=p+1}^{s-l-1}. \tag{B.103}$$

Recovering the arguments from (B.69), we now evaluate the k sum appearing in (B.95).

$$\begin{aligned}
&\sum_{k=0}^{s-n-l-1}\binom{n-1}{p-k}\binom{s-n-l-1}{k} \\
&= \sum_{k=0}^{N}\binom{n-1}{n-1-p+k}\binom{N}{k} \quad \text{for } N = s-n-l-1 \\
&= \binom{N+n-1}{N+n-1-p} \\
&= \binom{N+n-1}{p} \\
&= \binom{s-l-2}{p}, \qquad\qquad (B.104)
\end{aligned}$$

by (F.6). Then, from (B.69), (B.96), (B.102) and (B.104), we have

$$\begin{aligned}
\text{(B.69)} \;=\; & -\sum_{l=2}^{p}(-1)^l\binom{s-l-2}{p}\sum_{n=s-p-l}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \\
& -\sum_{l=p+1}^{s-2p-1}(-1)^l\binom{s-l-2}{p}\sum_{n=s-p-l}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \\
& -\sum_{l=s-2p}^{s-p-2}(-1)^l\binom{s-l-2}{p}\sum_{n=p+1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1}
\end{aligned}$$

$$- \sum_{l=1}^{s-2p-2} (-1)^l \binom{s-l-2}{p} \sum_{n=p+1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1}, \tag{B.105}$$

for $1 \leq p \leq \frac{s-2}{3}$. Similarly, from (B.69), (B.96), (B.103) and (B.104), we have

$$\begin{aligned}
\text{(B.69)} \quad = \quad & - \sum_{l=2}^{s-2p-1} (-1)^l \binom{s-l-2}{p} \sum_{n=s-p-l}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \\
& - \sum_{l=s-2p}^{p} (-1)^l \binom{s-l-2}{p} \sum_{n=p+1}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \\
& - \sum_{l=p+1}^{s-p-2} (-1)^l \binom{s-l-2}{p} \sum_{n=p+1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \\
& - \sum_{l=1}^{s-2p-2} (-1)^l \binom{s-l-2}{p} \sum_{n=p+1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1},
\end{aligned} \tag{B.106}$$

for $\frac{s-2}{3} \leq p \leq \frac{s-3}{2}$.

Now consider (B.70). Applying (B.74) to the k and l sums in (B.70) gives $a = 0$, $b = p-1$, $c = s-n-1$, $c-b = s-p-n$ and $c-a = s-n-1$, so that

$$\text{(B.70)} = \sum_{n=p+1}^{s-p-2} \sum_{k=0}^{p-1} \sum_{l=1}^{s-n-k-1} = \sum_{n=p+1}^{s-p-2} \sum_{l=1}^{s-p-n} \sum_{k=0}^{p-1} + \sum_{n=p+1}^{s-p-2} \sum_{l=s-p-n+1}^{s-n-1} \sum_{k=0}^{s-n-l-1}. \tag{B.107}$$

Applying (B.74) again to the first n and l sums in (B.107) gives $a = p+1$, $b = s-p-2$, $c = s-p$, $c-b = 2$ and $c-a = s-2p-1$, so that

$$\sum_{n=p+1}^{s-p-2} \sum_{l=1}^{s-p-n} = \sum_{l=1}^{2} \sum_{n=p+1}^{s-p-2} + \sum_{l=3}^{s-2p-1} \sum_{n=p+1}^{s-p-l}. \tag{B.108}$$

For the second n and l sums in (B.107), we must again consider whether (B.77) or (B.78) applies. We have $a = p+1$, $b = s-p-2$, $c = s-p+1$, and $d = s-1$, hence

$$c-a = s-2p, \quad c-b = 3, \quad d-a = s-p-2, \quad d-b = p+1. \tag{B.109}$$

(B.77) applies when

$$\begin{aligned}
c-a \leq d-b \quad & \Longleftrightarrow \quad s-2p \leq p+1 \\
& \Longleftrightarrow \quad p \geq \frac{s-1}{3},
\end{aligned} \tag{B.110}$$

and (B.78) applies when

$$\begin{aligned} d-b \leq c-a &\iff s-2p \geq p+1 \\ &\iff p \leq \frac{s-1}{3}. \end{aligned} \tag{B.111}$$

We are currently assuming $1 \leq p \leq (s-3)/2$ (B.65), which implies $s \geq 5$. However,

$$\frac{s-1}{3} \geq \frac{s-3}{2} \iff 2s-2 \geq 3s-9 \iff s \geq 7,$$

so that

$$\frac{s-3}{2} \geq \frac{s-2}{3}.$$

Again this means, for $1 \leq p \leq (s-3)/2$, both cases (B.110) and (B.111) occur. So we need to consider two sub-cases:

$$1 \leq \quad p \quad \leq \frac{s-1}{3}, \tag{B.112}$$

$$\frac{s-1}{3} \leq \quad p \quad \leq \frac{s-3}{2}, \tag{B.113}$$

for which (B.78) and (B.77) apply, respectively.

Consider (B.112) first. Using (B.109) in (B.78), we see that the second n and l sums in (B.107) become

$$\sum_{n=p+1}^{s-p-2} \sum_{l=s-p-n+1}^{s-n-1} = \sum_{l=3}^{p} \sum_{n=s-p-l+1}^{s-p-2} + \sum_{l=p+1}^{s-2p} \sum_{n=s-p-l+1}^{s-l-1} + \sum_{l=s-2p+1}^{s-p-2} \sum_{n=p+1}^{s-l-1}. \tag{B.114}$$

Now consider (B.113). Using (B.109) in (B.77), we see that the second n and l sums in (B.107) become

$$\sum_{n=p+1}^{s-p-2} \sum_{l=s-p-n+1}^{s-n-1} = \sum_{l=3}^{s-2p} \sum_{n=s-p-l+1}^{s-p-2} + \sum_{l=s-2p+1}^{p} \sum_{n=p+1}^{s-p-2} + \sum_{l=p+1}^{s-p-2} \sum_{n=p+1}^{s-l-1}. \tag{B.115}$$

Recovering the arguments from (B.70), we now evaluate the k sums appearing in (B.107). For the first k sum, we have

$$\sum_{k=0}^{p-1} \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} = \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix}, \tag{B.116}$$

by (F.4). For the second k sum,

$$\begin{aligned}
&\sum_{k=0}^{s-n-l-1} \left(\begin{array}{c} n-1 \\ p-k-1 \end{array} \right) \left(\begin{array}{c} s-n-l-1 \\ k \end{array} \right) \\
&= \sum_{k=0}^{N} \left(\begin{array}{c} n-1 \\ n-p+k \end{array} \right) \left(\begin{array}{c} N \\ k \end{array} \right) \quad \text{for } N = s-n-l-1 \\
&= \left(\begin{array}{c} N+n-1 \\ N+n-p \end{array} \right) \\
&= \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right), \qquad \text{(B.117)}
\end{aligned}$$

by (F.6). Then, from (B.70), (B.108), (B.114), (B.116) and (B.117), we have

$$\begin{aligned}
\text{(B.70)} &= \sum_{l=3}^{p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=s-p-l+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=p+1}^{s-2p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=s-p-l+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=s-2p+1}^{s-p-2} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=1}^{2} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=3}^{s-2p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1}, \qquad \text{(B.118)}
\end{aligned}$$

for $1 \leq p \leq \frac{s-1}{3}$. Similarly, from (B.70), (B.108), (B.115), (B.116) and (B.117), we have

$$\begin{aligned}
\text{(B.70)} &= \sum_{l=3}^{s-2p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=s-p-l+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=s-2p+1}^{p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \\
&+ \sum_{l=p+1}^{s-p-2} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1}
\end{aligned}$$

$$+ \quad \sum_{l=1}^{2}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1}$$

$$= \quad \sum_{l=3}^{s-2p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1}, \tag{B.119}$$

for $\frac{s-1}{3} \leq p \leq \frac{s-3}{2}$.

Recall that we are trying to show (B.58) for $p = 1, \ldots, s-3$. So far, we have all the cases for $1 \leq p \leq (s-3)/2$, and this interval is handled in two parts, $1 \leq p \leq (s-2)/3$ and $(s-1)/3 \leq p \leq (s-3)/2$. The left side of (B.58) is (B.73). When $1 \leq p \leq (s-2)/3$, the right side is assembled from (B.66), (B.83), (B.88), (B.105), (B.118), (B.92) and (B.94). This equals

$$(-1)^{s+p}\left[\begin{pmatrix} s-2 \\ p \end{pmatrix}(1-s) + \begin{pmatrix} s-2 \\ p-1 \end{pmatrix}\right. \tag{B.120}$$

$$- \quad \sum_{l=1}^{s-2p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.121}$$

$$- \quad \sum_{l=s-2p}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.122}$$

$$- \quad \sum_{l=s-2p}^{s-p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-l-p}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.123}$$

$$+ \quad \sum_{l=1}^{s-2p}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.124}$$

$$+ \quad \sum_{l=s-2p+1}^{s-p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.125}$$

$$+ \quad \sum_{l=s-2p+1}^{s-p}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-l-p+1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.126}$$

$$- \quad \sum_{l=2}^{p}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-l}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.127}$$

$$- \quad \sum_{l=p+1}^{s-2p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-l}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.128}$$

$$- \quad \sum_{l=s-2p}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=p+1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.129}$$

$$
\begin{aligned}
&- \sum_{l=1}^{s-2p-2} (-1)^l \binom{s-l-2}{p} \sum_{n=p+1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1} && \text{(B.130)}\\
&+ \sum_{l=3}^{p} (-1)^l \binom{s-l-2}{p-1} \sum_{n=s-p-l+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.131)}\\
&+ \sum_{l=p+1}^{s-2p} (-1)^l \binom{s-l-2}{p-1} \sum_{n=s-p-l+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.132)}\\
&+ \sum_{l=s-2p+1}^{s-p-2} (-1)^l \binom{s-l-2}{p-1} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.133)}\\
&+ \sum_{l=1}^{2} (-1)^l \binom{s-l-2}{p-1} \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.134)}\\
&+ \sum_{l=3}^{s-2p-1} (-1)^l \binom{s-l-2}{p-1} \sum_{n=p+1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.135)}\\
&- \sum_{l=1}^{p} (-1)^l \binom{s-l-2}{p} \sum_{n=s-p-1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} && \text{(B.136)}\\
&+ \sum_{l=1}^{p} (-1)^l \binom{s-l-2}{p-1} \sum_{n=s-p-1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \Bigg] . && \text{(B.137)}
\end{aligned}
$$

We need to simplify this large expression to show it equals (B.73). The first step is to coalesce sums with adjacent ranges. It may help when reading this to check off each expression as it is used. In (B.123), we may take the upper limit on l to be $s-p-2$, since the $l = s-p-1$ term is zero. Then

$$
\text{(B.122)} + \text{(B.123)} \;=\; -\sum_{l=s-2p}^{s-p-2} (-1)^l \binom{s-l-2}{p} \sum_{n=1}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.138)}
$$

$$
\begin{aligned}
\text{(B.136)} + \text{(B.127)} \;=\; &-\sum_{l=2}^{p} (-1)^l \binom{s-l-2}{p} \sum_{n=s-p-l}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} && \text{(B.139)}\\
&-(-1) \binom{s-3}{p} \sum_{n=s-p-1}^{s-2} n\tilde{\Sigma}_1^{s,s-n-1} && \text{(B.140)}
\end{aligned}
$$

$$
\text{(B.137)} + \text{(B.131)} \;=\; \sum_{l=3}^{p} (-1)^l \binom{s-l-2}{p-1} \sum_{n=s-p-l+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.141)}
$$

$$+\sum_{l=1}^{2}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-p-1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.142)}$$

$$\text{(B.121)} + \text{(B.130)} \;=\; -\sum_{l=1}^{s-2p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.143)}$$

$$-(-1)^{s-2p-1} \begin{pmatrix} 2p-1 \\ p \end{pmatrix} \sum_{n=1}^{p} n\tilde{\Sigma}_{s-2p-1}^{s,s-n-1} \qquad \text{(B.144)}$$

$$\text{(B.134)} + \text{(B.142)} \;=\; \sum_{l=1}^{2}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.145)}$$

$$\text{(B.138)} + \text{(B.129)} \;=\; -\sum_{l=s-2p}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.146)}$$

$$\text{(B.139)} + \text{(B.128)} \;=\; -\sum_{l=2}^{s-2p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-l}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.147)}$$

$$\text{(B.141)} + \text{(B.132)} \;=\; \sum_{l=3}^{s-2p}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-p-l+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.148)}$$

$$\text{(B.135)} + \text{(B.148)} \;=\; \sum_{l=3}^{s-2p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.149)}$$

$$+(-1)^{s-2p} \begin{pmatrix} 2p-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{2p-1} \tilde{\Sigma}_{s-2p}^{s,s-n-1} \qquad \text{(B.150)}$$

$$\text{(B.143)} + \text{(B.144)} + \text{(B.147)} + \text{(B.140)} \;=$$

$$-\sum_{l=1}^{s-2p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.151)}$$

$$\text{(B.146)} + \text{(B.151)} \quad = \quad -\sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.152)}$$

$$\text{(B.145)} + \text{(B.149)} \quad = \quad \sum_{l=1}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.153)}$$

$$\text{(B.153)} + \text{(B.150)} \quad = \quad \sum_{l=1}^{s-2p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.154)}$$

$$\text{(B.154)} + \text{(B.133)} \quad = \quad \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.155)}$$

In (B.126), we may take the upper limit on l to be $s-p-1$, since the $l = s-p$ term is zero. Then,

$$\text{(B.125)} + \text{(B.126)} \quad = \quad \sum_{l=s-2p+1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.156)}$$

$$\text{(B.156)} + \text{(B.124)} \quad = \quad \sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.157)}$$

$$\text{(B.155)} + \text{(B.157)} \quad = \quad \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.158)}$$

$$+(-1)^{s-p-1} \begin{pmatrix} p-1 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_{s-p-1}^{s,s-n-1} \qquad \text{(B.159)}$$

$$\text{(B.158)} + \text{(B.159)} \quad = \quad \sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.160)}$$

The only remaining expressions that have not yet been used are (B.120), (B.152) and

(B.160). Therefore we have established that the right side of (B.46) is their sum,

$$\begin{aligned}
& (-1)^{s+p}\left[\left(\begin{array}{c} s-2 \\ p \end{array}\right)(1-s)+\left(\begin{array}{c} s-2 \\ p-1 \end{array}\right)\right. \\
+ & \sum_{l=1}^{s-p-1}(-1)^l\left(\begin{array}{c} s-l-2 \\ p-1 \end{array}\right)\sum_{n=1}^{s-l-1}\tilde{\Sigma}_l^{s,s-n-1} \\
- & \left.\sum_{l=1}^{s-p-2}(-1)^l\left(\begin{array}{c} s-l-2 \\ p \end{array}\right)\sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1}\right]. \qquad \text{(B.161)}
\end{aligned}$$

We must show this equals (B.73). Note that in (B.161),

$$\begin{aligned}
\left(\begin{array}{c} s-2 \\ p \end{array}\right)(1-s)+\left(\begin{array}{c} s-2 \\ p-1 \end{array}\right) &= \left(\begin{array}{c} s-2 \\ p \end{array}\right)+\left(\begin{array}{c} s-2 \\ p-1 \end{array}\right)-s\left(\begin{array}{c} s-2 \\ p \end{array}\right) \\
&= \left(\begin{array}{c} s-1 \\ p \end{array}\right)-s\left(\begin{array}{c} s-2 \\ p \end{array}\right).
\end{aligned}$$

Since the first of these terms also appears in (B.73), what we need to show is

$$\begin{aligned}
& \sum_{l=1}^{s-p-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ p \end{array}\right)\Sigma_l^s+s\left(\begin{array}{c} s-2 \\ p \end{array}\right) \\
- & \sum_{l=1}^{s-p-1}(-1)^l\left(\begin{array}{c} s-l-2 \\ p-1 \end{array}\right)\sum_{n=1}^{s-l-1}\tilde{\Sigma}_l^{s,s-n-1} \\
+ & \sum_{l=1}^{s-p-2}(-1)^l\left(\begin{array}{c} s-l-2 \\ p \end{array}\right)\sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \\
= & \ 0 \qquad \text{(B.162)}
\end{aligned}$$

We now express the Σ_l^s in terms of $\tilde{\Sigma}_l^{s,s-n}$. Applying (B.54) to the first line of (B.162), we get

$$\begin{aligned}
& \sum_{l=1}^{s-p-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ p \end{array}\right)\Sigma_l^s \\
= & \sum_{l=1}^{s-p-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ p \end{array}\right)\sum_{n=1}^{s-l}\tilde{\Sigma}_l^{s,s-n} \\
= & \sum_{l=1}^{s-p-1}(-1)^l\left(\begin{array}{c} s-l-1 \\ p \end{array}\right)\sum_{n=0}^{s-l-1}\tilde{\Sigma}_l^{s,s-(n+1)}
\end{aligned}$$

$$= \sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-1 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} + \sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-1 \\ p \end{pmatrix} \tilde{\Sigma}_l^{s,s-1}.$$

Substituting this into (B.162), we now need to show

$$\begin{aligned} &\sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-1 \\ p \end{pmatrix} \tilde{\Sigma}_l^{s,s-1} + s \begin{pmatrix} s-2 \\ p \end{pmatrix} \\ &+ \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \\ &+ \sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \\ &= 0, \end{aligned} \tag{B.163}$$

which is equivalent to

$$\begin{aligned} &\sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-1 \\ p \end{pmatrix} \tilde{\Sigma}_l^{s,s-1} + s \begin{pmatrix} s-2 \\ p \end{pmatrix} \\ &+ \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} (n+1) \tilde{\Sigma}_l^{s,s-n-1} \\ &= 0, \end{aligned} \tag{B.164}$$

because the $l = s-p-1$ term in the third line of (B.163) is zero. Now the $l = 1$ term in the first sum of (B.164) is

$$- \begin{pmatrix} s-2 \\ p \end{pmatrix} \tilde{\Sigma}_1^{s,s-1} = - \begin{pmatrix} s-2 \\ p \end{pmatrix} \begin{pmatrix} s \\ s-1 \end{pmatrix} = -s \begin{pmatrix} s-2 \\ p \end{pmatrix},$$

by (B.22). Thus the first line of (B.164) becomes

$$\sum_{l=2}^{s-p-1} (-1)^l \begin{pmatrix} s-l-1 \\ p \end{pmatrix} \tilde{\Sigma}_l^{s,s-1} = - \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \tilde{\Sigma}_{l+1}^{s,s-1}.$$

So what we want to prove, (B.164), reduces to

$$- \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \tilde{\Sigma}_{l+1}^{s,s-1}$$

$$
\begin{aligned}
&+ \sum_{l=1}^{s-p-2} (-1)^l \binom{s-l-2}{p} \sum_{n=1}^{s-l-1} (n+1)\tilde{\Sigma}_l^{s,s-n-1} \\
&= 0,
\end{aligned}
$$

which is equivalent to

$$
\sum_{l=1}^{s-p-2} (-1)^l \binom{s-l-2}{p} \left[\sum_{n=1}^{s-l-1} (n+1)\tilde{\Sigma}_l^{s,s-n-1} - \tilde{\Sigma}_{l+1}^{s,s-1} \right] = 0,
$$

which in turn holds if

$$
\sum_{n=1}^{s-l-1} (n+1)\tilde{\Sigma}_l^{s,s-n-1} - \tilde{\Sigma}_{l+1}^{s,s-1} = 0, \quad \text{for } l = 1, \ldots, s-p-2. \tag{B.165}
$$

To establish (B.165), first consider $l \geq 2$. Using (B.22), we see that

$$
\begin{aligned}
&\sum_{n=1}^{s-l-1} (n+1)\tilde{\Sigma}_l^{s,s-n-1} \\
= &\sum_{n=1}^{s-l-1} (n+1) \sum_{n+2 \leq i_1 < \cdots < i_{l-1} \leq s-1} \binom{i_1}{n+1} \binom{i_2}{i_1} \cdots \binom{i_{l-1}}{i_{l-2}} \binom{s}{i_{l-1}} \\
= &\sum_{n=1}^{s-l-1} (n+1) \sum_{n+2 \leq i_2 < \cdots < i_l \leq s-1} \binom{i_2}{n+1} \binom{i_3}{i_2} \cdots \binom{i_l}{i_{l-1}} \binom{s}{i_l}. \qquad \text{(B.166)}
\end{aligned}
$$

On the other hand, $l \geq 2$ implies $l+1 \geq 2$, so

$$
\begin{aligned}
\tilde{\Sigma}_{l+1}^{s,s-1} &= \sum_{2 \leq i_1 < \cdots < i_l \leq s-1} \binom{i_1}{1} \binom{i_2}{i_1} \cdots \binom{i_l}{i_{l-1}} \binom{s}{i_l} \\
&= \sum_{i_1=2}^{s-l} \sum_{i_1 < i_2 < \cdots < i_l \leq s-1} \binom{i_1}{1} \binom{i_2}{i_1} \cdots \binom{i_l}{i_{l-1}} \binom{s}{i_l} \\
&= \sum_{i_1=3}^{s-l+1} \sum_{i_1 \leq i_2 < \cdots < i_l \leq s-1} \binom{i_1-1}{1} \binom{i_2}{i_1-1} \cdots \binom{i_l}{i_{l-1}} \binom{s}{i_l} \\
&= \sum_{n=1}^{s-l-1} \sum_{n+2 \leq i_2 < \cdots < i_l \leq s-1} \binom{n+1}{1} \binom{i_2}{n+1} \cdots \binom{i_l}{i_{l-1}} \binom{s}{i_l} \\
&= \text{(B.166)}.
\end{aligned}
$$

Now consider $l = 1$. Using (B.22), we have

$$\begin{aligned}\sum_{n=1}^{s-l-1}(n+1)\tilde{\Sigma}_l^{s,s-n-1} &= \sum_{n=1}^{s-2}(n+1)\tilde{\Sigma}_1^{s,s-n-1} \\ &= \sum_{n=1}^{s-2}(n+1)\left(\begin{array}{c} s \\ s-n-1 \end{array}\right) \\ &= \sum_{n=1}^{s-2}(n+1)\left(\begin{array}{c} s \\ n+1 \end{array}\right). \end{aligned} \tag{B.167}$$

Also,

$$\begin{aligned}\tilde{\Sigma}_{l+1}^{s,s-1} &= \tilde{\Sigma}_2^{s,s-1} \\ &= \sum_{2\le i_1\le s-1}\left(\begin{array}{c} i_1 \\ 1 \end{array}\right)\left(\begin{array}{c} s \\ i_1 \end{array}\right) \\ &= \sum_{n=2}^{s-1} n\left(\begin{array}{c} s \\ n \end{array}\right) \\ &= \sum_{n=1}^{s-2}(n+1)\left(\begin{array}{c} s \\ n+1 \end{array}\right) \\ &= \text{(B.166)}.\end{aligned}$$

Therefore (B.165) is proven, as required.

Recall that we are in the process of showing (B.58) for $1 \le p \le (s-3)/2$, and have just completed the case (B.65), where $1 \le p \le (s-2)/3$. We now address the case where $(s-2)/3 < p \le (s-3)/2$, that is, $(s-1)/3 \le p \le (s-3)/2$. In this case (B.106) and (B.119) apply, so that the right side of (B.58) is assembled from (B.66), (B.83), (B.88), (B.106), (B.119), (B.92) and (B.94). This equals

$$(-1)^{s+p}\left[\left(\begin{array}{c} s-2 \\ p \end{array}\right)(1-s) + \left(\begin{array}{c} s-2 \\ p-1 \end{array}\right)\right] \tag{B.168}$$

$$- \sum_{l=1}^{s-2p-1}(-1)^l\left(\begin{array}{c} s-l-2 \\ p \end{array}\right)\sum_{n=1}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.169}$$

$$- \sum_{l=s-2p}^{s-p-2}(-1)^l\left(\begin{array}{c} s-l-2 \\ p \end{array}\right)\sum_{n=1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.170}$$

$$- \sum_{l=s-2p}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-l-p}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.171}$$

$$+ \sum_{l=1}^{s-2p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.172}$$

$$+ \sum_{l=s-2p+1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.173}$$

$$+ \sum_{l=s-2p+1}^{s-p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-l-p+1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.174}$$

$$- \sum_{l=2}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-l}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.175}$$

$$- \sum_{l=s-2p}^{p} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=p+1}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.176}$$

$$- \sum_{l=p+1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=p+1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.177}$$

$$- \sum_{l=1}^{s-2p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=p+1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.178}$$

$$+ \sum_{l=3}^{s-2p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-p-l+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.179}$$

$$+ \sum_{l=s-2p+1}^{p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.180}$$

$$+ \sum_{l=p+1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.181}$$

$$+ \sum_{l=1}^{2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.182}$$

$$+ \sum_{l=3}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.183}$$

$$- \sum_{l=1}^{p} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \tag{B.184}$$

$$+ \sum_{l=1}^{p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-p-1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \Bigg]. \tag{B.185}$$

We now simplify this large expression to show it equals (B.73). We first coalesce sums with adjacent ranges. Again, it may help when reading this to check off each expression as it is used. This time, we'll do all the negative sums first, and then the positive ones. In (B.171), we may take the upper limit on l to be $s-p-2$, since the $l = s-p-1$ term is zero. Then

$$\text{(B.170)} + \text{(B.171)} \quad = \quad -\sum_{l=s-2p}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.186)}$$

$$\text{(B.169)} + \text{(B.178)} \quad = \quad -\sum_{l=1}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.187)}$$

$$\begin{aligned} \text{(B.175)} + \text{(B.187)} \quad &= \quad -\sum_{l=2}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \\ &\quad -(-1) \begin{pmatrix} s-3 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n\tilde{\Sigma}_1^{s,s-n-1} \\ &= \quad -\sum_{l=1}^{s-2p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.188)} \end{aligned}$$

Now $p \leq (s-3)/2 \implies p \leq (s-2)/2 \implies p \leq s-p-2$, so that

$$\text{(B.176)} + \text{(B.186)} \quad = \quad -\sum_{l=s-2p}^{p} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.189)}$$

$$-\sum_{l=p+1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.190)}$$

$$\text{(B.190)} + \text{(B.177)} \quad = \quad -\sum_{l=p+1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.191)}$$

$$\text{(B.188)} + \text{(B.189)} \quad = \quad -\sum_{l=1}^{p} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.192)}$$

$$\text{(B.184)} + \text{(B.192)} \quad = \quad -\sum_{l=1}^{p}(-1)^l \left(\begin{array}{c} s-l-2 \\ p \end{array} \right) \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.193)}$$

$$\text{(B.191)} + \text{(B.193)} \quad = \quad -\sum_{l=1}^{s-p-2}(-1)^l \left(\begin{array}{c} s-l-2 \\ p \end{array} \right) \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.194)}$$

Now we do the positive sums.

$$\text{(B.172)} + \text{(B.183)} + \text{(B.179)} \quad = \quad \sum_{l=3}^{s-2p-1}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.195)}$$

$$+ \sum_{l=1}^{2}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.196)}$$

$$(-1)^{s-2p} \left(\begin{array}{c} 2p-2 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_{s-2p}^{s,s-n-1} \qquad \text{(B.197)}$$

$$(-1)^{s-2p} \left(\begin{array}{c} 2p-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_{s-2p}^{s,s-n-1} \qquad \text{(B.198)}$$

$$\text{(B.196)} + \text{(B.182)} \quad = \quad \sum_{l=1}^{2}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.199)}$$

$$\text{(B.195)} + \text{(B.199)} \quad = \quad \sum_{l=1}^{s-2p-1}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.200)}$$

$$\text{(B.180)} + \text{(B.198)} \quad = \quad \sum_{l=s-2p}^{p}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.201)}$$

$$\text{(B.200)} + \text{(B.201)} \quad = \quad \sum_{l=1}^{p}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=p+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.202)}$$

$$+ \sum_{l=1}^{s-2p-1}(-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.203)}$$

$$\text{(B.173)} + \text{(B.174)} \quad = \quad \sum_{l=s-2p+1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.204)}$$

$$\text{(B.203)} + \text{(B.197)} + \text{(B.204)} \quad = \quad \sum_{l=1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.205)}$$

$$\text{(B.181)} + \text{(B.205)} \quad = \quad \sum_{l=p+1}^{s-p-2} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.206)}$$

$$+ \sum_{l=1}^{p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.207)}$$

$$(-1)^{s-p-1} \left(\begin{array}{c} p-1 \\ p-1 \end{array} \right) \sum_{n=1}^{p} \tilde{\Sigma}_{s-p-1}^{s,s-n-1} \qquad \text{(B.208)}$$

$$\text{(B.202)} + \text{(B.207)} \quad = \quad \sum_{l=1}^{p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.209)}$$

$$\text{(B.209)} + \text{(B.185)} \quad = \quad \sum_{l=1}^{p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.210)}$$

$$\text{(B.210)} + \text{(B.206)} + \text{(B.208)} \quad = \quad \sum_{l=1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.211)}$$

The only remaining expressions that have not yet been used are (B.168), (B.194) and (B.211). Therefore we have established that the right side of (B.46) is their sum,

$$(-1)^{s+p} \left[\left(\begin{array}{c} s-2 \\ p \end{array} \right) (1-s) + \left(\begin{array}{c} s-2 \\ p-1 \end{array} \right) \right.$$

$$+ \quad \sum_{l=1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1}$$

$$- \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \Bigg].$$

This is exactly the same as (B.161), which we have already shown to be equal to (B.73). Therefore we have completed the entire range of $1 \leq p \leq (s-3)/2$. Now we assume that

$$\frac{s-1}{2} \leq p \leq s-3.$$

As in the case of (B.65), we have (B.61) and (B.63), (B.61) and (B.64), and (B.62) and (B.64). Thus (B.60) becomes

$$(-1)^{s+p} \left[\begin{pmatrix} s-2 \\ p \end{pmatrix} (1-s) + \begin{pmatrix} s-2 \\ p-1 \end{pmatrix} \right] \tag{B.212}$$

$$+ \quad (-1)^{s+p} \left\{ \sum_{n=1}^{s-p-2} \left[-n \sum_{k=p-n+1}^{p} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} T^{(s)}_{s-n-1,k} \right.\right.$$

$$\left.\left. + \sum_{k=p-n}^{p-1} \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} T^{(s)}_{s-n-1,k} \right] \right. \tag{B.213}$$

$$+ \quad \sum_{n=s-p-1}^{p} \left[-n \sum_{k=p-n+1}^{s-n-2} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} T^{(s)}_{s-n-1,k} + \sum_{k=p-n}^{s-n-2} \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} T^{(s)}_{s-n-1,k} \right] \tag{B.214}$$

$$+ \quad \left. \sum_{n=p+1}^{s-2} \left[-n \sum_{k=0}^{s-n-2} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} T^{(s)}_{s-n-1,k} + \sum_{k=0}^{s-n-2} \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} T^{(s)}_{s-n-1,k} \right] \right\}. \tag{B.215}$$

Using (B.40), we see that this reduces to

$$(-1)^{s+p} \left\{ \begin{pmatrix} s-2 \\ p \end{pmatrix} (1-s) + \begin{pmatrix} s-2 \\ p-1 \end{pmatrix} \right. \tag{B.216}$$

$$- \quad \sum_{n=1}^{s-p-2} \sum_{k=p-n+1}^{p} \sum_{l=1}^{s-n-k-1} (-1)^l n \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.217}$$

$$+ \quad \sum_{n=1}^{s-p-2} \sum_{k=p-n}^{p-1} \sum_{l=1}^{s-n-k-1} (-1)^l \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.218}$$

$$- \quad \sum_{n=s-p-1}^{p} \sum_{k=p-n-1}^{s-n-2} \sum_{l=1}^{s-n-k-1} (-1)^l n \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.219}$$

$$\left. + \quad \sum_{n=s-p-1}^{p} \sum_{k=p-n}^{s-n-2} \sum_{l=1}^{s-n-k-1} (-1)^l \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,s-n-1} \right\} \tag{B.220}$$

$$- \sum_{n=p+1}^{s-2} \sum_{k=0}^{s-n-2} \sum_{l=1}^{s-n-k-1} (-1)^l n \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.221}$$

$$+ \sum_{n=p+1}^{s-2} \sum_{k=0}^{s-n-2} \sum_{l=1}^{s-n-k-1} (-1)^l \begin{pmatrix} n-1 \\ p-k-1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \tilde{\Sigma}_l^{s,s-n-1} \Bigg\}. \tag{B.222}$$

Comparing this to (B.73), we again see that an appropriate strategy is to switch the order of the sums from n, k, l to l, k, n and then to evaluate the k sum, since $\tilde{\Sigma}_l^{s,s-n-1}$ is independent of k. Note that the k and l sums of (B.217), (B.218), (B.221) and (B.222) are the same as in (B.67), (B.68), (B.71) and (B.72), respectively, so we can use the same reversals as before, namely (B.79), (B.84) and (B.89). Therefore we have from (B.79), using a different range on the n sum,

$$\text{(B.217)} \;=\; - \sum_{n=1}^{s-p-2} \sum_{l=1}^{s-n-p-1} \sum_{k=p-n+1}^{p} - \sum_{n=1}^{s-p-2} \sum_{l=s-n-p}^{s-p-2} \sum_{k=p-n+1}^{s-n-l-1}. \tag{B.223}$$

Applying (B.74) to the first n and l sums in (B.223), we have $a = 1$, $b = s-p-2$, $c = s-p-1$, $c-b = 1$ and $c-a = s-p-2$. Applying (B.75) to the second n and l sums in (B.223), we have $a = s-p$ and $b = s-p-2$, $p \leftarrow s-p-2$. Therefore,

$$\text{(B.217)} \;=\; -\sum_{l=1}^{0} - \sum_{l=1}^{s-p-2} \sum_{n=1}^{s-p-l-1} \sum_{k=p-n+1}^{p} - \sum_{l=2}^{s-p-1} \sum_{n=s-p-l}^{s-p-2} \sum_{k=p-n+1}^{s-n-l-1} - \sum_{l=s-p}^{s-p-2}. \tag{B.224}$$

(Note that the first and last sums are empty). The k sums are the same as (B.81) and (B.82), so we have

$$\begin{aligned} \text{(B.217)} \;=\; & - \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \\ & - \sum_{l=2}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-l}^{s-p-2} n \tilde{\Sigma}_l^{s,s-n-1}. \end{aligned} \tag{B.225}$$

From (B.84), only changing the ranges on the n sums, we have

$$\text{(B.218)} \;=\; \sum_{n=1}^{s-p-2} \sum_{l=1}^{s-p-n} \sum_{k=p-n}^{p-1} + \sum_{n=1}^{s-p-2} \sum_{l=s-p-n+1}^{s-p-1} \sum_{k=p-n}^{s-n-l-1}. \tag{B.226}$$

Applying (B.74) to the first n and l sums in (B.226), we have $a = 1, b = s-p-2, c = s-p$, $c-b = 2$ and $c-a = s-p-1$. Applying (B.75) to the second n and l sums in (B.226),

we have $a = s - p + 1$ and $b = s - p - 1$, $p \leftarrow s - p - 2$. Therefore,

$$\text{(B.218)} \quad = \quad \sum_{l=1}^{2} \sum_{n=1}^{s-p-2} \sum_{k=p-n}^{p-1} + \sum_{l=3}^{s-p-1} \sum_{n=1}^{s-p-l} \sum_{k=p-n}^{p-1} + \sum_{l=3}^{s-p} \sum_{n=s-p-l+1}^{s-p-2} \sum_{k=p-n}^{s-n-l-1} + \sum_{l=s-p+1}^{s-p-1} \quad \text{(B.227)}$$

(Note that the last sum is empty). The k sums are the same as (B.86) and (B.87), so we have

$$\begin{aligned} \text{(B.218)} \quad = \quad & \sum_{l=1}^{2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-1} \tilde{\Sigma}_l^{s,s-n-1} \\ & + \sum_{l=3}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} \\ & + \sum_{l=3}^{s-p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=s-p-l+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1}. \end{aligned} \quad \text{(B.228)}$$

From (B.89), only changing the range on the n sum, we have

$$\text{(B.221)} \quad = \quad - \sum_{n=p+1}^{s-2} \sum_{l=1}^{s-n-1} \sum_{k=0}^{s-n-l-1} . \quad \text{(B.229)}$$

Applying a slight variation of (B.74) to the n and l sums in (B.229), we have $a = p + 1$, $b = s - 2$, $c = s - 1$, $c - b = 1$ and $c - a = s - p - 2$. Therefore,

$$\text{(B.221)} \quad = \quad - \sum_{l=1}^{0} - \sum_{l=1}^{s-p-2} \sum_{n=p+1}^{s-l-1} \sum_{k=0}^{s-n-l-1} . \quad \text{(B.230)}$$

(Note that the first sum is empty). The k sum is the same as (B.91), so we have

$$\text{(B.221)} \quad = \quad - \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=p+1}^{s-l-1} n \tilde{\Sigma}_l^{s,s-n-1}. \quad \text{(B.231)}$$

Now (B.222) has the same k and l sums as (B.221), and its k sum is (B.93), so we have

$$\text{(B.222)} \quad = \quad \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1}. \quad \text{(B.232)}$$

Expressions (B.219) and (B.220) cannot be recycled, so we compute them from scratch. Applying a slight variation of (B.74) to (B.219), we have $a = p - n - 1$, $b = s - n - 2$,

$c = s - n - 1$, $c - b = 1$ and $c - a = s - p$. Therefore

$$\begin{aligned}
\text{(B.219)} &= -\sum_{n=s-p-1}^{p} \sum_{k=p-n-1}^{s-n-2} \sum_{l=1}^{s-n-k-1} \\
&= -\sum_{n=s-p-1}^{p} \sum_{l=1}^{s-p} \sum_{k=p-n-1}^{s-n-l-1} \\
&= -\sum_{l=1}^{s-p} \sum_{n=s-p-1}^{p} \sum_{k=p-n-1}^{s-n-l-1} . \qquad \text{(B.233)}
\end{aligned}$$

The k sum is

$$\begin{aligned}
& \sum_{k=p-n-1}^{s-n-l-1} \begin{pmatrix} n-1 \\ p-k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ k \end{pmatrix} \\
= & \sum_{k=0}^{s-p-l} \begin{pmatrix} n-1 \\ n-k+1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ p-n+k-1 \end{pmatrix} \\
= & \sum_{k=0}^{s-p-l-2} \begin{pmatrix} n-1 \\ n-k+1 \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ p-n+k-1 \end{pmatrix} \\
= & \sum_{k=0}^{s-p-l-2} \begin{pmatrix} n-1 \\ k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ s-p-l-k-2 \end{pmatrix} \\
= & \sum_{k=0}^{N} \begin{pmatrix} n-1 \\ k \end{pmatrix} \begin{pmatrix} s-n-l-1 \\ N-k \end{pmatrix} \quad \text{for } N = s-p-l-2 \\
= & \begin{pmatrix} s-l-2 \\ s-p-l-2 \end{pmatrix} \\
= & \begin{pmatrix} s-l-2 \\ p \end{pmatrix} , \qquad \text{(B.234)}
\end{aligned}$$

by (F.4). Thus we have

$$\text{(B.219)} = -\sum_{l=1}^{s-p} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=s-p-1}^{p} n \tilde{\Sigma}_l^{s,s-n-1} . \qquad \text{(B.235)}$$

Applying a slight variation of (B.74) to (B.220), we have $a = p - n$, $b = s - n - 2$,

$c = s - n - 1$, $c - b = 1$ and $c - a = s - p - 1$. Therefore

$$\begin{aligned}
\text{(B.220)} &= \sum_{n=s-p-1}^{p} \sum_{k=p-n}^{s-n-2} \sum_{l=1}^{s-n-k-1} \\
&= \sum_{n=s-p-1}^{p} \sum_{l=1}^{s-p-1} \sum_{k=p-n}^{s-n-l-1} \\
&= \sum_{l=1}^{s-p-1} \sum_{n=s-p-1}^{p} \sum_{k=p-n}^{s-n-l-1} . \qquad \text{(B.236)}
\end{aligned}$$

The k sum is (B.87), so we have

$$\text{(B.220)} = \sum_{l=1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=s-p-1}^{p} \tilde{\Sigma}_l^{s,s-n-1} . \qquad \text{(B.237)}$$

We can now assemble the right side of (B.58) from (B.212), (B.225), (B.228), (B.235), (B.237), (B.231) and (B.232). This equals

$$\begin{aligned}
& (-1)^{s+p} \left[\left(\begin{array}{c} s-2 \\ p \end{array} \right) (1-s) + \left(\begin{array}{c} s-2 \\ p-1 \end{array} \right) \right. && \text{(B.238)} \\
- & \sum_{l=1}^{s-p-2} (-1)^l \left(\begin{array}{c} s-l-2 \\ p \end{array} \right) \sum_{n=1}^{s-p-l-1} n \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.239)} \\
- & \sum_{l=2}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p \end{array} \right) \sum_{n=s-p-l}^{s-p-2} n \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.240)} \\
+ & \sum_{l=1}^{2} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.241)} \\
+ & \sum_{l=3}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=1}^{s-p-l} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.242)} \\
+ & \sum_{l=3}^{s-p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=s-p-l+1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.243)} \\
- & \sum_{l=1}^{s-p} (-1)^l \left(\begin{array}{c} s-l-2 \\ p \end{array} \right) \sum_{n=s-p-1}^{p} n \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.244)} \\
+ & \sum_{l=1}^{s-p-1} (-1)^l \left(\begin{array}{c} s-l-2 \\ p-1 \end{array} \right) \sum_{n=s-p-1}^{p} \tilde{\Sigma}_l^{s,s-n-1} && \text{(B.245)}
\end{aligned}$$

$$- \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=p+1}^{s-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \tag{B.246}$$

$$+ \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=p+1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \Bigg]. \tag{B.247}$$

We now simplify this large expression to show it equals (B.73). We first coalesce sums with adjacent ranges. As before, it may help when reading this to check off each expression as it is used.

$$\begin{aligned} \text{(B.239) + (B.240)} \quad = \quad & - \sum_{l=2}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n \tilde{\Sigma}_l^{s,s-n-1} \\ & -(-1) \begin{pmatrix} s-3 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n \tilde{\Sigma}_1^{s,s-n-1} \\ & -(-1)^{s-p-1} \begin{pmatrix} p-1 \\ p \end{pmatrix} \sum_{n=1}^{s-p-2} n \tilde{\Sigma}_{s-p-1}^{s,s-n-1} \end{aligned} \tag{B.248}$$

$$\begin{aligned} \text{(B.242) + (B.243)} \quad = \quad & \sum_{l=3}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \\ & +(-1)^{s-p} \begin{pmatrix} p-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-2} \tilde{\Sigma}_{s-p}^{s,s-n-1} \\ = \quad & \sum_{l=3}^{s-p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \end{aligned} \tag{B.249}$$

$$\text{(B.241) + (B.249)} \quad = \quad \sum_{l=1}^{s-p} (-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-p-2} \tilde{\Sigma}_l^{s,s-n-1} \tag{B.250}$$

$$\begin{aligned} \text{(B.244) + (B.248)} \quad = \quad & - \sum_{l=1}^{s-p-1} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n \tilde{\Sigma}_l^{s,s-n-1} \\ & -(-1)^{s-p} \begin{pmatrix} p-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n \tilde{\Sigma}_{s-p}^{s,s-n-1} \end{aligned}$$

$$= -\sum_{l=1}^{s-p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{p} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.251)}$$

$$\begin{aligned} \text{(B.245) + (B.250)} &= \sum_{l=1}^{s-p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \\ &\quad +(-1)^{s-p} \begin{pmatrix} p-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_{s-p}^{s,s-n-1} \\ &= \sum_{l=1}^{s-p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.252)} \end{aligned}$$

$$\begin{aligned} \text{(B.246) + (B.251)} &= -\sum_{l=1}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \\ &\quad -(-1)^{s-p-1} \begin{pmatrix} p-1 \\ p \end{pmatrix} \sum_{n=1}^{p} n\tilde{\Sigma}_{s-p}^{s,s-n-1} \\ &= -\sum_{l=1}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n\tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.253)} \end{aligned}$$

$$\begin{aligned} \text{(B.247) + (B.252)} &= \sum_{l=1}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \\ &\quad +(-1)^{s-p-1} \begin{pmatrix} p-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{p} \tilde{\Sigma}_{s-p-1}^{s,s-n-1} \\ &= \sum_{l=1}^{s-p-2}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1} \qquad \text{(B.254)} \end{aligned}$$

The only remaining expressions that have not yet been used are (B.238), (B.253) and (B.254). Therefore we have established that the right side of (B.46) is their sum,

$$\begin{aligned} &(-1)^{s+p}\left[\begin{pmatrix} s-2 \\ p \end{pmatrix}(1-s) + \begin{pmatrix} s-2 \\ p-1 \end{pmatrix}\right. \\ &+ \left.\sum_{l=1}^{s-p-1}(-1)^l \begin{pmatrix} s-l-2 \\ p-1 \end{pmatrix} \sum_{n=1}^{s-l-1} \tilde{\Sigma}_l^{s,s-n-1}\right. \end{aligned}$$

$$- \sum_{l=1}^{s-p-2} (-1)^l \begin{pmatrix} s-l-2 \\ p \end{pmatrix} \sum_{n=1}^{s-l-1} n \tilde{\Sigma}_l^{s,s-n-1} \Bigg] . \tag{B.255}$$

This is exactly the same as (B.161), which we have already shown to be equal to (B.73). Therefore we have shown that (B.161) = (B.73) for $1 \leq p \leq s-3$, completing the proof of Theorem 5. □

Appendix C

Theorem 8

Theorem 8 *An explicit SEM (a GLM whose stability matrix has all but one eigenvalue at zero), for which $q(\phi) \neq 0$, with local discretisation error of order at least two, has a finite stability region.*

Remark. Although this is generally believed to be true for GLMs, we know of no proof in the existing literature, and so include this proof for SEMs. This proof is easily extended to GLMs for which $M(0)$ has a simple eigenvalue at 1. It is also possible to prove that, if an explicit GLM has an infinite stability region R, the connected component of R containing the origin consists of the origin only.

Proof. The stability matrix is

$$\begin{aligned} M(z) &= A_2 + zB_2(I - zB_1)^{-1}A_1 \\ &= A_2 + zB_2\left(\sum_{k=0}^{\infty}(-1)^k z^k B_1^k\right)A_1, \end{aligned}$$

provided the series converges. For an explicit method, however, B_1 is lower triangular with zeroes on the diagonal, so that $B_1^s = 0$ and the series is finite. Therefore,

$$M(z) = A_2 + zB_2\left(\sum_{k=0}^{s-1}(-1)^k z^k B_1^k\right)A_1,$$

and the entries of $M(z)$ are polynomials in z. The characteristic polynomial of $M(z)$ can be written as in (2.44):

$$\varphi(w,z) = \det(wI - M(z))$$

$$= \quad w^r - e_1(z)w^{r-1},$$

where the nonzero eigenvalue $e_1(z)$ is a polynomial in z since the entries of $M(z)$ are.

By Lemma 1 (Section 2.2),

$$e_1(z) = e^z + O(z^q), \quad q \geq 2.$$

Thus $e_1(z)$ is a non-constant polynomial, and it follows that

$$|e_1(z)| \to \infty \text{ as } |z| \to \infty.$$

Therefore, for $|z|$ sufficiently large, $|e_1(z)| > 1$, and the GLM has a finite stability region. $\square$

Appendix D

Theorem 9

Theorem 9 *A GLM is stable on the vector linear problem*

$$u'(t) = Cu(t), \quad C \in \mathbb{R}^{K\times K},$$

if it is stable on each of the scalar problems

$$u'(t) = \mu_i u(t), \quad i = 1, \ldots, K,$$

if C is diagonalisable, or

$$u'(t) = \mu_i u(t) + d_i(t), \quad i = 1, \ldots, K,$$

otherwise, where $\mu_1, \ldots, \mu_K$ are the eigenvalues of C and all the d_i's are analytic.

Proof. Write the vector method as

$$\begin{aligned} Y_n &= (A_1 \otimes I_K)y_n + h(B_1 \otimes I_K)F(Y_n), \\ y_{n+1} &= (A_2 \otimes I_K)y_n + h(B_2 \otimes I_K)F(Y_n). \end{aligned}$$

For the ODE $u'(t) = Cu(t)$, $F(Y_n) = (I_s \otimes C)Y_n$, so that the method becomes

$$\begin{aligned} Y_n &= (A_1 \otimes I_K)y_n + h(B_1 \otimes C)Y_n, \\ y_{n+1} &= (A_2 \otimes I_K)y_n + h(B_2 \otimes C)Y_n. \end{aligned}$$

First, assume that C is diagonalisable and let T be a matrix such that

$$TCT^{-1} = D = \begin{pmatrix} \mu_1 & & \\ & \ddots & \\ & & \mu_K \end{pmatrix}.$$

Using

$$\begin{aligned} z_n &= (I_r \otimes T)y_n, \\ z_{n+1} &= (I_r \otimes T)y_{n+1}, \\ Z_n &= (I_s \otimes T)Y_n, \end{aligned}$$

we can rewrite the method as

$$z_{n+1} = (A_2 \otimes I_K)z_n + h(B_2 \otimes D)Z_n.$$

Now define

$$e_{l,k} = (0, \ldots, 0, 1, 0, \ldots, 0) \in \mathbb{R}^l,$$

where the 1 occurs in the k-th component. Then $z_{n+1,k}$, the sub-vector of z_{n+1} consisting of the k-th component of each of the r block components in $\mathbb{R}^K$, is given by

$$\begin{aligned} z_{n+1,k} &= (I_r \otimes e_{K,k})z_{n+1} \\ &= (I_r \otimes e_{K,k})(A_2 \otimes I_K)z_n + h(I_r \otimes e_{K,k})(B_2 \otimes D)Z_n \\ &= (A_2 \otimes e_{K,k})z_n + h(B_2 \otimes e_{K,k}D)Z_n \\ &= (A_2 \otimes I_1)(I_r \otimes e_{K,k})z_n + h(B_2 \otimes \mu_k e_{K,k})Z_n \\ &= A_2 z_{n,k} + h\mu_k(B_2 \otimes I_1)(I_s \otimes e_{K,k})Z_n \\ &= A_2 z_{n,k} + h\mu_k B_2 Z_{n,k}. \end{aligned}$$

Similarly,

$$Z_{n,k} = A_1 z_{n,k} + h\mu_k B_1 Z_{n,k}.$$

Thus $z_{n+1,k}$ is the result of applying the method to the scalar problem $u' = \mu_k u$. Since the method is stable on each of these scalar problems,

$$\|z_{n+1}\| \leq \alpha \|z_0\|$$

for some constant α independent of n, so

$$\begin{aligned}
\|y_{n+1}\| &= \|(I_r \otimes T^{-1})z_{n+1}\| \\
&\leq \|I_r \otimes T^{-1}\|\|z_{n+1}\| \\
&\leq \|I_r \otimes T^{-1}\|\alpha\|z_0\| \\
&= \alpha\|I_r \otimes T^{-1}\|\|(I_r \otimes T)y_0\|, \\
&\leq \alpha\|I_r \otimes T^{-1}\|\|I_r \otimes T\|\|y_0\|,
\end{aligned}$$

which shows the method is stable on the vector problem too.

If C is not diagonalisable, consider the Jordan Canonical Form J of C. Without loss of generality, we may assume J consists of one Jordan block only (corresponding to μ, say), because the separate blocks decouple in a manner similar to the diagonalisable case considered above. Therefore assume

$$TCT^{-1} = J = \begin{pmatrix} \mu & & & \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & \mu \end{pmatrix}.$$

An argument similar to that for the diagonalisable case shows that

$$\begin{aligned}
Z_{n,k} &= A_1 z_{n,k} + h\mu_k B_1 Z_{n,k} + h B_1 Z_{n,k-1} \in \mathbb{R}^s, \\
z_{n+1,k} &= A_2 z_{n,k} + h\mu_k B_2 Z_{n,k} + h B_2 Z_{n,k-1} \in \mathbb{R}^r,
\end{aligned}$$

where the $Z_{n-1,k}$ terms are omitted for $k = 1$. Thus $z_{n+1,k}$ is the result of applying the original method to the scalar problem $u'(t) = \mu u(t) + d(t)$, where $d(t_n) = Z_{n,k-1}$. Since the scalar method is stable on such a problem for any analytic d, it follows, as in the diagonalisable case, that the vector method is also stable. □

Appendix E

Lemmas

The following lemmas are used in the proofs of Theorems 4 and 5.

E.1 Lemma 2

Lemma 2 *Let M_n be defined as in (B.19). Recall that the notation $\left[\begin{array}{c} i \\ j \end{array} \right]$ is being used as a scalar function of 2 variables, which will eventually be substituted for the binomial coefficient $\left(\begin{array}{c} i \\ j \end{array} \right)$. Then, for $n \geq 1$,*

$$M_n = \sum_{k=0}^{n-1} (-1)^{n-k} w^k \sum_{l=1}^{n-k} (-1)^l \left(\begin{array}{c} n-l \\ k \end{array} \right) \bar{\Sigma}_l^n, \tag{E.1}$$

where

$$\bar{\Sigma}_l^n = \begin{cases} \sum_{1 \leq i_1 < \cdots < i_{l-1} \leq n-1} \left[\begin{array}{c} i_1 \\ 1 \end{array} \right] \left[\begin{array}{c} i_2 \\ i_1+1 \end{array} \right] \cdots \left[\begin{array}{c} i_{l-1} \\ i_{l-2}+1 \end{array} \right] \left[\begin{array}{c} n \\ i_{l-1}+1 \end{array} \right], & l \geq 2 \\ \left[\begin{array}{c} n \\ 1 \end{array} \right], & l = 1. \end{cases} \tag{E.2}$$

Proof. Expanding M_n along the top row gives

$$M_n = \sum_{i=1}^{n} (-1)^{i-1} \left[\begin{array}{c} i \\ 1 \end{array} \right] M_{n,i}, \tag{E.3}$$

where $M_{n,i}$ is the determinant of the $(n-1)\times(n-1)$ matrix formed from M_n by crossing out the top row and the i-th column. Note $M_{n,i}$ is block upper-triangular and the top left block is upper triangular. So $M_{n,n}$ is upper triangular, and $M_{n,n} = (1-w)^{n-1}$. If $1 \le i < n$, then

$$M_{n,i} = (1-w)^{i-1} \begin{vmatrix} \begin{bmatrix} i+1 \\ i+1 \end{bmatrix} & \cdots & \cdots & \begin{bmatrix} n \\ i+1 \end{bmatrix} \\ 1-w & \begin{bmatrix} i+2 \\ i+2 \end{bmatrix} & \cdots & \begin{bmatrix} n \\ i+2 \end{bmatrix} \\ & \ddots & \ddots & \vdots \\ \bigcirc & & 1-w & \begin{bmatrix} n \\ n \end{bmatrix} \end{vmatrix},$$

where the matrix on the right side is $(n-i)\times(n-i)$. Now note that

$$M_{n-i} = \begin{vmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} & \cdots & \cdots & \begin{bmatrix} n-i \\ 1 \end{bmatrix} \\ 1-w & \begin{bmatrix} 2 \\ 2 \end{bmatrix} & \cdots & \begin{bmatrix} n-i \\ 2 \end{bmatrix} \\ & \ddots & & \vdots \\ \bigcirc & & 1-w & \begin{bmatrix} n-i \\ n-i \end{bmatrix} \end{vmatrix},$$

is also the determinant of a $(n-i)\times(n-i)$ matrix and that

$$M_{n,i} = (1-w)^{i-1} M_{n-i} \Big|_{\begin{bmatrix} u \\ v \end{bmatrix} \leftarrow \begin{bmatrix} u+i \\ v+i \end{bmatrix}} \tag{E.4}$$

for $i = 1, \ldots, n-1$.

Since $i \ge 1$ in (E.3) and (E.4), we can apply induction on (E.1). The base case, $n = 1$, states that

$$M_1 = (-1)^1 w^0 (-1)^1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} \bar{\Sigma}_1^1,$$

which follows immediately from (E.2) and (B.19). Induction then gives

$$M_{n-i} = \sum_{k=0}^{n-i-1} (-1)^{n-i-k} w^k \sum_{l=1}^{n-i-k} (-1)^l \begin{pmatrix} n-i-l \\ k \end{pmatrix} \bar{\Sigma}_l^{n-i}$$

for $i = 1, \ldots, n-1$. So

$$M_{n-i}\Big|_{\left[\begin{array}{c} u \\ v \end{array}\right] \leftarrow \left[\begin{array}{c} u+i \\ v+i \end{array}\right]} = \sum_{k=0}^{n-i-1} (-1)^{n-i-k} w^k \sum_{l=1}^{n-i-k} (-1)^l \left(\begin{array}{c} n-i-l \\ k \end{array} \right) \hat{\Sigma}_l^{n,i} \qquad \text{(E.5)}$$

for $i = 1, \ldots, n-1$, where

$$\hat{\Sigma}_k^{n,i} = \bar{\Sigma}_l^{n-i}\Big|_{\left[\begin{array}{c} u \\ v \end{array}\right] \leftarrow \left[\begin{array}{c} u+i \\ v+i \end{array}\right]}.$$

Using (E.2) to evaluate $\hat{\Sigma}_k^{n,i}$ gives

$$\begin{aligned} &\hat{\Sigma}_k^{n,i} \\ &= \begin{cases} \sum_{1 \le i_1 < \cdots < i_{l-1} \le n-i-1} \left[\begin{array}{c} i_1+i \\ i+1 \end{array}\right] \left[\begin{array}{c} i_2+i \\ i_1+i+1 \end{array}\right] \cdots \\ \qquad \left[\begin{array}{c} i_{l-1}+i \\ i_{l-2}+i+1 \end{array}\right] \left[\begin{array}{c} n \\ i_{l-1}+i+1 \end{array}\right], & l \ge 2 \\ \left[\begin{array}{c} n \\ i+1 \end{array}\right], & l = 1 \end{cases} \\ &= \begin{cases} \sum_{i+1 \le i_1 < \cdots < i_{l-1} \le n-1} \left[\begin{array}{c} i_1 \\ i+1 \end{array}\right] \left[\begin{array}{c} i_2 \\ i_1+1 \end{array}\right] \cdots \\ \qquad \left[\begin{array}{c} i_{l-1} \\ i_{l-2}+1 \end{array}\right] \left[\begin{array}{c} n \\ i_{l-1}+1 \end{array}\right], & l \ge 2 \\ \left[\begin{array}{c} n \\ i+1 \end{array}\right], & l = 1. \end{cases} \end{aligned} \qquad \text{(E.6)}$$

Now (E.4), (E.5) and (E.6) imply

$$M_{n,i} = (1-w)^{i-1} \sum_{k=0}^{n-i-1} (-1)^{n-i-k} w^k S_k^{n,i} \qquad \text{(E.7)}$$

for $i = 1, \ldots, n-1$, where

$$S_k^{n,i} = \sum_{l=1}^{n-i-k} (-1)^l \left(\begin{array}{c} n-i-l \\ k \end{array} \right) \hat{\Sigma}_l^{n,i}.$$

Then, from (E.3) and (E.7), we have

$$\begin{aligned}
M_n &= \sum_{i=1}^{n}(-1)^{i-1}\left[\begin{array}{c} i \\ 1 \end{array}\right] M_{n,i} \\
&= \sum_{i=1}^{n-1}(-1)^{i-1}\left[\begin{array}{c} i \\ 1 \end{array}\right](1-w)^{i-1}\sum_{k=0}^{n-i-1}(-1)^{n-i-k}S_k^{n,i}w^k \\
&\quad +(-1)^{n-1}\left[\begin{array}{c} n \\ 1 \end{array}\right](1-w)^{n-1} \\
&= (-1)^{n-1}\left[\sum_{i=1}^{n-1}\left[\begin{array}{c} i \\ 1 \end{array}\right](1-w)^{i-1}\sum_{k=0}^{n-i-1}(-1)^{k}S_k^{n,i}w^k + \left[\begin{array}{c} n \\ 1 \end{array}\right](1-w)^{n-1}\right] \\
&= (-1)^{n-1}\left[\sum_{i=1}^{n-1}\left[\begin{array}{c} i \\ 1 \end{array}\right](1-w)^{i-1}\sum_{k=0}^{n-i-1}(-1)^{k}\sum_{l=1}^{n-i-k}(-1)^{l}\left(\begin{array}{c} n-i-l \\ k \end{array}\right)\hat{\Sigma}_l^{n,i}w^k\right. \\
&\qquad \left. + \left[\begin{array}{c} n \\ 1 \end{array}\right](1-w)^{n-1}\right]. \qquad \text{(E.8)}
\end{aligned}$$

We must now show that (E.8) agrees with (E.1). Since (E.8) contains $\hat{\Sigma}_l^{n,i}$ and (E.1) is written in terms of $\bar{\Sigma}_l^n$, we need to find a relationship between these quantities. From (E.6),

$$\hat{\Sigma}_l^{n,i} = \begin{cases} \sum_{i+1\le i_1<\cdots<i_{l-1}\le n-1}\left[\begin{array}{c} i_1 \\ i+1 \end{array}\right]\left[\begin{array}{c} i_2 \\ i_1+1 \end{array}\right]\cdots\left[\begin{array}{c} i_{l-1} \\ i_{l-2}+1 \end{array}\right]\left[\begin{array}{c} n \\ i_{l-1}+1 \end{array}\right], & l\ge 2 \\ \left[\begin{array}{c} n \\ i+1 \end{array}\right], & l=1, \end{cases}$$

and, for $l \ge 1$, we see from (E.2) that

$$\begin{aligned}
\bar{\Sigma}_{l+1}^n &= \sum_{1\le i_1<\cdots<i_l\le n-1}\left[\begin{array}{c} i_1 \\ 1 \end{array}\right]\left[\begin{array}{c} i_2 \\ i_1+1 \end{array}\right]\cdots\left[\begin{array}{c} i_l \\ i_{l-1}+1 \end{array}\right]\left[\begin{array}{c} n \\ i_l+1 \end{array}\right] \\
&= \sum_{i_1=1}^{n-l}\left[\begin{array}{c} i_1 \\ 1 \end{array}\right]\sum_{i_1<i_2<\cdots<i_l\le n-1}\left[\begin{array}{c} i_2 \\ i_1+1 \end{array}\right]\cdots\left[\begin{array}{c} n \\ i_l+1 \end{array}\right] \\
&= \sum_{i=1}^{n-l}\left[\begin{array}{c} i \\ 1 \end{array}\right]\sum_{i+1\le i_1<\cdots<i_{l-1}\le n-1}\left[\begin{array}{c} i_1 \\ i+1 \end{array}\right]\cdots\left[\begin{array}{c} n \\ i_{l-1}+1 \end{array}\right] \\
&= \sum_{i=1}^{n-l}\left[\begin{array}{c} i \\ 1 \end{array}\right]\hat{\Sigma}_l^{n,i}.
\end{aligned}$$

Thus we have that

$$\bar{\Sigma}_l^n = \sum_{i=1}^{n-l+1} \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_{l-1}^{n,i} \tag{E.9}$$

for $l \geq 2$, while, for $l = 1$,

$$\bar{\Sigma}_1^n = \left[\begin{array}{c} n \\ 1 \end{array} \right] \quad \text{and} \quad \hat{\Sigma}_1^{n,i} = \left[\begin{array}{c} n \\ i+1 \end{array} \right].$$

We now return to showing that (E.1) equals (E.8). Using (E.9) gives

$$\begin{aligned}
\text{(E.1)} &= \sum_{k=0}^{n-1} (-1)^{n-k} w^k \left[\sum_{l=2}^{n-k} (-1)^l \left(\begin{array}{c} n-l \\ k \end{array} \right) \sum_{i=1}^{n-l+1} \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_{l-1}^{n,i} - \left(\begin{array}{c} n-1 \\ k \end{array} \right) \left[\begin{array}{c} n \\ 1 \end{array} \right] \right] \\
&= \sum_{k=0}^{n-1} (-1)^{n-k} w^k \left[- \sum_{l=1}^{n-k-1} (-1)^l \left(\begin{array}{c} n-l-1 \\ k \end{array} \right) \sum_{i=1}^{n-l} \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} - \left(\begin{array}{c} n-1 \\ k \end{array} \right) \left[\begin{array}{c} n \\ 1 \end{array} \right] \right] \\
&= (-1)^{n-1} \sum_{k=0}^{n-1} (-1)^k w^k \left[\sum_{l=1}^{n-k-1} (-1)^l \left(\begin{array}{c} n-l-1 \\ k \end{array} \right) \sum_{i=1}^{n-l} \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \right. \\
&\qquad \left. + \left(\begin{array}{c} n-1 \\ k \end{array} \right) \left[\begin{array}{c} n \\ 1 \end{array} \right] \right] \\
&= (-1)^{n-1} \left[\sum_{k=0}^{n-1} \sum_{l=1}^{n-k-1} \sum_{i=1}^{n-l} (-1)^{k+l} w^k \left(\begin{array}{c} n-l-1 \\ k \end{array} \right) \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \right. \\
&\qquad \left. + \left[\begin{array}{c} n \\ 1 \end{array} \right] \sum_{k=0}^{n-1} (-1)^k \left(\begin{array}{c} n-1 \\ k \end{array} \right) w^k \right].
\end{aligned}$$

On the other hand,

$$\begin{aligned}
\text{(E.8)} &= (-1)^{n-1} \left[\sum_{i=1}^{n-1} \left[\begin{array}{c} i \\ 1 \end{array} \right] (1-w)^{i-1} \sum_{k=0}^{n-i-1} (-1)^k \sum_{l=1}^{n-i-k} (-1)^l \left(\begin{array}{c} n-i-l \\ k \end{array} \right) \hat{\Sigma}_l^{n,i} w^k \right. \\
&\qquad \left. + \left[\begin{array}{c} n \\ 1 \end{array} \right] (1-w)^{n-1} \right].
\end{aligned}$$

Comparing these expressions, we see that they both begin with the factor $(-1)^{n-1}$, so we can ignore that part. Also, the term after the $+$ sign in both expressions agrees, so that too

can be ignored. Therefore we must prove that

$$\sum_{i=1}^{n-1} \left[\begin{array}{c} i \\ 1 \end{array} \right] (1-w)^{i-1} \sum_{k=0}^{n-i-1} (-1)^k \sum_{l=1}^{n-i-k} (-1)^l \left(\begin{array}{c} n-i-l \\ k \end{array} \right) \hat{\Sigma}_l^{n,i} w^k \tag{E.10}$$

$$= \sum_{k=0}^{n-1} \sum_{l=1}^{n-k-1} \sum_{i=1}^{n-l} (-1)^{k+l} w^k \left(\begin{array}{c} n-l-1 \\ k \end{array} \right) \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i}. \tag{E.11}$$

Expanding the $(1-w)^{i-1}$ in (E.10) gives

$$\sum_{i=1}^{n-1} \left[\begin{array}{c} i \\ 1 \end{array} \right] \sum_{m=0}^{i-1} (-1)^m \left(\begin{array}{c} i-1 \\ m \end{array} \right) w^m \sum_{k=0}^{n-i-1} (-1)^k \sum_{l=1}^{n-i-k} (-1)^l \left(\begin{array}{c} n-i-l \\ k \end{array} \right) \hat{\Sigma}_l^{n,i} w^k$$

$$= \sum_{i=1}^{n-1} \sum_{m=0}^{i-1} \sum_{k=0}^{n-i-1} \sum_{l=1}^{n-i-k} (-1)^{k+l+m} \left[\begin{array}{c} i \\ 1 \end{array} \right] \left(\begin{array}{c} i-1 \\ m \end{array} \right) \left(\begin{array}{c} n-i-l \\ k \end{array} \right) \hat{\Sigma}_l^{n,i} w^{k+m}. \tag{E.12}$$

Now, to compare (E.12) and (E.11), express (E.12) as a sum of powers of w. The part of (E.12) after the i sum has the form

$$\sum_{k=0}^{n-i-1} \sum_{m=0}^{i-1} \alpha(m,k) w^{m+k}. \tag{E.13}$$

Let $p = m + k$, so that $k = p - m$. Then $0 \leq k \leq n-i-1$ and $0 \leq m \leq i-1 \Rightarrow 0 \leq p \leq n-2$. Therefore, we want to write (E.13) as

$$\sum_{p=0}^{n-2} \beta_p w^p,$$

for suitable β_p's. Given p, we find the restrictions on m: $0 \leq m \leq i-1$ and $k = p-m$ with $0 \leq k \leq n-i-1 \Rightarrow p-n+i+1 \leq m \leq p \Rightarrow \max\{0, p-n+i+1\} \leq m \leq \min\{i-1, p\}$. Therefore,

$$\text{(E.13)} = \sum_{p=0}^{n-2} \sum_{m=\max\{0,p-n+i-1\}}^{\min\{i-1,p\}} \alpha(m, p-m) w^p.$$

Applying this to (E.12) gives

$$\sum_{i=1}^{n-1} \sum_{p=0}^{n-2} \sum_{m=\max\{0,p-n+i-1\}}^{\min\{i-1,p\}} \sum_{l=1}^{n-i-p+m} (-1)^{p+l} \left[\begin{array}{c} i \\ 1 \end{array} \right] \left(\begin{array}{c} i-1 \\ m \end{array} \right) \left(\begin{array}{c} n-i-l \\ p-m \end{array} \right) \hat{\Sigma}_l^{n,i} w^p,$$

which, upon exchanging the first two sums, becomes

$$\sum_{p=0}^{n-2}\sum_{i=1}^{n-1}\sum_{m=\max\{0,p-n+i-1\}}^{\min\{i-1,p\}}\sum_{l=1}^{n-i-p+m}(-1)^{p+l}w^p\begin{bmatrix} i \\ 1 \end{bmatrix}\begin{pmatrix} i-1 \\ m \end{pmatrix}\begin{pmatrix} n-i-l \\ p-m \end{pmatrix}\hat{\Sigma}_l^{n,i}. \tag{E.14}$$

In (E.11), we may take the first sum from $k=0$ to $n-2$, since for $k=n-1$ the second sum is empty. This, as well as replacing k by p, gives

$$\text{(E.11)} = \sum_{p=0}^{n-2}\sum_{l=1}^{n-p-1}\sum_{i=1}^{n-l}(-1)^{p+l}w^p\begin{bmatrix} i \\ 1 \end{bmatrix}\begin{pmatrix} n-l-1 \\ p \end{pmatrix}\hat{\Sigma}_l^{n,i}. \tag{E.15}$$

We must show (E.14) and (E.15) are equal. First, we eliminate the $\max$ and $\min$ occurring in (E.14) by considering several cases. We have

$$\begin{aligned} 0 \geq p-n+i-1 &\iff i \leq n-p+1, \\ i-1 \leq p &\iff i \leq p+1. \end{aligned}$$

Thus,

$$\begin{aligned} 1 \leq i \leq n-p &\implies \max\{0,p-n+i-1\} = 0, \\ n-p+1 \leq i \leq n-1 &\implies \max\{0,p-n+i-1\} = p-n+i-1, \\ 1 \leq i \leq p &\implies \min\{i-1,p\} = i-1, \\ p+1 \leq i \leq n-1 &\implies \min\{i-1,p\} = p. \end{aligned}$$

When $p \leq n/2$, $p \leq n-p$, so that

$$\begin{aligned} 1 \leq i \leq p &\implies \max\{0,p-n+i-1\} = 0, && \min\{i-1,p\} = i-1, \\ p+1 \leq i \leq n-p &\implies \max\{0,p-n+i-1\} = 0, && \min\{i-1,p\} = p, \\ n-p+1 \leq i \leq n-1 &\implies \max\{0,p-n+i-1\} = p-n+i-1, && \min\{i-1,p\} = p, \end{aligned}$$

and when $n/2 \leq p$, $p \geq n-p$, so that

$$\begin{aligned} 1 \leq i \leq n-p &\implies \max\{0,p-n+i-1\} = 0, && \min\{i-1,p\} = i-1, \\ n-p+1 \leq i \leq p &\implies \max\{0,p-n+i-1\} = p-n+i-1, && \min\{i-1,p\} = i-1, \\ p+1 \leq i \leq n-1 &\implies \max\{0,p-n+i-1\} = p-n+i-1, && \min\{i-1,p\} = p. \end{aligned}$$

Using the relations above and omitting the arguments to the sums for brevity, we can rewrite (E.14) as

$$
\begin{aligned}
\text{(E.14)} \quad = \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=1}^{p} \sum_{m=0}^{i-1} \sum_{l=1}^{n-i-p+m} && \text{(E.16)} \\
+ \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=p+1}^{n-p} \sum_{m=0}^{p} \sum_{l=1}^{n-i-p+m} && \text{(E.17)} \\
+ \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=n-p+1}^{n-1} \sum_{m=p-n+i-1}^{p} \sum_{l=1}^{n-i-p+m} && \text{(E.18)} \\
+ \quad & \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=1}^{n-p} \sum_{m=0}^{i-1} \sum_{l=1}^{n-i-p+m} && \text{(E.19)} \\
+ \quad & \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=n-p+1}^{p} \sum_{m=p-n+i-1}^{i-1} \sum_{l=1}^{n-i-p+m} && \text{(E.20)} \\
+ \quad & \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=p+1}^{n-1} \sum_{m=p-n+i-1}^{p} \sum_{l=1}^{n-i-p+m} . && \text{(E.21)}
\end{aligned}
$$

Comparing (E.15) with (E.16) to (E.21), we find that the m and l sums in (E.16) to (E.21) must be exchanged. We first derive a general method for switching sums of this form and then apply it to each case.

Consider the sums

$$\sum_{m=a}^{b} \sum_{l=1}^{c+m},$$

where a, b, c are free. When $m = a$, $l = 1, \ldots, c + a$. When $m = b$, $l = 1, \ldots, c + b$. Thus, for $l = 1, \ldots, c + a$, $m = a, \ldots, b$, and for $l = c + a, \ldots, c + b$, $m = l - c, \ldots, b$. Therefore,

$$\sum_{m=a}^{b} \sum_{l=1}^{c+m} = \sum_{l=1}^{c+a} \sum_{m=a}^{b} + \sum_{l=c+a+1}^{c+b} \sum_{m=l-c}^{b} . \tag{E.22}$$

Applying (E.22) to (E.20) gives $a = p - n + i + 1$ (because the l sum in (E.20) is empty when $m < p - n + i + 1$), $b = i - 1$ and $c = n - i - p \Rightarrow c + a = 1$ and $c + b = n - p - 1$. Therefore,

$$
\begin{aligned}
\sum_{m=p-n+i-1}^{i-1} \sum_{l=1}^{n-i-p+m} \quad = \quad & \sum_{m=p-n+i+1}^{i-1} \sum_{l=1}^{n-i-p+m} \\
= \quad & \sum_{l=1}^{1} \sum_{m=p-n+i+1}^{i-1} + \sum_{l=2}^{n-p-1} \sum_{m=l-n+i+p}^{i-1}
\end{aligned}
$$

$$= \sum_{l=1}^{n-p-1} \sum_{m=l-n+i+p}^{i-1} . \tag{E.23}$$

Applying (E.22) to (E.16) and (E.19) gives $a = 0$, $b = i-1$, and $c = n-i-p \Rightarrow c+a = n-i-p$ and $c+b = n-p-1$. Therefore,

$$\sum_{m=0}^{i-1} \sum_{l=1}^{n-i-p+m} = \sum_{l=1}^{n-i-p} \sum_{m=0}^{i-1} + \sum_{l=n-i-p+1}^{n-p-1} \sum_{m=l-n+i+p}^{i-1} . \tag{E.24}$$

Applying (E.22) to (E.18) and (E.21) gives $a = p-n+i+1$ (because the l sum in (E.18) and (E.21) is empty when $m < p-n+i+1$), $b = p$ and $c = n-i-p \Rightarrow c+a = 1$ and $c+b = n-i$. Therefore,

$$\begin{aligned} \sum_{m=p-n+i-1}^{p} \sum_{l=1}^{n-i-p+m} &= \sum_{m=p-n+i+1}^{p} \sum_{l=1}^{n-i-p+m} \\ &= \sum_{l=1}^{1} \sum_{m=p-n+i+1}^{p} + \sum_{l=2}^{n-i} \sum_{m=l-n+i+p}^{p} \\ &= \sum_{l=1}^{n-i} \sum_{m=l-n+i+p}^{p} . \end{aligned} \tag{E.25}$$

Applying (E.22) to (E.17) gives $a = 0$, $b = p$, and $c = n-i-p \Rightarrow c+a = n-i-p$ and $c+b = n-i$. Therefore,

$$\sum_{m=0}^{p} \sum_{l=1}^{n-i-p+m} = \sum_{l=1}^{n-i-p} \sum_{m=0}^{p} + \sum_{l=n-i-p+1}^{n-i} \sum_{m=l-n+i+p}^{p} . \tag{E.26}$$

Now, using (E.23) to (E.26) in (E.16) to (E.21) and inserting the arguments to the sums from (E.14), we find that (E.14) =

$$\sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=1}^{p} \sum_{l=1}^{n-i-p} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=0}^{i-1} \binom{i-1}{m} \binom{n-i-l}{p-m} \tag{E.27}$$

$$+ \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=1}^{p} \sum_{l=n-i-p+1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=l-n+i+p}^{i-1} \binom{i-1}{m} \binom{n-i-l}{p-m} \tag{E.28}$$

$$+ \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=p+1}^{n-p} \sum_{l=1}^{n-i-p} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=0}^{p} \binom{i-1}{m} \binom{n-i-l}{p-m} \tag{E.29}$$

$$+ \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=p+1}^{n-p} \sum_{l=n-i-p+1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=l-n+i+p}^{p} \binom{i-1}{m} \binom{n-i-l}{p-m} \tag{E.30}$$

$$+ \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=n-p+1}^{n-1} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=l-n+i+p}^{p} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix} \tag{E.31}$$

$$+ \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=1}^{n-p} \sum_{l=1}^{n-i-p} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=0}^{i-1} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix} \tag{E.32}$$

$$+ \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=1}^{n-p} \sum_{l=n-i-p+1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=l-n+i+p}^{i-1} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix} \tag{E.33}$$

$$+ \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=n-p+1}^{p} \sum_{l=1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=l-n+i+p}^{i-1} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix} \tag{E.34}$$

$$+ \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=p+1}^{n-1} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \sum_{m=l-n+i+p}^{p} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix}. \tag{E.35}$$

We now simplify the m-sums in the above expression. Using (F.5) in (E.27) and (E.32), we have

$$\sum_{m=0}^{i-1} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix} = \begin{pmatrix} n-l-1 \\ p \end{pmatrix}.$$

In (E.28), (E.33) and (E.34), we have

$$\begin{aligned}
\sum_{m=l-n+i+p}^{i-1} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} n-i-l \\ p-m \end{pmatrix} &= \sum_{m=u}^{i-1} \begin{pmatrix} i-1 \\ m \end{pmatrix} \begin{pmatrix} p-u \\ p-m \end{pmatrix}, \quad u = l-n+i+p \\
&= \sum_{k=0}^{i-1-u} \begin{pmatrix} i-1 \\ k+u \end{pmatrix} \begin{pmatrix} p-u \\ p-(k+u) \end{pmatrix} \\
&= \sum_{k=0}^{a} \begin{pmatrix} a+u \\ k+u \end{pmatrix} \begin{pmatrix} p-u \\ p-k-u \end{pmatrix}, \quad a = i-1-u \\
&= \sum_{k=0}^{a} \begin{pmatrix} a+u \\ a-k \end{pmatrix} \begin{pmatrix} p-u \\ k \end{pmatrix} \\
&= \begin{pmatrix} a+p \\ a \end{pmatrix} \\
&= \begin{pmatrix} n-l-1 \\ p \end{pmatrix}
\end{aligned}$$

by (F.4). In (E.29), using (F.4), we have

$$\sum_{m=0}^{p} \left(\begin{array}{c} i-1 \\ m \end{array} \right) \left(\begin{array}{c} n-i-l \\ p-m \end{array} \right) = \left(\begin{array}{c} n-l-1 \\ p \end{array} \right).$$

In (E.31) and (E.35), we have

$$\begin{aligned}
\sum_{m=l-n+i+p}^{p} \left(\begin{array}{c} i-1 \\ m \end{array} \right) \left(\begin{array}{c} n-i-l \\ p-m \end{array} \right) &= \sum_{m=p-u}^{p} \left(\begin{array}{c} i-1 \\ m \end{array} \right) \left(\begin{array}{c} u \\ p-m \end{array} \right), \quad \text{for } u = n-i-l \\
&= \sum_{k=0}^{u} \left(\begin{array}{c} i-1 \\ k+p-u \end{array} \right) \left(\begin{array}{c} u \\ u-k \end{array} \right) \\
&= \sum_{k=0}^{u} \left(\begin{array}{c} i-1 \\ k+p-u \end{array} \right) \left(\begin{array}{c} u \\ k \end{array} \right) \\
&= \left(\begin{array}{c} u+i-1 \\ u+p-u \end{array} \right) \\
&= \left(\begin{array}{c} n-l-1 \\ p \end{array} \right)
\end{aligned}$$

by (F.6). Replacing the m-sums in (E.27) to (E.35) with the above expressions gives

$$\begin{aligned}
\text{(E.14)} \quad = \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=1}^{p} \sum_{l=1}^{n-i-p} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right) \\
+ \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=1}^{p} \sum_{l=n-i-p+1}^{n-p-1} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right) \\
+ \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=p+1}^{n-p} \sum_{l=1}^{n-i-p} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right) \\
+ \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=p+1}^{n-p} \sum_{l=n-i-p+1}^{n-i} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right) \\
+ \quad & \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{i=n-p+1}^{n-1} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right) \\
+ \quad & \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=1}^{n-p} \sum_{l=1}^{n-i-p} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right) \\
+ \quad & \sum_{p=\lfloor n/2 \rfloor+1}^{n-2} \sum_{i=1}^{n-p} \sum_{l=n-i-p+1}^{n-p-1} (-1)^{p+l} w^p \left[\begin{array}{c} i \\ 1 \end{array} \right] \hat{\Sigma}_l^{n,i} \left(\begin{array}{c} n-l-1 \\ p \end{array} \right)
\end{aligned}$$

$$+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2} \sum_{i=n-p+1}^{p} \sum_{l=1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix}$$

$$+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2} \sum_{i=p+1}^{n-1} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix},$$

and, by coalescing adjacent sums, we get

$$\text{(E.14)} \quad = \sum_{p=0}^{\lfloor n/2\rfloor} \sum_{i=1}^{p} \sum_{l=1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix} \tag{E.36}$$

$$+ \sum_{p=0}^{\lfloor n/2\rfloor} \sum_{i=p+1}^{n-p} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix} \tag{E.37}$$

$$+ \sum_{p=0}^{\lfloor n/2\rfloor} \sum_{i=n-p+1}^{n-1} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix} \tag{E.38}$$

$$+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2} \sum_{i=1}^{n-p} \sum_{l=1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix} \tag{E.39}$$

$$+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2} \sum_{i=n-p+1}^{p} \sum_{l=1}^{n-p-1} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix} \tag{E.40}$$

$$+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2} \sum_{i=p+1}^{n-1} \sum_{l=1}^{n-i} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \hat{\Sigma}_l^{n,i} \begin{pmatrix} n-l-1 \\ p \end{pmatrix}. \tag{E.41}$$

Now switch the i and l sums in (E.36) to (E.41). The ones in (E.36), (E.39) and (E.40) are independent, so they just exchange. The remaining ones have the form

$$\sum_{i=a}^{b} \sum_{l=1}^{c-i},$$

where a, b, c are free. When $i = a$, $l = 1, \ldots, c-a$. When $i = b$, $l = 1, \ldots, c-b$. Thus, for $l = 1, \ldots, c-b$, $i = a, \ldots, b$, and, for $l = c-b, \ldots, c-a$, $i = a, \ldots, c-l$. Therefore,

$$\sum_{i=a}^{b} \sum_{l=1}^{c-i} = \sum_{l=1}^{c-b} \sum_{i=a}^{b} + \sum_{l=c-b+1}^{c-a} \sum_{i=a}^{c-l}. \tag{E.42}$$

Applying (E.42) to (E.37) gives $a = p+1$, $b = n-p$ and $c = n \Rightarrow c-b = p$ and $c-a = n-p-1$. Therefore,

$$\sum_{i=p+1}^{n-p}\sum_{l=1}^{n-i} = \sum_{l=1}^{p}\sum_{i=p+1}^{n-p} + \sum_{l=p+1}^{n-p-1}\sum_{i=p+1}^{n-l}. \tag{E.43}$$

Applying (E.42) to (E.38) gives $a = n-p+1$, $b = n-1$ and $c = n \Rightarrow c-b = 1$ and $c-a = p-1$. Therefore,

$$\begin{aligned}\sum_{i=n-p+1}^{n-1}\sum_{l=1}^{n-i} &= \sum_{l=1}^{1}\sum_{i=n-p+1}^{n-1} + \sum_{l=2}^{p-1}\sum_{i=n-p+1}^{n-l} \\ &= \sum_{l=1}^{p-1}\sum_{i=n-p+1}^{n-l}.\end{aligned} \tag{E.44}$$

Applying (E.42) to (E.41) gives $a = p+1$, $b = n-1$ and $c = n \Rightarrow c-b = 1$ and $c-a = n-p-1$. Therefore,

$$\begin{aligned}\sum_{i=p+1}^{n-1}\sum_{l=1}^{n-i} &= \sum_{l=1}^{1}\sum_{i=p+1}^{n-1} + \sum_{l=2}^{n-p-1}\sum_{i=p+1}^{n-l} \\ &= \sum_{l=1}^{n-p-1}\sum_{i=p+1}^{n-l}.\end{aligned} \tag{E.45}$$

Now, switching the i and l sums in (E.36), (E.39) and (E.40) and applying (E.43), (E.44) and (E.45) to (E.37), (E.38) and (E.41), respectively, as well as removing the sum arguments for clarity, gives

$$\begin{aligned}\text{(E.14)} &= \sum_{p=0}^{\lfloor n/2\rfloor}\sum_{l=1}^{n-p-1}\sum_{i=1}^{p} + \sum_{p=0}^{\lfloor n/2\rfloor}\sum_{l=1}^{p}\sum_{i=p+1}^{n-p} + \sum_{p=0}^{\lfloor n/2\rfloor}\sum_{l=p+1}^{n-p-1}\sum_{i=p+1}^{n-l} + \sum_{p=0}^{\lfloor n/2\rfloor}\sum_{l=1}^{p-1}\sum_{i=n-p+1}^{n-l} \\ &+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2}\sum_{l=1}^{n-p-1}\sum_{i=1}^{n-p} + \sum_{p=\lfloor n/2\rfloor+1}^{n-2}\sum_{l=1}^{n-p-1}\sum_{i=n-p+1}^{p} + \sum_{p=\lfloor n/2\rfloor+1}^{n-2}\sum_{l=1}^{n-p-1}\sum_{i=p+1}^{n-l} \\ &= \sum_{p=0}^{\lfloor n/2\rfloor}\left[\sum_{l=1}^{n-p-1}\sum_{i=1}^{p} + \sum_{l=1}^{p}\sum_{i=p+1}^{n-p} + \sum_{l=p+1}^{n-p-1}\sum_{i=p+1}^{n-l} + \sum_{l=1}^{p-1}\sum_{i=n-p+1}^{n-l}\right] \\ &+ \sum_{p=\lfloor n/2\rfloor+1}^{n-2}\left[\sum_{l=1}^{n-p-1}\sum_{i=1}^{n-p} + \sum_{l=1}^{n-p-1}\sum_{i=n-p+1}^{p} + \sum_{l=1}^{n-p-1}\sum_{i=p+1}^{n-l}\right] \\ &= \sum_{p=0}^{\lfloor n/2\rfloor}\left[\sum_{l=1}^{n-p-1}\sum_{i=1}^{p} + \sum_{l=1}^{p}\sum_{i=p+1}^{n-p} + \sum_{l=p+1}^{n-p-1}\sum_{i=p+1}^{n-l} + \sum_{l=1}^{p-1}\sum_{i=n-p+1}^{n-l}\right]\end{aligned}$$

$$+ \sum_{p=\lfloor n/2 \rfloor + 1}^{n-2} \sum_{l=1}^{n-p-1} \sum_{i=1}^{n-l}. \tag{E.46}$$

It turns out that the arguments to both of the p sums are in fact equal. That is,

$$\begin{aligned}
& \sum_{l=1}^{n-p-1} \sum_{i=1}^{p} + \sum_{l=1}^{p} \sum_{i=p+1}^{n-p} + \sum_{l=p+1}^{n-p-1} \sum_{i=p+1}^{n-l} + \sum_{l=1}^{p-1} \sum_{i=n-p+1}^{n-l} \\
= & \sum_{l=1}^{n-p-1} \sum_{i=1}^{p} + \sum_{l=1}^{p} \sum_{i=p+1}^{n-p} + \sum_{l=p+1}^{n-p-1} \sum_{i=p+1}^{n-l} + \sum_{l=1}^{p} \sum_{i=n-p+1}^{n-l} \\
= & \sum_{l=1}^{n-p-1} \sum_{i=1}^{p} + \sum_{l=p+1}^{n-p-1} \sum_{i=p+1}^{n-l} + \sum_{l=1}^{p} \sum_{i=p+1}^{n-l}, \quad \text{since } l \leq p \Rightarrow n-l \geq n-p, \\
= & \sum_{l=1}^{n-p-1} \sum_{i=1}^{p} + \sum_{l=1}^{n-p-1} \sum_{i=p+1}^{n-l}, \quad \text{since } l \leq n-p-1 \Rightarrow n-l \geq p+1, \\
= & \sum_{l=1}^{n-p-1} \sum_{i=1}^{n-l}.
\end{aligned}$$

Using this in (E.46), we finally have that

$$\text{(E.14)} = \sum_{p=0}^{\lfloor n/2 \rfloor} \sum_{l=1}^{n-p-1} \sum_{i=1}^{n-l} + \sum_{p=\lfloor n/2 \rfloor + 1}^{n-2} \sum_{l=1}^{n-p-1} \sum_{i=1}^{n-l} = \sum_{p=0}^{n-2} \sum_{l=1}^{n-p-1} \sum_{i=1}^{n-l}.$$

Putting the sum arguments from (E.36) to (E.41) back in, we get

$$\text{(E.14)} = \sum_{p=0}^{n-2} \sum_{l=1}^{n-p-1} \sum_{i=1}^{n-l} (-1)^{p+l} w^p \begin{bmatrix} i \\ 1 \end{bmatrix} \begin{pmatrix} n-l-1 \\ p \end{pmatrix} \hat{\Sigma}_l^{n,i},$$

which equals (E.15), completing the proof of Lemma 2. □

E.2 Lemma 3

Lemma 3

$$\sum_{l=1}^{n} (-1)^l \tilde{\Sigma}_l^{s,n} = (-1)^n \begin{pmatrix} s \\ n \end{pmatrix}, \quad \textit{for } n = 1, \ldots, s-1,\ s \geq 1. \tag{E.47}$$

Proof. For $l \geq 2$, we see from (B.22) that

$$\begin{aligned}
\tilde{\Sigma}_l^{s,n} &= \sum_{s-n+1\leq i_1<\cdots<i_{l-1}\leq s-1} \binom{i_1}{s-n}\binom{i_2}{i_1}\cdots\binom{i_{l-1}}{i_{l-2}}\binom{s}{i_{l-1}} \\
&= \sum_{i_1=s-n+1}^{s-l+1} \binom{i_1}{s-n} \sum_{i_1<i_2<\cdots<i_{l-1}\leq s-1} \binom{i_2}{i_1}\cdots\binom{i_{l-1}}{i_{l-2}}\binom{s}{i_{l-1}} \\
&= \sum_{k=s-n+1}^{s-l+1} \binom{k}{s-n} \sum_{k<i_1<i_2\cdots<i_{l-2}\leq s-1} \binom{i_1}{k}\binom{i_2}{i_1}\cdots\binom{i_{l-2}}{i_{l-3}}\binom{s}{i_{l-2}} \\
&= \sum_{k=s-n+2}^{s-l+2} \binom{k-1}{s-n} \sum_{k\leq i_1<i_2\cdots<i_{l-2}\leq s-1} \binom{i_1}{k-1}\binom{i_2}{i_1}\cdots\binom{i_{l-2}}{i_{l-3}}\binom{s}{i_{l-2}} \\
&= \sum_{N=l-1}^{n-1} \binom{s-N}{s-n} \tilde{\Sigma}_{l-1}^{s,N}
\end{aligned}$$

while, for $l = 1$,

$$\tilde{\Sigma}_1^{s,n} = \binom{s}{n}.$$

So the left side of (E.47) reduces to

$$-\binom{s}{n} + \sum_{l=2}^{n}(-1)^l \sum_{N=l-1}^{n-1} \binom{s-N}{s-n} \tilde{\Sigma}_{l-1}^{s,N}. \tag{E.48}$$

Now switch the l and N sums. For $l = 2$, $N = 1, \ldots, n-1$. For $l = n$, $N = n-1$. Thus $N = 1, \ldots, n-1$ and $l = 2, \ldots, N+1$. Thus, continuing from (E.48), we find that the left side of (E.47) becomes

$$\begin{aligned}
&-\binom{s}{n} + \sum_{N=1}^{n-1}\sum_{l=2}^{N+1}(-1)^l \binom{s-N}{s-n} \tilde{\Sigma}_{l-1}^{s,N} \\
&= -\binom{s}{n} - \sum_{N=1}^{n-1} \binom{s-N}{s-n} \sum_{l=1}^{N}(-1)^l \tilde{\Sigma}_l^{s,N} \\
&= -\binom{s}{n} - \sum_{N=1}^{n-1} \binom{s-N}{s-n} (-1)^N \binom{s}{N} \quad \text{by induction, since } N \leq n-1 \\
&= -\sum_{N=0}^{n-1} \binom{s-N}{s-n} (-1)^N \binom{s}{N}
\end{aligned}$$

$$= \quad -(-1)^{n-1} \begin{pmatrix} s \\ n \end{pmatrix} \quad \text{by Lemma 4 in Appendix E.3}$$

$$= \quad (-1)^n \begin{pmatrix} s \\ n \end{pmatrix},$$

completing the proof of the lemma. □

E.3 Lemma 4

Lemma 4

$$\sum_{k=0}^{n} (-1)^k \begin{pmatrix} s-k \\ s-n-1 \end{pmatrix} \begin{pmatrix} s \\ k \end{pmatrix} = (-1)^n \begin{pmatrix} s \\ n+1 \end{pmatrix}, \quad \textit{for } n = 0, \ldots, s-1,\ s \geq 1. \tag{E.49}$$

Proof.

$$\begin{aligned} LS = RS \quad &\Longleftrightarrow \quad \sum_{k=0}^{n+1} (-1)^k \begin{pmatrix} s-k \\ s-n-1 \end{pmatrix} \begin{pmatrix} s \\ k \end{pmatrix} = 0 \\ &\Longleftrightarrow \quad \sum_{k=0}^{n+1} (-1)^k \frac{(s-k)!}{(s-n-1)!(n+1-k)!} \frac{s!}{k!(s-k)!} = 0 \\ &\Longleftrightarrow \quad \sum_{k=0}^{n+1} (-1)^k \frac{1}{(n+1-k)!k!} = 0 \\ &\Longleftrightarrow \quad \frac{1}{(n+1)!} \sum_{k=0}^{n+1} (-1)^k \begin{pmatrix} n+1 \\ k \end{pmatrix} = 0, \end{aligned}$$

which is true by (F.3). □

E.4 Lemma 5

Lemma 5

$$\Sigma_l^n = (-1)^{l+1} \sum_{m=1}^{l+1} (-1)^m \begin{pmatrix} l+1 \\ m \end{pmatrix} m^n, \quad \textit{for } l = 0, \ldots, n-1,\ n \geq 1. \tag{E.50}$$

Proof. The proof is by induction on l. We first handle the base cases. For $l = 0$,

$$RS = (-1)^1(-1)^1 \left(\begin{array}{c} 1 \\ 1 \end{array} \right) 1^n = 1 = \Sigma_0^n = LS$$

by (B.6). For $l = 1$,

$$RS = -\left(\begin{array}{c} 2 \\ 1 \end{array} \right) 1^n + \left(\begin{array}{c} 2 \\ 2 \end{array} \right) 2^n = 2^n - 2,$$

while

$$LS = \Sigma_1^n = \sum_{j=1}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right),$$

by (B.6). However, (F.2) gives

$$\sum_{j=0}^{n} \left(\begin{array}{c} n \\ j \end{array} \right) = 2^n,$$

from which it follows that

$$LS = 2^n - \left(\begin{array}{c} n \\ 0 \end{array} \right) - \left(\begin{array}{c} n \\ n \end{array} \right) = 2^n - 2 = RS.$$

Now consider the general case, $l \geq 2$ (hence $n \geq 3$). We use the recursive expression for Σ_l^n given by Lemma 11 in Appendix E.10, which, together with (E.50) by induction, gives

$$\begin{aligned}
\Sigma_l^n &= \sum_{j=1}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) (-1)^l \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^j \\
&= (-1)^l \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) \sum_{j=1}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) m^j \\
&= (-1)^l \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) [(1+m)^n - m^n - 1] \\
&= (-1)^l \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) (1+m)^n \right. \\
&\qquad \left. - \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^n - \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) \right] \\
&= (-1)^l \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) (1+m)^n - \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^n + 1 \right], \quad \text{(E.51)}
\end{aligned}$$

by (F.3), since in the general inductive step $l \geq 1$. For the right side of (E.50), we have

$$\begin{aligned}
&(-1)^{l+1} \sum_{m=1}^{l+1} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) m^n \\
&= (-1)^{l+1} \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) m^n + (-1)^{l+1} (l+1)^n \right] \\
&= (-1)^{l+1} \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^n + \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m-1 \end{array} \right) m^n + (-1)^{l+1} (l+1)^n \right] \\
&= (-1)^{l+1} \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^n - \sum_{m=0}^{l-1} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) (1+m)^n + (-1)^{l+1} (l+1)^n \right] \\
&= (-1)^{l+1} \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^n - \sum_{m=0}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) (1+m)^n \right] \\
&= (-1)^{l+1} \left[\sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) m^n - \sum_{m=1}^{l} (-1)^m \left(\begin{array}{c} l \\ m \end{array} \right) (1+m)^n + 1 \right],
\end{aligned}$$

which equals (E.51), as required. □

E.5 Lemma 6

Lemma 6

$$U_m^{(s)} = \sum_{i=1}^{s} \gamma_{s-i}^{(s)} m^i \quad \textit{from (A.27)} \tag{E.52}$$

$$= s! \left[\left(\begin{array}{c} m+s \\ s \end{array} \right) - 1 \right], \quad \textit{for } s \geq 1,\ m \geq 0. \tag{E.53}$$

Proof. We first establish a recursive relationship for $\gamma_k^{(s)}$. For $k \geq 2$, it follows from (A.7) that

$$\begin{aligned}
\gamma_k^{(s)} &= \sum_{1 \leq i_1 < \cdots < i_k \leq s} i_1 \cdots i_k \\
&= \sum_{1 \leq i_1 < \cdots < i_k < s} i_1 \cdots i_k + \sum_{1 \leq i_1 < \cdots < i_k = s} i_1 \cdots i_k \\
&= \sum_{1 \leq i_1 < \cdots < i_k \leq s-1} i_1 \cdots i_k + s \sum_{1 \leq i_1 < \cdots < i_{k-1} \leq s-1} i_1 \cdots i_k
\end{aligned}$$

$$= \quad s\gamma_{k-1}^{(s-1)} + \gamma_k^{(s-1)}.$$

It is easy to see from (A.7) that this recursive relationship holds for $k = 1$ also. Therefore we have shown that

$$\gamma_k^{(s)} = s\gamma_{k-1}^{(s-1)} + \gamma_k^{(s-1)} \tag{E.54}$$

for $k \geq 1$.

We now prove the lemma by induction on s. For $s = 1$, it follows from (E.52) and (A.7) that

$$LS = \gamma_0^{(1)} m^1 = m = 1! \left[\begin{pmatrix} m+1 \\ 1 \end{pmatrix} - 1 \right] = RS.$$

For $s > 1$, we get by using (E.54) in (E.52) and then by using (A.7) that

$$\begin{aligned}
U_m^{(s)} &= \sum_{i=1}^{s} \gamma_{s-i}^{(s)} m^i \\
&= \sum_{i=1}^{s-1} \gamma_{s-i}^{(s)} m^i + \gamma_0^{(s)} m^s \\
&= \sum_{i=1}^{s-1} \left(\gamma_{s-i}^{(s-1)} + s\gamma_{s-i-1}^{(s-1)} \right) m^i + \gamma_0^{(s)} m^s \\
&= \sum_{i=1}^{s-1} \gamma_{s-i}^{(s-1)} m^i + \sum_{i=1}^{s-1} s\gamma_{s-i-1}^{(s-1)} m^i + m^s \\
&= m \sum_{i=0}^{s-2} \gamma_{s-i-1}^{(s-1)} m^i + s \sum_{i=1}^{s-1} \gamma_{s-i-1}^{(s-1)} m^i + m^s \\
&= m \left(U_m^{(s-1)} + \gamma_{s-1}^{(s-1)} m^0 - \gamma_0^{(s-1)} m^{s-1} \right) + s U_m^{(s-1)} + m^s \\
&= m U_m^{(s-1)} + m(s-1)! - m m^{s-1} + s U_m^{(s-1)} + m^s \\
&= (m+s) U_m^{(s-1)} + m(s-1)! \\
&= (m+s)(s-1)! \left[\begin{pmatrix} m+s-1 \\ s-1 \end{pmatrix} - 1 \right] + m(s-1)! \quad \text{by induction on (E.53)} \\
&= (s-1)! \left[(m+s) \begin{pmatrix} m+s-1 \\ s-1 \end{pmatrix} - (m+s) + m \right] \\
&= s! \left[\frac{(m+s)}{s} \begin{pmatrix} m+s-1 \\ s-1 \end{pmatrix} - 1 \right] \\
&= s! \left[\frac{(m+s)(m+s-1)!}{s(s-1)!m!} - 1 \right]
\end{aligned}$$

$$
\begin{aligned}
&= s!\left[\frac{(m+s)!}{s!m!}-1\right] \\
&= s!\left[\left(\begin{array}{c} m+s \\ s \end{array}\right)-1\right],
\end{aligned}
$$

as required. □

E.6 Lemma 7

Lemma 7

$$
\sum_{l=1}^{s-k}(-1)^l\left(\begin{array}{c} s \\ l \end{array}\right)\left(\begin{array}{c} s-l \\ k \end{array}\right) = -\left(\begin{array}{c} s \\ k \end{array}\right), \quad \textit{for } k=0,\ldots,s-1,\ s\geq 1. \tag{E.55}
$$

Proof.

$$
\begin{aligned}
LS = RS &\iff \sum_{l=0}^{s-k}(-1)^l\left(\begin{array}{c} s \\ l \end{array}\right)\left(\begin{array}{c} s-l \\ k \end{array}\right) = 0 \\
&\iff \sum_{l=0}^{s-k}(-1)^l\frac{s!}{l!(s-l)!}\frac{(s-l)!}{k!(s-l-k)!} = 0 \\
&\iff \sum_{l=0}^{s-k}(-1)^l\frac{1}{l!(s-l-k)!} = 0 \\
&\iff \frac{1}{(s-k)!}\sum_{l=0}^{s-k}(-1)^l\left(\begin{array}{c} s-k \\ l \end{array}\right) = 0,
\end{aligned}
$$

which is true by (F.3). □

E.7 Lemma 8

Lemma 8

$$
M_l^{(s)} = \sum_{i=2}^{s} i\Sigma_l^{i-1}S_{i+1}^{(s)} \quad \textit{from (A.34)} \tag{E.56}
$$

$$= \begin{pmatrix} s \\ l+2 \end{pmatrix}, \quad \textit{for } l = 0, \ldots, s-1,\ s \geq 2. \tag{E.57}$$

In particular, for $l = s-1$ *we have*

$$M_{s-1}^{(s)} = \begin{pmatrix} s \\ s+1 \end{pmatrix} = 0.$$

Proof. From (E.56),

$$M_l^{(s)} = \sum_{i=2}^{s} i \Sigma_l^{i-1} S_{i+1}^{(s)}. \tag{E.58}$$

We begin by finding a convenient expression for $S_{i+1}^{(s)}$. From (E.68) of Lemma 13, we have

$$\begin{aligned} S_{i+1}^{(s)} &= \frac{1}{s!i!} L_{i+1}^{(s)} \\ &= \frac{1}{s!i!} \sum_{k=0}^{s-i} \frac{(i+k)!}{(k+1)!} \gamma_{s-i-k}^{(s)} \\ &= \frac{1}{s!i!} \sum_{k=0}^{s-i} \frac{(s-k)!}{(s-i-k+1)!} \gamma_k^{(s)} \\ &= \frac{1}{s!i} \sum_{k=0}^{s-i} \begin{pmatrix} s-k \\ i-1 \end{pmatrix} \gamma_k^{(s)}. \end{aligned}$$

Using this in (E.58) and then using (E.50) of Lemma 5 for Σ_l^{i-1}, we get

$$\begin{aligned} M_l^{(s)} &= \sum_{i=2}^{s} i \Sigma_l^{i-1} \frac{1}{s!i} \sum_{k=0}^{s-i} \begin{pmatrix} s-k \\ i-1 \end{pmatrix} \gamma_k^{(s)} \\ &= \frac{1}{s!} \sum_{i=2}^{s} \Sigma_l^{i-1} \sum_{k=0}^{s-i} \begin{pmatrix} s-k \\ i-1 \end{pmatrix} \gamma_k^{(s)} \\ &= \frac{1}{s!} \sum_{i=2}^{s} (-1)^{l+1} \sum_{m=1}^{l+1} (-1)^m \begin{pmatrix} l+1 \\ m \end{pmatrix} m^{i-1} \sum_{k=0}^{s-i} \begin{pmatrix} s-k \\ i-1 \end{pmatrix} \gamma_k^{(s)} \\ &= \frac{1}{s!} (-1)^{l+1} \sum_{m=1}^{l+1} (-1)^m \begin{pmatrix} l+1 \\ m \end{pmatrix} \sum_{i=2}^{s} m^{i-1} \sum_{k=0}^{s-i} \begin{pmatrix} s-k \\ i-1 \end{pmatrix} \gamma_k^{(s)} \\ &= (-1)^{l+1} \sum_{m=1}^{l+1} (-1)^m \begin{pmatrix} l+1 \\ m \end{pmatrix} V_m^{(s)}, \end{aligned} \tag{E.59}$$

where

$$V_m^{(s)} = \frac{1}{s!}\sum_{i=2}^{s} m^{i-1} \sum_{k=0}^{s-i} \left(\begin{array}{c} s-k \\ i-1 \end{array} \right) \gamma_k^{(s)}. \tag{E.60}$$

Substituting the expression for $V_m^{(s)}$ given by Lemma 12 in Appendix E.11 into (E.59), we get

$$\begin{aligned}
M_l^{(s)} &= (-1)^{l+1} \sum_{m=1}^{l+1} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) \left[\left(\begin{array}{c} s+m \\ 1+m \end{array} \right) - s \right] \\
&= (-1)^{l+1} \left\{ \sum_{m=1}^{l+1} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) \left(\begin{array}{c} s+m \\ 1+m \end{array} \right) - s \sum_{m=1}^{l+1} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) \right\} \\
&= (-1)^{l+1} \left\{ \sum_{m=0}^{l+1} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) \left(\begin{array}{c} s+m \\ 1+m \end{array} \right) - \left(\begin{array}{c} l+1 \\ 0 \end{array} \right) \left(\begin{array}{c} s \\ 1 \end{array} \right) \right. \\
&\quad \left. -s \left[\sum_{m=0}^{l+1} (-1)^m \left(\begin{array}{c} l+1 \\ m \end{array} \right) - \left(\begin{array}{c} l+1 \\ 0 \end{array} \right) \right] \right\} \\
&= (-1)^{l+1} \left\{ (-1)^{l+1} \left(\begin{array}{c} s \\ l+2 \end{array} \right) - s - s(0-1) \right\} \\
&= \left(\begin{array}{c} s \\ l+2 \end{array} \right),
\end{aligned}$$

by (F.8) and (F.3), completing the proof of the lemma. □

E.8 Lemma 9

Lemma 9

$$S_2^{(s)} = s, \quad \textit{for } s \geq 1. \tag{E.61}$$

Proof. From the expression for $S_i^{(s)}$ given by Lemma 13 in Appendix E.12, it follows that

$$S_2^{(s)} = \frac{1}{s!} L_2^{(s)} = \frac{1}{s!} \sum_{l=0}^{s-1} \gamma_{s-l-1}^{(s)} = \frac{1}{s!} \sum_{l=0}^{s-1} \gamma_l^{(s)}.$$

So we must prove

$$\sum_{l=0}^{s-1} \gamma_l^{(s)} = s s!, \quad \text{for } s \geq 1,$$

which is equivalent to

$$\sum_{l=0}^{s} \gamma_l^{(s)} = ss! + s! = (s+1)!, \quad \text{for } s \geq 1, \tag{E.62}$$

since $\gamma_s^{(s)} = s!$ by (A.7). The relation (E.62) follows from Lemma 14 in Appendix E.13. □

E.9 Lemma 10

Lemma 10

$$\sum_{l=0}^{N} (-1)^l \begin{pmatrix} s-l \\ N-l \end{pmatrix} \begin{pmatrix} s \\ l+1 \end{pmatrix} = \begin{pmatrix} s+1 \\ s-N \end{pmatrix}, \quad \textit{for } N = 1, \ldots, s,\ s \geq 1. \tag{E.63}$$

Proof.

$$RS = \begin{pmatrix} s+1 \\ N+1 \end{pmatrix},$$

so that

$$\begin{aligned}
LS = RS \iff & \sum_{l=-1}^{N} (-1)^l \begin{pmatrix} s-l \\ N-l \end{pmatrix} \begin{pmatrix} s \\ l+1 \end{pmatrix} = 0 \\
\iff & \sum_{l=0}^{M} (-1)^l \begin{pmatrix} s-l+1 \\ M-l \end{pmatrix} \begin{pmatrix} s \\ l \end{pmatrix} = 0, \quad M = N+1 \\
\iff & \sum_{l=0}^{M} (-1)^l \frac{(s-l+1)!}{(M-l)!(s+1-M)!} \frac{s!}{l!(s-l)!} = 0 \\
\iff & \sum_{l=0}^{M} (-1)^l \frac{s-l+1}{(M-l)!l!} = 0 \\
\iff & (s+1) \sum_{l=0}^{M} (-1)^l \frac{1}{(M-l)!l!} - \sum_{l=0}^{M} (-1)^l \frac{l}{(M-l)!l!} = 0 \\
\iff & \frac{s+1}{M!} \sum_{l=0}^{M} (-1)^l \begin{pmatrix} M \\ l \end{pmatrix} - \frac{1}{(M-1)!} \sum_{l=0}^{M} (-1)^l \begin{pmatrix} M-1 \\ l-1 \end{pmatrix} = 0 \\
\iff & \frac{s+1}{M!} \sum_{l=0}^{M} (-1)^l \begin{pmatrix} M \\ l \end{pmatrix} + \frac{1}{N!} \sum_{l=0}^{N} (-1)^l \begin{pmatrix} N \\ l \end{pmatrix} = 0,
\end{aligned}$$

which is true since both the sums are zero by (F.3). □

E.10 Lemma 11

Lemma 11

$$\Sigma_l^n = \sum_{j=1}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) \Sigma_{l-1}^j = \sum_{j=l}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) \Sigma_{l-1}^j, \quad \textit{for } l = 2, \ldots, n-1,\ n \geq 3. \tag{E.64}$$

Proof. Since $l \geq 2$, it follows from (B.6) that

$$\begin{aligned}
\Sigma_l^n &= \sum_{1 \leq i_1 < \cdots < i_l \leq n-1} \left(\begin{array}{c} i_2 \\ i_1 \end{array} \right) \left(\begin{array}{c} i_3 \\ i_2 \end{array} \right) \cdots \left(\begin{array}{c} i_l \\ i_{l-1} \end{array} \right) \left(\begin{array}{c} n \\ i_l \end{array} \right) \\
&= \sum_{i_l=l}^{n-1} \left(\begin{array}{c} n \\ i_l \end{array} \right) \sum_{1 \leq i_1 < \cdots < i_{l-1} < i_l} \left(\begin{array}{c} i_2 \\ i_1 \end{array} \right) \left(\begin{array}{c} i_3 \\ i_2 \end{array} \right) \cdots \left(\begin{array}{c} i_l \\ i_{l-1} \end{array} \right) \\
&= \sum_{j=l}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) \sum_{1 \leq i_1 < \cdots < i_{l-1} \leq j-1} \left(\begin{array}{c} i_2 \\ i_1 \end{array} \right) \left(\begin{array}{c} i_3 \\ i_2 \end{array} \right) \cdots \left(\begin{array}{c} i_{l-1} \\ i_{l-2} \end{array} \right) \left(\begin{array}{c} j \\ i_{l-1} \end{array} \right) \\
&= \sum_{j=l}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) \Sigma_{l-1}^j,
\end{aligned}$$

as required. This also equals

$$\sum_{j=1}^{n-1} \left(\begin{array}{c} n \\ j \end{array} \right) \Sigma_{l-1}^j,$$

because $\Sigma_l^j = 0$ for $j \leq l-1$ by (B.6). □

E.11 Lemma 12

Lemma 12

$$V_m^{(s)} = \frac{1}{s!} \sum_{i=2}^{s} m^{i-1} \sum_{k=0}^{s-i} \left(\begin{array}{c} s-k \\ i-1 \end{array} \right) \gamma_k^{(s)} \quad \textit{from (E.60)} \tag{E.65}$$

$$= \begin{pmatrix} s+m \\ 1+m \end{pmatrix} - s, \quad \textit{for } s \geq 1,\ m \geq 0. \tag{E.66}$$

Proof. We first switch the i and k sums in (E.65). When $i = 2$, $k = 0, \ldots, s-2$. When $i = s$, $k = 0$. Therefore, for $k = 0, \ldots, s-2$, $i = 2, \ldots, s-k$. This gives

$$\begin{aligned} V_m^{(s)} &= \frac{1}{s!} \sum_{k=0}^{s-2} \gamma_k^{(s)} \sum_{i=2}^{s-k} \begin{pmatrix} s-k \\ i-1 \end{pmatrix} m^{i-1} \\ &= \frac{1}{s!} \sum_{k=0}^{s-2} \gamma_k^{(s)} \sum_{i=1}^{s-k-1} \begin{pmatrix} s-k \\ i \end{pmatrix} m^{i} \\ &= \frac{1}{s!} \sum_{k=0}^{s-2} \gamma_k^{(s)} \left[\sum_{i=0}^{s-k} \begin{pmatrix} s-k \\ i \end{pmatrix} m^{i} - m^{s-k} - 1 \right] \\ &= \frac{1}{s!} \left[\sum_{k=0}^{s-2} \gamma_k^{(s)} (1+m)^{s-k} - \sum_{k=0}^{s-2} \gamma_k^{(s)} m^{s-k} - \sum_{k=0}^{s-2} \gamma_k^{(s)} \right]. \end{aligned} \tag{E.67}$$

We now find expressions for the individual sums. From Lemma 6 in Appendix E.5, we have

$$\begin{aligned} &\sum_{i=1}^{s} \gamma_{s-i}^{(s)} m^i = s! \begin{pmatrix} m+s \\ s \end{pmatrix} - s! \\ \implies &\sum_{i=0}^{s} \gamma_{s-i}^{(s)} m^i = s! \begin{pmatrix} m+s \\ s \end{pmatrix}, \quad \text{since } \gamma_s^{(s)} = s! \text{ by (A.7)} \\ \implies &\sum_{i=0}^{s} \gamma_{i}^{(s)} m^{s-i} = s! \begin{pmatrix} m+s \\ s \end{pmatrix} \\ \implies &\sum_{i=0}^{s-2} \gamma_{i}^{(s)} m^{s-i} = s! \begin{pmatrix} m+s \\ s \end{pmatrix} - s! - \gamma_{s-1}^{(s)} m, \end{aligned}$$

and, replacing m by $m+1$, we get

$$\sum_{i=0}^{s-2} \gamma_{i}^{(s)} (m+1)^{s-i} = s! \begin{pmatrix} s+m+1 \\ s \end{pmatrix} - s! - \gamma_{s-1}^{(s)} (m+1).$$

Using these relations, together with (E.74) of Lemma 14, in (E.67), we find that

$$V_m^{(s)} = \frac{1}{s!} \left\{ \left[s! \begin{pmatrix} s+m+1 \\ s \end{pmatrix} - s! - (m+1) \gamma_{s-1}^{(s)} \right] \right.$$

$$
\begin{aligned}
&\quad - \left[s! \begin{pmatrix} m+s \\ s \end{pmatrix} - s! - m\gamma_{s-1}^{(s)} \right] \\
&\quad - \left[(s+1)! - \gamma_s^{(s)} - \gamma_{s-1}^{(s)} \right] \Big\} \\
&= \frac{1}{s!} \left\{ s! \left[\begin{pmatrix} s+m+1 \\ s \end{pmatrix} - \begin{pmatrix} m+s \\ s \end{pmatrix} \right] - (s+1)! + \gamma_s^{(s)} \right\} \\
&= \begin{pmatrix} s+m+1 \\ s \end{pmatrix} - \begin{pmatrix} m+s \\ s \end{pmatrix} - (s+1) + 1 \quad \text{by (A.7)} \\
&= \begin{pmatrix} s+m \\ s-1 \end{pmatrix} - s \\
&= \begin{pmatrix} s+m \\ 1+m \end{pmatrix} - s,
\end{aligned}
$$

as required. □

E.12 Lemma 13

Lemma 13

$$
S_i^{(s)} = \sum_{j=1}^{s} b_{ij} = \frac{1}{s!(i-1)!} L_i^{(s)}, \quad \textit{for } i = 2, \ldots, r,\ s \geq 1, \tag{E.68}
$$

where

$$
L_i^{(s)} = \sum_{l=0}^{r-i} \frac{(i+l-1)!}{(l+1)!} \gamma_{s-i-l+1}^{(s)}. \tag{E.69}
$$

Proof. Using (A.5), (A.6) and (A.1), we see for $i = 1, \ldots, r$ that

$$
\begin{aligned}
S_i^{(s)} &= \sum_{j=1}^{s} b_{ij} \\
&= \sum_{j=1}^{s} T_{ij}(\lambda) b_{rj} \\
&= \sum_{j=1}^{s} \left[\left(\frac{1}{(i-1)!} \sum_{l=0}^{r-i} (i-1+l)! \gamma_{s-i-l+1}^{(s)} \lambda_j^l \right) \left(\frac{1}{s!} \frac{\sum_{k=0}^{s-1} \frac{\theta_{kj}^{(s)}}{(s-k)!}}{\prod_{\substack{m=1 \\ m \neq j}}^{s} (\lambda_m - \lambda_j)} \right) \right]
\end{aligned}
$$

$$= \frac{1}{s!(i-1)!}\sum_{l=0}^{r-i}(i-1+l)!\gamma_{s-i-l+1}^{(s)}\sum_{k=0}^{s-1}\frac{1}{(s-k)!}\sum_{j=1}^{s}\frac{\lambda_j^l\theta_{kj}^{(s)}}{\prod_{\substack{m=1\\m\neq j}}^{s}(\lambda_m-\lambda_j)}$$

$$= \frac{1}{s!(i-1)!}\sum_{l=0}^{r-i}(i-1+l)!\gamma_{s-i-l+1}^{(s)}K_l^{(s)}, \tag{E.70}$$

where

$$K_l^{(s)} = \sum_{k=0}^{s-1}\frac{1}{(s-k)!}J_{l,k}^{(s)} \tag{E.71}$$

and

$$J_{l,k}^{(s)} = \sum_{j=1}^{s}\frac{\lambda_j^l\theta_{kj}^{(s)}}{\prod_{\substack{m=1\\m\neq j}}^{s}(\lambda_m-\lambda_j)}. \tag{E.72}$$

Using Lemma 15 of Appendix E.14 together with (E.71), we find that

$$K_l^{(s)} = \frac{1}{[(s-(s-l-1)]!} = \frac{1}{(l+1)!}, \quad \text{for } l=0,\ldots,s. \tag{E.73}$$

Therefore, from (E.70),

$$\begin{aligned} S_i^{(s)} &= \frac{1}{s!(i-1)!}\sum_{l=0}^{r-i}\frac{(i-1+l)!}{(l+1)!}\gamma_{s-i-l+1}^{(s)} \\ &= \frac{1}{s!(i-1)!}L_i^{(s)}, \end{aligned}$$

as required. □

E.13 Lemma 14

Lemma 14

$$\sum_{k=0}^{s}\gamma_k^{(s)} = (s+1)!, \quad \textit{for } s \geq 0. \tag{E.74}$$

Proof. The proof is by induction on s. For base case, $s=0$, so

$$LS = \gamma_0^{(s)} = 1 = RS$$

by (A.7).

Before considering the general case, $s \geq 1$, we first prove the following recurrence for $\gamma_l^{(s)}$ for $l \geq 1$ by induction on l:

$$\gamma_l^{(s)} = \sum_{n=l-1}^{s-1} (n+1)\gamma_{l-1}^{(n)}. \tag{E.75}$$

In the base case, $l = 1$,

$$RS = \sum_{n=0}^{s-1}(n+1)\gamma_0^{(n)} = \sum_{n=1}^{s} n = \gamma_1^{(s)} = LS$$

by (A.7). For the general case, $l \geq 2$, we use (E.54) to show

$$\begin{aligned}
\gamma_l^{(s)} &= s\gamma_{l-1}^{(s-1)} + \gamma_l^{(s-1)} \\
&= s\gamma_{l-1}^{(s-1)} + \sum_{n=l-1}^{s-2} (n+1)\gamma_{l-1}^{(n)} \quad \text{by induction on (E.75)} \\
&= \sum_{n=l-1}^{s-1} (n+1)\gamma_{l-1}^{(n)} \\
&= RS,
\end{aligned}$$

which proves the recurrence (E.75).

To prove the general case of (E.74) for $s \geq 1$, we use (E.75) in (E.74) to get

$$LS = \gamma_0^{(s)} + \sum_{l=1}^{s} \sum_{n=l-1}^{s-1} (n+1)\gamma_{l-1}^{(n)}.$$

Now switch the l and n sums. When $l = 1$, $n = 0, \ldots, s-1$. When $l = s$, $n = s-1$. Thus, for $n = 0, \ldots, s-1$, $l = 1, \ldots, n+1$. Therefore,

$$\begin{aligned}
LS &= 1 + \sum_{n=0}^{s-1} \sum_{l=1}^{n+1} (n+1)\gamma_{l-1}^{(n)} \\
&= 1 + \sum_{n=0}^{s-1} (n+1) \sum_{l=0}^{n} \gamma_l^{(n)} \\
&= 1 + \sum_{n=0}^{s-1} (n+1)(n+1)! \qquad \text{(E.76)}
\end{aligned}$$

where (E.76) follows by induction on (E.74), since $n < s$. To prove (E.76) equals the right side of (E.74), we need to show

$$\sum_{n=0}^{s-1}(n+1)(n+1)! = (s+1)! - 1,$$

which is equivalent to

$$\sum_{n=1}^{s} nn! = (s+1)! - 1. \tag{E.77}$$

We prove (E.77) by induction on $s \geq 1$.

In the base case, $s = 1$,

$$LS = 1 = 2 - 1 = RS.$$

In the general case, $s \geq 2$,

$$\begin{aligned}
\sum_{n=1}^{s} nn! &= \sum_{n=1}^{s-1} nn! + ss! \\
&= s! - 1 + ss! \quad \text{by induction on (E.77)} \\
&= s!(1+s) - 1 \\
&= (s+1)! - 1,
\end{aligned}$$

as required. □

E.14 Lemma 15

Lemma 15

$$\begin{aligned}
J_{l,k}^{(s)} &= \sum_{j=1}^{s} \frac{\lambda_j^l \theta_{kj}^{(s)}}{\prod_{\substack{m=1 \\ m\neq j}}^{s}(\lambda_m - \lambda_j)} \quad \textit{from (E.72)} \\
&= \left\{ \begin{array}{ll} 0, & k = 0, \ldots, s-l-2, \\ 1, & k = s-l-1, \\ 0, & k \geq s-l, \end{array} \right\} \quad \textit{for } l = 0, \ldots, s-1,\ s \geq 1.
\end{aligned} \tag{E.78}$$

Proof. Fix s, l, k for now and let

$$a_j = \lambda_j^l \theta_{kj}^{(s)}. \tag{E.79}$$

Then

$$J_{l,k}^{(s)} = \sum_{j=1}^{s} \frac{a_j}{\prod_{\substack{m=1 \\ m\neq j}}^{s}(\lambda_m - \lambda_j)}. \tag{E.80}$$

Consider the common denominator

$$d = \prod_{1\leq i<m\leq s} (\lambda_i - \lambda_m),$$

and let

$$n_j = (-1)^{s-j} \prod_{\substack{1\leq i<m\leq s \\ i,m\neq j}} (\lambda_i - \lambda_m).$$

Then

$$\begin{aligned}
\frac{n_j}{d} &= \frac{(-1)^{s-j}\prod_{\substack{1\leq i<m\leq s \\ i,m\neq j}}(\lambda_i - \lambda_m)}{\prod_{1\leq i<m\leq s}(\lambda_i - \lambda_m)} \\
&= \frac{(-1)^{s-j}\prod_{\substack{1\leq i<m\leq s \\ i,m\neq j}}(\lambda_i - \lambda_m)}{\prod_{\substack{1\leq i<m\leq s \\ i,m\neq j}}(\lambda_i - \lambda_m)\prod_{\substack{1\leq i<m\leq s \\ i=j}}(\lambda_i - \lambda_m)\prod_{\substack{1\leq i<m\leq s \\ m=j}}(\lambda_i - \lambda_m)} \\
&= \frac{(-1)^{s-j}}{\prod_{\substack{1\leq i<m\leq s \\ i=j}}(\lambda_i - \lambda_m)\prod_{\substack{1\leq i<m\leq s \\ m=j}}(\lambda_i - \lambda_m)} \\
&= \frac{(-1)^{s-j}}{\prod_{1\leq j<m\leq s}(\lambda_j - \lambda_m)\prod_{1\leq i<j\leq s}(\lambda_i - \lambda_j)} \\
&= \frac{1}{\prod_{1\leq j<m\leq s}(\lambda_m - \lambda_j)\prod_{1\leq m<j\leq s}(\lambda_m - \lambda_j)} \\
&= \frac{1}{\prod_{\substack{m=1 \\ m\neq j}}^{s}(\lambda_m - \lambda_j).}
\end{aligned}$$

Therefore, from (E.80), we have

$$\begin{aligned}
J_{l,k}^{(s)} &= \frac{\sum_{j=1}^{s} a_j n_j}{d} \\
&= \frac{\sum_{j=1}^{s} \lambda_j^l \theta_{kj}^{(s)} (-1)^{s-j} \prod_{\substack{1\leq i<m\leq s \\ i,m\neq j}}(\lambda_i - \lambda_m)}{\prod_{1\leq i<j\leq s}(\lambda_i - \lambda_j)} \\
&= \frac{(-1)^s A_{l,k}^{(s)}}{\prod_{1\leq i<j\leq s}(\lambda_i - \lambda_j)},
\end{aligned} \tag{E.81}$$

where

$$A_{l,k}^{(s)} = \sum_{j=1}^{s} \lambda_j^l \theta_{kj}^{(s)} (-1)^j \prod_{\substack{1 \le i < m \le s \\ i,m \ne j}} (\lambda_i - \lambda_m). \tag{E.82}$$

So (E.78) is equivalent to

$$A_{l,k}^{(s)} = \left\{ \begin{array}{ll} 0, & k \ge 0,\ k \ne s-l-1, \\ (-1)^s \prod_{1 \le i < j \le s} (\lambda_i - \lambda_j), & k = s-l-1, \end{array} \right\} \quad \text{for } l = 0, \ldots, s,\ s \ge 1. \tag{E.83}$$

From (A.3) and (A.4), it follows that

$$\theta_{kj}^{(s)} = \left\{ \begin{array}{ll} (-1)^{s-1-k} \sum_{\substack{1 \le i_1 < \cdots < i_k \le s \\ i_1, \ldots, i_k \ne j}} \lambda_{i_1} \cdots \lambda_{i_k}, & k \ne 0, \\ (-1)^{s-1}, & k = 0. \end{array} \right. \tag{E.84}$$

Therefore, for any fixed k, both the left side and the right side of (E.83) may be considered as polynomials in λ_s, say. The degree of the $j = s$ term of $A_{l,k}^{(s)}$ in λ_s is $l + 0 + 0 \le s - 1$. The degree of any $j < s$ term is $0 + 1 + (s-2) = s - 1$. Thus, it suffices to show (E.83) is true for s distinct values of λ_s, for example, $\lambda_s = 0, \lambda_1, \ldots, \lambda_{s-1}$.

We begin with the case of $\lambda_s = \lambda_n$, for some $n = 1, \ldots, s-1$. Consider $j \ne n$ and $j \ne s$ in (E.82). In this case, $\lambda_n - \lambda_s$ occurs in the product, so the only possibly nonzero terms of the sum occur when $j = n$ or $j = s$. That is,

$$A_{l,k}^{(s)} = \lambda_n^l \theta_{kn}^{(s)} (-1)^n \prod_{\substack{1 \le i < m \le s \\ i,m \ne n}} (\lambda_i - \lambda_m) + \lambda_s^l \theta_{ks}^{(s)} (-1)^s \prod_{\substack{1 \le i < m \le s \\ i,m \ne s}} (\lambda_i - \lambda_m). \tag{E.85}$$

Note that, from (E.84), $\theta_{k,n}^{(s)} = \theta_{k,s}^{(s)}$, since $\lambda_n = \lambda_s$. Also,

$$\begin{aligned} & \prod_{\substack{1 \le i < m \le s \\ i,m \ne n}} (\lambda_i - \lambda_m) \\ = & \prod_{\substack{1 \le i < m < s \\ i,m \ne n}} (\lambda_i - \lambda_m) \prod_{\substack{1 \le i < s \\ i \ne n}} (\lambda_i - \lambda_s) \\ = & \prod_{\substack{1 \le i < m < s \\ i,m \ne n}} (\lambda_i - \lambda_m) \prod_{\substack{1 \le i < s \\ i \ne n}} (\lambda_i - \lambda_n) \\ = & \prod_{\substack{1 \le i < m < s \\ i,m \ne n}} (\lambda_i - \lambda_m) \prod_{1 \le i < n} (\lambda_i - \lambda_n) \prod_{n < i < s} (\lambda_i - \lambda_n) \end{aligned}$$

$$
\begin{aligned}
&= \prod_{\substack{1\le i<m<s\\ i,m\neq n}} (\lambda_i-\lambda_m) \prod_{1\le i<n} (\lambda_i-\lambda_n)(-1)^{s-n-1} \prod_{n<i<s} (\lambda_n-\lambda_i)\\
&= (-1)^{s-n-1} \prod_{1\le i<m<s} (\lambda_i-\lambda_m)\\
&= (-1)^{s-n-1} \prod_{\substack{1\le i<m\le s\\ i,m\neq s}} (\lambda_i-\lambda_m). \qquad \text{(E.86)}
\end{aligned}
$$

Thus, continuing from (E.85), we find that

$$
\begin{aligned}
A^{(s)}_{l,k} &= \lambda_n^l \theta^{(s)}_{ks} (-1)^n (-1)^{s-n-1} \prod_{\substack{1\le i<m\le s\\ i,m\neq s}} (\lambda_i-\lambda_m) + \lambda_s^l \theta^{(s)}_{ks} (-1)^s \prod_{\substack{1\le i<m\le s\\ i,m\neq s}} (\lambda_i-\lambda_m)\\
&= 0 \qquad \text{(E.87)}
\end{aligned}
$$

We must now show that this agrees with the right side of (E.83). When $k \neq s-l-1$, the right side is 0. When $k = s-l-1$, it is

$$
(-1)^s \prod_{1\le i<j\le s} (\lambda_i-\lambda_j) = 0,
$$

because $\lambda_n = \lambda_s$. We have just shown (E.83) for $\lambda_s = \lambda_n$, $n = 1,\ldots,s-1$.

It remains to show (E.83) for $\lambda_s = 0$, which we do by induction on s and l. First we show the general case, $s > 1$ and $l > 0$. From (E.82),

$$
\begin{aligned}
A^{(s)}_{l,k} &= \sum_{j=1}^{s-1} \lambda_j^l \theta^{(s)}_{kj} (-1)^j \prod_{\substack{1\le i<m\le s\\ i,m\neq j}} (\lambda_i-\lambda_m) + (-1)^s \lambda_s^l \theta^{(s)}_{ks} \prod_{\substack{1\le i<m\le s\\ i,m\neq s}} (\lambda_i-\lambda_m)\\
&= \sum_{j=1}^{s-1} \lambda_j^l \theta^{(s)}_{kj} (-1)^j \prod_{\substack{1\le i<m\le s\\ i,m\neq j}} (\lambda_i-\lambda_m), \qquad \text{(E.88)}
\end{aligned}
$$

because $l > 0$ implies $\lambda_s^l = 0^l = 0$. We now find recursive relationships for the expressions in (E.88). When $\lambda_s = 0$, (E.84) implies that

$$
\begin{aligned}
\theta^{(s)}_{kj} &= (-1)^{s-k-1} \sum_{\substack{1\le i_1<\cdots<i_k\le s\\ i_1,\ldots,i_k\neq j}} \lambda_{i_1}\cdots\lambda_{i_k}, \quad k\neq 0\\
&= (-1)^{s-k-1} \left(\sum_{\substack{1\le i_1<\cdots<i_k=s\\ i_1,\ldots,i_k\neq j}} \lambda_{i_1}\cdots\lambda_{i_k} + \sum_{\substack{1\le i_1<\cdots<i_k\le s-1\\ i_1,\ldots,i_k\neq j}} \lambda_{i_1}\cdots\lambda_{i_k} \right)
\end{aligned}
$$

$$= (-1)^{s-k-1} \sum_{\substack{1\le i_1<\cdots<i_k\le s-1 \\ i_1,\ldots,i_k\neq j}} \lambda_{i_1}\cdots\lambda_{i_k}, \quad j\neq s,$$
$$= -\theta_{kj}^{(s-1)}, \quad j\neq s, \tag{E.89}$$

since $\lambda_s = 0$. (This holds when $k = 0$ too, by (E.84).) Also, since $\lambda_s = 0$,

$$\begin{aligned}\prod_{\substack{1\le i<m\le s \\ i,m\neq j}} (\lambda_i-\lambda_m) &= \prod_{\substack{1\le i<m=s \\ i\neq j}} (\lambda_i-\lambda_m) \prod_{\substack{1\le i<m\le s-1 \\ i,m\neq j}} (\lambda_i-\lambda_m) \\ &= \prod_{\substack{i=1 \\ i\neq j}}^{s-1} \lambda_i \prod_{\substack{1\le i<m\le s-1 \\ i,m\neq j}} (\lambda_i-\lambda_m), \quad j\neq s. \end{aligned} \tag{E.90}$$

Using (E.89) and (E.90) in (E.88) gives

$$\begin{aligned} A_{l,k}^{(s)} &= \sum_{j=1}^{s-1} (-1)^j \lambda_j^l (-\theta_{kj}^{(s-1)}) \prod_{\substack{i=1 \\ i\neq j}}^{s-1} \lambda_i \prod_{\substack{1\le i<m\le s-1 \\ i,m\neq j}} (\lambda_i-\lambda_m) \\ &= -\prod_{i=1}^{s-1}\lambda_i \sum_{j=1}^{s-1} (-1)^j \lambda_j^{l-1} \theta_{kj}^{(s-1)} \prod_{\substack{1\le i<m\le s-1 \\ i,m\neq j}} (\lambda_i-\lambda_m) \\ &= -A_{l-1,k}^{(s-1)} \prod_{i=1}^{s-1}\lambda_i, \end{aligned} \tag{E.91}$$

by (E.82). From (E.83) by induction on s and l,

$$A_{l-1,k}^{(s-1)} = \left\{ \begin{array}{l} 0, \quad k\ge 0, \quad k\neq (s-1)-(l-1)-1 = s-l-1, \\ (-1)^{s-1}\prod_{1\le i<j\le s-1}(\lambda_i-\lambda_j), \quad k = s-l-1, \end{array} \right\} \quad l = 0,\ldots,s. \tag{E.92}$$

We now use this in (E.91). If $k \neq s-l-1$ then $A_{l,k}^{(s)} = 0$, as required by (E.83). If $k = s-l-1$, then

$$A_{l,k}^{(s)} = -(-1)^{s-1} \prod_{1\le i<j\le s-1} (\lambda_i-\lambda_j) \prod_{i=1}^{s-1}\lambda_i = (-1)^s \prod_{1\le i<j\le s} (\lambda_i-\lambda_j),$$

(since $\lambda_s = 0$) also as required.

All that remains is to check the base cases of (E.83). When $s = 1$, $LS = -\lambda_1^l \theta_{k1}^{(1)}$ by (E.82). From (E.84), $\theta_{k1}^{(1)} = 0$, since the sum is empty. So $LS = 0$. But $RS = 0$ also, if $k \neq s-l-1 = -l$. If $k = -l$, then $k = l = 0$, which is handled in the next case.

When $l = 0$,

$$LS = \sum_{j=1}^{s} \theta_{kj}^{(s)} (-1)^j \prod_{\substack{1 \le i < m \le s \\ i,m \ne j}} (\lambda_i - \lambda_m), \tag{E.93}$$

and

$$RS = \begin{cases} 0, & k \ge 0,\ k \ne s-1, \\ (-1)^s \prod_{1 \le i < j \le s} (\lambda_i - \lambda_j), & k = s-1. \end{cases} \tag{E.94}$$

We will show that $LS = RS$ by induction on k. First, the base case, $k = 0$. From (E.84), $\theta_{0j}^{(s)} = (-1)^{s-1}$. Thus,

$$LS = (-1)^{s-1} \sum_{j=1}^{s} (-1)^j \prod_{\substack{1 \le i < m \le s \\ i,m \ne j}} (\lambda_i - \lambda_m). \tag{E.95}$$

If $k < s-1$, that is, $s > 1$, then $RS = 0$ by (E.94). To show $LS = 0$ as well, consider it as a polynomial in λ_s. It has degree $s-2$, so it suffices to show $LS = 0$ for $\lambda_s = \lambda_1, \ldots, \lambda_{s-1}$. Let $\lambda_s = \lambda_n$, $\lambda_n = \lambda_1, \ldots, \lambda_{s-1}$. Then the product in LS is zero unless $j = n$ or $j = s$. Now, using (E.86) in (E.95) gives

$$\begin{aligned} LS &= (-1)^{s-1} \left((-1)^n \prod_{\substack{1 \le i < m \le s \\ i,m \ne n}} (\lambda_i - \lambda_m) + (-1)^{s-1} \prod_{\substack{1 \le i < m \le s \\ i,m \ne s}} (\lambda_i - \lambda_m) \right) \\ &= (-1)^s \left((-1)^n (-1)^{s-n-1} \prod_{\substack{1 \le i < m \le s \\ i,m \ne s}} (\lambda_i - \lambda_m) + (-1)^s \prod_{\substack{1 \le i < m \le s \\ i,m \ne s}} (\lambda_i - \lambda_m) \right) \\ &= 0 \\ &= RS. \end{aligned}$$

If $k = s-1$ then $s = 1$, and the products in LS and RS are both empty. Thus, $LS = -1 = RS$. This establishes the base case, $k = 0$.

Now consider the general case, $s \ge 2$, $k \ge 1$. We first derive a recurrence for $\theta_{kj}^{(s)}$ for $k \ge 1$ from (E.84).

$$\theta_{kj}^{(s)} = (-1)^{s-1-k} \sum_{\substack{1 \le i_1 < \cdots < i_k \le s \\ i_1, \ldots, i_k \ne j}} \lambda_{i_1} \cdots \lambda_{i_k}$$

$$
\begin{aligned}
&= (-1)^{s-1-k}\left(\sum_{1\le i_1<\cdots<i_k\le s}\lambda_{i_1}\cdots\lambda_{i_k} - \sum_{\substack{1\le i_1<\cdots<i_k\le s\\ some\ i_l=j}}\lambda_{i_1}\cdots\lambda_{i_k}\right)\\
&= (-1)^{s-1-k}\left(\sum_{1\le i_1<\cdots<i_k\le s}\lambda_{i_1}\cdots\lambda_{i_k} - \lambda_j\sum_{\substack{1\le i_1<\cdots<i_{k-1}\le s\\ i_1,\ldots,i_{k-1}\neq j}}\lambda_{i_1}\cdots\lambda_{i_{k-1}}\right)\\
&= (-1)^{s-1-k}\left[(-1)^{s-k}\theta_k^{(s)} - \lambda_j(-1)^{s-1-(k-1)}\theta_{k-1,j}^{(s)}\right]\\
&= -\theta_k^{(s)} + \lambda_j\theta_{k-1,j}^{(s)}, \qquad \text{(E.96)}
\end{aligned}
$$

where

$$\theta_k^{(s)} \text{ is the coefficient of } x^k \text{ in } p(x) = \textstyle\prod_{j=1}^{s}(\lambda_j x - 1),\ k \ge 1.$$

Using this in (E.93) gives

$$
\begin{aligned}
LS &= \sum_{j=1}^{s}\theta_{kj}^{(s)}(-1)^j\prod_{\substack{1\le i<m\le s\\ i,m\neq j}}(\lambda_i-\lambda_m)\\
&= \sum_{j=1}^{s}\left(-\theta_k^{(s)} + \lambda_j\theta_{k-1,j}^{(s)}\right)(-1)^j\prod_{\substack{1\le i<m\le s\\ i,m\neq j}}(\lambda_i-\lambda_m)\\
&= -(-1)^j\theta_k^{(s)}\sum_{j=1}^{s}\prod_{\substack{1\le i<m\le s\\ i,m\neq j}}(\lambda_i-\lambda_m) \qquad \text{(E.97)}\\
&+ \sum_{j=1}^{s}(-1)^j\lambda_j\theta_{k-1,j}^{(s)}\prod_{\substack{1\le i<m\le s\\ i,m\neq j}}(\lambda_i-\lambda_m). \qquad \text{(E.98)}
\end{aligned}
$$

Now (E.97)=0 by the base case, (E.95)=0, and hence (E.93)=(E.98). Therefore we must show that

$$\text{(E.98)} = \text{(E.94)}. \qquad \text{(E.99)}$$

We use induction on s. Recall that $s \ge 2$ and $k \ge 1$, so the base case ($s = 2$) is

$$
\begin{aligned}
LS &= -\lambda_1\theta_{k-1,1}^{(2)} + \lambda_2\theta_{k-1,2}^{(2)},\\
RS &= \begin{cases} 0, & k \ge 2,\\ \lambda_1 - \lambda_2, & k = 1. \end{cases}
\end{aligned}
$$

From (E.84), $\theta_{k-1,1}^{(2)} = \theta_{k-1,2}^{(2)} = -1$ when $k = 1$ and 0 for all $k > 1$. For $k = 1$ then, $LS = \lambda_1 - \lambda_2 = RS$, while for $k > 1$, $LS = 0 = RS$. We now handle the general case,

$s > 2$. In (E.98), the $j = s$ term of the sum is zero because $\lambda_s = 0$. Also, by (E.89), $\theta^{(s)}_{k-1,j} = -\theta^{(s-1)}_{k-1,j},\ j \neq s$. Thus,

$$
\begin{aligned}
\text{(E.98)} &= -\sum_{j=1}^{s-1}(-1)^j \lambda_j \theta^{(s-1)}_{k-1,j} \prod_{\substack{1\le i<m\le s\\ i,m\neq j}} (\lambda_i - \lambda_m) \\
&= -\sum_{j=1}^{s-1}(-1)^j \lambda_j \theta^{(s-1)}_{k-1,j} \prod_{\substack{1\le i<m\le s-1\\ i,m\neq j}} (\lambda_i - \lambda_m) \prod_{\substack{1\le i<m=s\\ i,m\neq j}} \lambda_i \\
&= -\left(\prod_{i=1}^{s-1}\lambda_i\right) \sum_{j=1}^{s-1}(-1)^j \lambda_j \theta^{(s-1)}_{k-1,j} \prod_{\substack{1\le i<m\le s-1\\ i,m\neq j}} (\lambda_i - \lambda_m) \\
&= -\left(\prod_{i=1}^{s-1}\lambda_i\right) \left\{ \begin{array}{ll} 0, & k-1 \neq s-2, \\ (-1)^{s-1}\prod_{1\le i<j\le s-1}(\lambda_i-\lambda_j), & k-1 = s-2, \end{array} \right\} \\
&\qquad\qquad \text{by induction on (E.99)} \\
&= \left\{ \begin{array}{ll} 0, & k \neq s-1 \\ (-1)^{s}\prod_{1\le i<j\le s}(\lambda_i-\lambda_j), & k = s-1 \end{array} \right. \\
&= \text{(E.94)},
\end{aligned}
$$

since $\lambda_s = 0$. This completes the proof of the Lemma. □

Appendix F

Combinatorial Identities

The following combinatorial identities are taken from [19]. (The numbers given in brackets at the left are the original equation numbers from [19]).

(1.13)

$$\sum_{k=0}^{n}(-1)^k \left(\begin{array}{c} n \\ k \end{array} \right) k^j = \left\{ \begin{array}{ll} 0, & 0 \leq j < n, \\ (-1)^n n!, & j = n. \end{array} \right. \tag{F.1}$$

(1.24)

$$\sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) = 2^n \tag{F.2}$$

(1.25)

$$\sum_{k=0}^{n}(-1)^k \left(\begin{array}{c} n \\ k \end{array} \right) = \left\{ \begin{array}{ll} 0, & n \neq 0, \\ 1, & n = 0. \end{array} \right. \tag{F.3}$$

(3.1)

$$\sum_{k=0}^{n} \left(\begin{array}{c} x \\ k \end{array} \right) \left(\begin{array}{c} y \\ n-k \end{array} \right) = \left(\begin{array}{c} x+y \\ n \end{array} \right) \tag{F.4}$$

(3.4)

$$\sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) \left(\begin{array}{c} x+y-n \\ x-k \end{array} \right) = \left(\begin{array}{c} x+y \\ x \end{array} \right) \tag{F.5}$$

(3.20)

$$\sum_{k=0}^{n} \left(\begin{array}{c} n \\ k \end{array} \right) \left(\begin{array}{c} x \\ k+r \end{array} \right) = \left(\begin{array}{c} n+x \\ n+r \end{array} \right) \tag{F.6}$$

(3.47)

$$\sum_{k=0}^{n}(-1)^k \left(\begin{array}{c} n \\ k \end{array} \right) \left(\begin{array}{c} x+k \\ r \end{array} \right) = (-1)^n \left(\begin{array}{c} x \\ r-n \end{array} \right) \tag{F.7}$$

(3.48)

$$\sum_{k=0}^{n}(-1)^k \left(\begin{array}{c} n \\ k \end{array} \right) \left(\begin{array}{c} x+k \\ r+k \end{array} \right) = (-1)^n \left(\begin{array}{c} x \\ n+r \end{array} \right) \tag{F.8}$$

Bibliography

[1] A. Bellen. PADETEST - a set of real-life test differential equations for parallel computing. Technical Report 103, Università di Trieste, 1992.

[2] P. N. Brown, G. D. Byrne, and A. C. Hindmarsh. VODE: A variable-coefficient ODE solver. *SIAM J. Sci. Stat. Comput.*, 10(5):1038–1051, 1989.

[3] K. Burrage. Order and stability properties of explicit multivalue methods. *Appl. Numer. Math.*, 1:363, 1985.

[4] K. Burrage. Order properties of implicit multivalue methods for ordinary differential equations. *IMA J. Numer. Anal.*, 8:43, 1988.

[5] K. Burrage. *Parallel and Sequential Methods for Ordinary Differential Equations*. Oxford University Press, 1995.

[6] K. Burrage and F. H. Chipman. Construction of A-stable diagonally implicit multivalue methods. *SIAM J. Numer. Anal.*, 26(2):397, 1989.

[7] K. Burrage and P. Moss. Simplifying assumptions for the order of partitioned multivalue methods. *BIT*, 20:452, 1980.

[8] J. C. Butcher. DIMSIMs — Diagonally implicit multi-stage integration methods. Seminar given at the University of Toronto, Nov. 26, 1991.

[9] J. C. Butcher. General linear method: A survey. *Appl. Numer. Math.*, 1:273, 1985.

[10] J. C. Butcher. *The Numerical Analysis of Ordinary Differential Equations*. John Wiley & Sons, 1987.

[11] J. C. Butcher. Diagonally-implicit multi-stage integration methods. *Appl. Numer. Math.*, 11:347–363, 1993.

[12] J. C. Butcher. General linear methods for the parallel solution of ordinary differential equations. Technical report, University of Auckland, 1993.

[13] J. C. Butcher. A transformation for the analysis of DIMSIMs. *BIT*, 34:25–32, 1994.

[14] J. C. Butcher and Z. Jackiewicz. Diagonally implicit general linear methods for ordinary differential equations. *BIT*, 33:452–472, 1993.

[15] P. Chartier. *Parallélisme dans la résolution numérique des problèmes de valeur initiale pour les équations différentielles ordinaires et algébriques*. PhD thesis, l'Université de Rennes 1, 1993.

[16] G. J. Cooper. The order of convergence of general linear methods for ordinary differential equations. *SIAM J. Numer. Anal.*, 15(4):643, Aug. 1978.

[17] W. H. Enright. Improving the efficiency of matrix operations in the numerical solution of stiff ordinary differential equations. *ACM Trans. Math. Softw.*, 4(2):127–136, June 1978.

[18] W. H. Enright and J. D. Pryce. Two FORTRAN packages for assessing initial value methods. Technical Report 167/83, University of Toronto, 1983.

[19] H. W. Gould. *Combinatorial Identities*. Manuscript, 1972.

[20] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I*. Springer-Verlag, 1987.

[21] E. Hairer and G. Wanner. Multistep-multistage-multiderivative methods for ordinary differential equations. *Computing*, 11:287, 1973.

[22] E. Hairer and G. Wanner. On the Butcher group and general multi-value methods. *Computing*, 13:1, 1974.

[23] E. Hairer, G. Wanner, and S. P. Nørsett. Order stars and stability theorems. *BIT*, 18:475, 1978.

[24] D. Higham. Everything you wanted to know about aSymptotic Error eXpansions but were afraid to ask. Course handout, May 1991.

[25] A. C. Hindmarsh. LSODE and LSODI, two new initial value ordinary differential equations solvers. *ACM SIGNUM Newsletter*, 15(4):10–11, 1980.

[26] K. R. Jackson and S. P. Nørsett. The potential for parallelism in Runge-Kutta methods. Part I: Runge-Kutta formulas in standard form. *SIAM J. Numer. Anal.*, 32:49–82, 1995.

[27] A. Prothero and A. Robinson. On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations. *Math. Comp.*, 28:145–162, Jan. 1974.

[28] P. Sharp. Numerical testing of explicit Runge-Kutta pairs. Technical Report 216/88, University of Toronto, 1988.

[29] R. Skeel. Analysis of fixed-stepsize methods. *SIAM J. Numer. Anal.*, 13(5):664, Oct. 1976.

[30] Waterloo Maple Software. Maple V. A symbolic algebraic manipulation package.

[31] J. H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, 1965.