7-2020

Optimal Allocation of Two Resources in Annual Plants

David McMorris
University of Nebraska - Lincoln, david.mcmorris@huskers.unl.edu

OPTIMAL ALLOCATION OF TWO RESOURCES IN ANNUAL PLANTS

by

David McMorris

A DISSERTATION

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Doctor of Philosophy

Major: Mathematics

Under the Supervision of Professor Glenn Ledder

Lincoln, Nebraska

August, 2020

OPTIMAL ALLOCATION OF TWO RESOURCES IN ANNUAL PLANTS

David McMorris, Ph.D.

University of Nebraska, 2020

Advisor: Glenn Ledder

The fitness of an annual plant can be thought of as how much fruit is produced by the end of its growing season. Under the assumption that annual plants grow to maximize fitness, we can use techniques from optimal control theory to understand this process. We introduce two models for resource allocation in annual plants which extend classical work by Iwasa and Roughgarden to a case where both carbohydrates and mineral nutrients are allocated to shoots, roots, and fruits in annual plants. In each case, we use optimal control theory to determine the optimal resource allocation strategy for the plant throughout its growing season as well as develop numerical schemes to implement the models in MATLAB. Our results suggest that what is optimal for an individual plant is highly dependent on initial conditions, and optimal growth has the effect of driving a wide range of initial conditions toward common configurations of biomass by the end of a growing season.

ACKNOWLEDGMENTS

I would like to start by thanking my advisor, Glenn Ledder, for your years of teaching and guidance, from the applied mathematics course you taught my first year of graduate school to the completion of my dissertation. You've challenged me to approach problems more as a modeler and less as a mathematician, and helped me become a more self-sufficient researcher than I would have believed possible six years ago. I am immensely grateful for the time and energy you have invested in advising me over the years and in particular during these last months of quarantined dissertation writing.

I would also like to thank the rest of my committee, Clay Cressler, Huijing Du, and Richard Rebarber, for the time spent reviewing my work, as well as everything you have done to make this possible. Clay, our discussions about plant biology have been an invaluable resource in making sense of the models, and it's been great working with another Hope College alum.

To the rest of the UNL Math Department, I am particularly grateful for the strong sense of community we have here, and for everything you all do to make this a place I will dearly miss. I would like to thank Nathan Wakefield and Allan Donsig for your contributions for my development as an educator, and the breadth of teaching experiences I have had at UNL as a result. I would also like to thank all my professors for helping me develop as a mathematician. Thank you also to Marilyn Johnson and the rest of the office staff.

Thank you to my fellow graduate students, and especially those in my cohort, for taking this journey with me from the beginning. To my officemates over the years, Andrew Connner, Mohsen Gheibi, and Ash DeClerk, thank you for all the laughs, good conversations, and for letting me take over the whiteboard so often.

Special thanks to my undergraduate advisor at Hope College, Brian Yurk, for first

introducing me to mathematical biology and research in mathematics, and encouraging me to pursue a doctorate.

I would like to thank my parents, Marc and Jan, for encouraging my interest in math and science from a young age, and for always being there for me throughout my years of education. Thank you also to my siblings, Katie, John, and Paul, as well as the rest of my support system of family and friends.

To my wife, Marla Williams, you are amazing, and I am so grateful for all your love and support these past few years. You have believed in me when I didn't think I could reach this point, and have been a constant source of encouragement through what have been some of the most difficult years of my life. I have been especially thankful for your presence during quarantine these last several months as we finished our dissertations together, and am excited to see where life takes us next.

Table of Contents

CHAPTER 1

INTRODUCTION

Plant life history theory is generally concerned with the survival and reproductive strategies plants employ throughout their life cycle, as well as how these processes influence population dynamics. The question of how plants allocate resources, e.g. C, N, P, and in some models, biomass, and what drives these allocation rules, is central to this pursuit. There are several schools of thought which seek to provide a framework for understanding these allocation patterns. One school of thought is that allocation rules should ultimately be the result of natural selection, and therefore resource allocation should optimize fitness in some sense (see [5, 14]). Another framework views biomass allocation as following certain allometric scaling relationships (see [3, 9]), and yet another views allocation not through the lens of an individual organism or specific genome, but rather from a game-theoretical perspective in which allocation rules are driven by competition, and follow an evolutionary stable strategy (see [2]). For a more complete review of allocation theory, we refer the reader to Ledder et al. [7] or Poorter et al. [13].

In this dissertation, we take the viewpoint that resource allocation, in annual plants specifically, should serve to optimize overall fitness. It's important to note that, while patterns of growth consistent with optimal allocation have been observed to some extent (see e.g. [9]), even if this is not universally true it is still important

to have a theory of optimal growth for comparison with observed behavior. Whereas previous work has focused primarily on optimal allocation of a single resource, be it carbon or biomass, and ignore the role of mineral nutrients, our work seeks to develop a theory that acknowledges the importance of both carbon and mineral nutrients. This is in line with the functional equilibrium hypothesis, which states that optimal growth occurs when resources are allocated in such a way that no single resource is any more limiting than any other (see [1, 12, 15]).

This work is primarily an extension of classical work by Iwasa and Roughgarden [5], who considered a model in which photosynthate (C) was allocated to shoots, roots, and fruits, with the objective of optimizing fruit yield. Their work, which we review in Section 1.1.2, uses optimal control theory to determine that fruit yield is maximized by a three-phase growth path, characterized by an initial phase of shoot-only or root-only growth, a period of 'balanced growth' during which shoots and roots grow simultaneously, and ultimately a period of fruit-only growth at the end of the growing season. We don't assume the reader has any particular knowledge of optimal control theory. In Section 1.1.1 we outline some of the basic theory, and throughout this dissertation we introduce any additional control theoretical results as they become relevant.

We present two models which extend the work of Iwasa and Roughgarden to a case which incorporates the role of mineral nutrients directly into the model. In particular, we model the allocation of carbon and nitrogen in an annual plant with the objective of optimizing fruit yield, and use optimal control theory to determine the optimal allocation strategies. The first model, discussed in Chapter 2, considers a case where, although we incorporate a second resource, fruits remain carbon-only as in [5]. This simplification allows us to obtain mathematical results which we use to guide the analysis of our second model in Chapter 3, which removes this simplification

and incorporates nitrogen into the fruits as well. We find, among other things, that this addition results in four, rather than three, phases of growth. This additional phase consists of a period of mixed vegetative/reproductive growth, during which the fruits and either the shoots or roots grow simultaneously, depending on the C:N ratios in each organ. Furthermore, our results indicate that what is optimal for one plant may not be optimal for another, and optimal growth is largely dependent on initial conditions. Additionally, our results suggest that the presence of this range of optimal strategies may have the ability to eliminate a high degree of variation in a population, thus driving the population toward common sizes and optimal yields. The question remains, however, as to whether plants actually have this degree of strategic plasticity.

This dissertation is structured from here on in four parts. In Chapter 1, we present background material on optimal control theory and the work conducted by Iwasa and Roughgarden in [5]. In Chapters 2 and 3, we present our work with the two models we have just described for optimal allocation of two resources in annual plants. Appendix A contains control theoretical results we derive for the specific types of control problems we encounter herein. Appendix B contains mathematical details which we deemed necessary to include, but not enlightening to the reader. Lastly, Appendix C contains the MATLAB code used to simulate each model.

1.1 Background Material

1.1.1 Optimal Control Theory

We will give a concise overview of basic optimal control theory and the Pontryagin Maximum Principle, and refer the reader to [8] for a more thorough discussion of the theory and application to problems in biology. In some sense, the simplest problem

in optimal control theory is of the form

$$\begin{aligned} &\max_u \int_{t_0}^{t_1} f(t,x,u)\,dt \\ \text{subject to:}\quad &x'(t) = g(t,x,u) \\ &x(t_0) = x_0. \end{aligned} \tag{1.1}$$

Here we call $x(t)$ the state, and $u(t)$ the control, and the goal is to find the pair that maximizes the functional, subject to the given constraints. This is typically accomplished with the aid of a set of necessary conditions, which an optimal pair (x,u) must satisfy. Note that, whereas x must satisfy a differential equation, and is thus continuous, we make no such assumption about u. We do, however, assume that f and g are continuously differentiable in each argument, so that the control is, at minimum, piecewise continuous. We will omit the technical details, and instead opt for an overview of the procedure for solving these types of problems.

The solution process begins with forming a Hamiltonian, given by

$$H(t,x,u,\lambda) = f(t,x,u) + \lambda(t)g(t,x,u), \tag{1.2}$$

where $\lambda(t)$ is a piecewise differentiable function, referred to as either the adjoint or costate. For an optimal pair (x^*,u^*), the adjoint must satisfy

$$\lambda'(t) = -\frac{\partial H}{\partial x}(t,x^*,u^*,\lambda), \qquad \lambda(t_1) = 0. \tag{1.3}$$

The necessary condition that the optimal control must satisfy is given by

$$\frac{\partial H}{\partial u}(t,x^*,u^*,\lambda) = 0. \tag{1.4}$$

It may be helpful to think of optimal control problems as infinite-dimensional analogs of constrained optimization problems from multivariable calculus, where the adjoint plays a similar role to that of the Lagrange multiplier. This process is formalized in Pontryagin's Maximum Principle [11], a version of which appears in [8, Theorem 1.2] for the optimal control problem (1.1).

Theorem 1.1 (Pontryagin's Maximum Principle)**.** *If u^* and x^* are optimal for problem* (1.1)*, then there exists a piecewise differentiable adjoint variable $\lambda(t)$ such that*

$$H(t, x^*(t), u(t), \lambda(t)) \leq H(t, x^*(t), u^*(t), \lambda(t))$$

for all controls u at each time t, where the Hamiltonian H is

$$H = f(t, x(t), u(t)) + \lambda(t) g(t, x(t), u(t)),$$

and

$$\begin{aligned} \lambda'(t) &= -\frac{\partial H(t, x^*(t), u^*(t), \lambda(t))}{\partial x} \\ \lambda(t_1) &= 0. \end{aligned}$$

It is worth mentioning one additional class of optimal control problems, involving problems with several states and several bounded controls. Here, and throughout the dissertation, we will use the notation $\vec{x}(t) = \langle x_1(t), x_2(t), \ldots, x_n(t) \rangle$ to represent the function $\vec{x}\colon \mathbb{R} \to \mathbb{R}^n$ and $\vec{g}(t, \vec{x}, \vec{u}) = \langle g_1(t, \vec{x}, \vec{u}), g_2(t, \vec{x}, \vec{u}), \ldots, g_n(t, \vec{x}, \vec{u}) \rangle$ to represent the function $\vec{g}\colon \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$. Of particular importance to us are problems of

the form

$$\begin{aligned} &\max_{\vec{u}} \int_{t_0}^{t_1} f(t, \vec{x}, \vec{u})\, dt \\ \text{subject to:} \quad &\vec{x}'(t) = \vec{g}(t, \vec{x}, \vec{u}), \quad \vec{x}(t_0) = \vec{x}_0 \\ &a_k \leq u_k \leq b_k. \end{aligned} \tag{1.5}$$

Here we form a Hamiltonian with n adjoints, one for each of the states:

$$H(t, \vec{x}, \vec{u}, \vec{\lambda}) = f(t, \vec{x}, \vec{u}) + \vec{\lambda}(t) \cdot \vec{g}(t, \vec{x}, \vec{u}), \tag{1.6}$$

and (1.3) is replaced with the following:

$$\lambda_i'(t) = -\frac{\partial H}{\partial x_i}(t, \vec{x}^*, \vec{u}^*, \vec{\lambda}), \quad \lambda_i(t_1) = 0 \quad \text{for} \quad i = 1, \ldots, n. \tag{1.7}$$

Rather than (1.4), the necessary conditions for optimal controls for (1.5) state that the following must hold at $(\vec{x}^*, \vec{u}^*)$.

$$\begin{cases} u_k = a_k, & \text{if } \frac{\partial H}{\partial u_k} < 0 \\ a_k \leq u_k \leq b_k, & \text{if } \frac{\partial H}{\partial u_k} = 0 \\ u_k = b_k, & \text{if } \frac{\partial H}{\partial u_k} > 0 \end{cases} \tag{1.8}$$

The types of control problems which arise from the modeling in this dissertation are an extension of the class of problems given by (1.5), with additional constraints on the controls. We will discuss how the necessary conditions change in these cases as they are introduced, as well as work through the derivation of the necessary conditions in Appendix A.

1.1.2 Iwasa and Roughgarden

The work presented in this dissertation primarily serves as an extension of a classical work by Iwasa and Roughgarden [5], in which they model optimal allocation of photosynthate (carbohydrates) in a plant composed of shoots ($S(t)$), roots ($R(t)$), and fruits ($F(t)$). They assumed that shoots and roots work together to produce photosynthate, some fraction of which is allocated to each of the three organs, with the objective of maximizing fruit production by the end of the growing season, assumed to be of fixed length T.

Let $g(S, R)$ be the rate of photosynthate production, and controls $u_0(t), u_1(t)$, and $u_2(t)$ be the fractions of photosynthate allocated to fruits, shoots, and roots at time t, respectively. The model they used is given by

$$\frac{dS}{dt} = u_1(t)g(S,R), \quad S(0) = S_0 \tag{1.9}$$

$$\frac{dR}{dt} = u_2(t)g(S,R), \quad R(0) = R_0 \tag{1.10}$$

$$\frac{dF}{dt} = u_0(t)g(S,R), \quad F(0) = 0 \tag{1.11}$$

with the optimality condition

$$\max_{\vec{u}} F(T) = \max_{\vec{u}} \int_0^T \frac{dF}{dt}\, dt = \max_{\vec{u}} \int_0^T u_0(t)g(S,R)\, dt. \tag{1.12}$$

In order to make the model biologically realistic, they also required the controls to be bounded in $[0, 1]$ and sum to one.

In line with what we discussed in Section 1.1.1, Iwasa and Roughgarden solved this optimal control problem by means of Pontryagin's Maximum Principle, and the

use of the Hamiltonian

$$H = u_0(t)g(S,R) + \lambda_1(t)u_1(t)g(S,R) + \lambda_2(t)u_2(t)g(S,R) \tag{1.13}$$

where the adjoints λ_1 and λ_2 satisfy

$$\begin{aligned}
\frac{d\lambda_1}{dt} &= -u_0(t)g_S(S,R) - \lambda_1(t)u_1(t)g_S(S,R) - \lambda_2(t)u_2(t)g_S(S,R), \quad \lambda_1(T) = 0 \\
\frac{d\lambda_2}{dt} &= -u_0(t)g_R(S,R) - \lambda_1(t)u_1(t)g_R(S,R) - \lambda_2(t)u_2(t)g_R(S,R), \quad \lambda_2(T) = 0.
\end{aligned}$$

Iwasa and Roughgarden found that the optimal strategy for a plant, under their model, was to allocate carbohydrates to the organ with the greatest ability to contribute to the final fruit yield, that is, the organ with the highest marginal value. They were able to associate the marginal values of shoots and roots with the adjoints $\lambda_1(t)$ and $\lambda_2(t)$. This means that at time t, the value of $\lambda_1(t)$ is the amount of additional units of fruits that will be present at time T, given a unit investment of carbohydrates into the shoots at time t, and likewise for $\lambda_2(t)$ and the roots. Furthermore, the marginal value of fruits is always 1, because a unit of carbohydrate invested in the fruits cannot be compounded for increased fruit production. Note that this strategy corresponds to maximizing the Hamiltonian (1.13) pointwise, which is to be expected from Pontryagin's Maximum Principle.

This rule for allocation results in a three-phase optimal growth path, an example of which is shown in Figure 1.1. The plant first addresses any deficiency in either shoots or roots and converges to a phase during which growing both shoots and roots together is optimal. Iwasa and Roughgarden termed this phase of mixed growth 'balanced growth.' Following balanced growth, there is a phase during which only the fruits grow, which begins when both the marginal values of shoots and roots drop

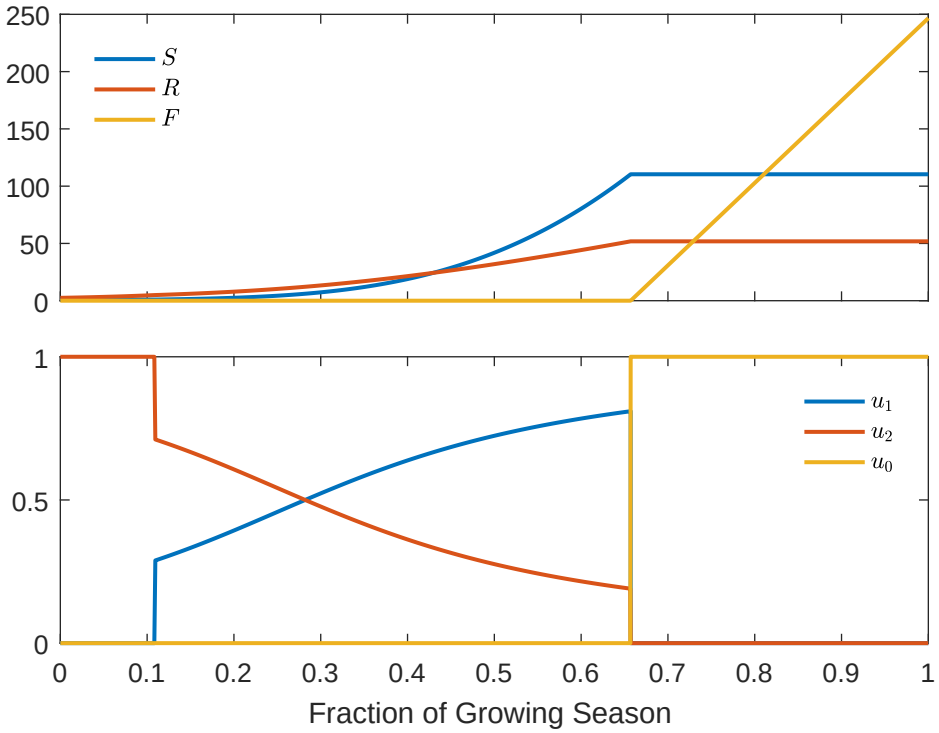

Figure 1.1: Typical three-phase growth pattern from Iwasa and Roughgarden's model. The top plot contains shoots, root, and fruits in units of carbon and the bottom plot contains the (dimensionless) controls.

below that of fruits, i.e. both $\lambda_1(t)$ and $\lambda_2(t)$ drop below 1.

In Figure 1.1, we can see an example of this three-phase pattern of growth. Note that here the x-axis represents the fraction of the growing season rather than the actual growing time. The plant begins with a short period of root-only growth, followed by a period of balanced growth, and lastly a period of fruit-only growth at the end of the season. Furthermore, note that during balanced growth we see a gradual increase in allocation to shoots and corresponding decrease in allocation to roots, reflecting the fact that in this simulation g was chosen so that the amount of shoots was the limiting factor in photosynthesis. So, while the plant initial invested in roots to make up for the deficiency, during balanced growth it transitioned toward investing more in shoots to avoid photosynthesis limitation.

CHAPTER 2

FIRST MODEL: CARBON-ONLY FRUITS

2.1 Introduction

The first model for resource allocation in annual plants we will discuss extends the work of Iwasa and Roughgarden [5] to a scenario where the growth of the roots and shoots relies on two resources, rather than only one. For the sake of convenience, we will use the terms 'carbon' and 'nitrogen' to refer to more complicated classes of carbohydrates and mineral nutrients. As a simplifying assumption, we assume that fruits are only reliant on carbon for growth.

This assumption makes the mathematical analysis more feasible, and aligns biologically with plants that encase small seeds in large carbon-rich fruits. While this may match our colloquial definition of fruits, e.g. watermelons, it is not an accurate representation of many types of annual plants for which the term 'fruits' refers to seeds, nuts, etc., and for which nitrogen content is not negligible. In Chapter 3 we will discuss a second model which extends our first model to a case where the nitrogen content in fruits is accounted for.

In this chapter, we will begin with a description of the model and the optimal control framework in Section 2.2. Next, we will go through the mathematical results we have obtained. These are split into two sections - Section 2.3 which outlines

the four-phase structure of the optimal solution, and Section 2.4 which discusses the dynamics within each phase as well as results about how the plant transitions from one phase to the next. In Section 2.5 we will present the numerical scheme used to simulate the model in MATLAB, followed by a description of our numerical results in Section 2.6 and ultimately a discussion in Section 2.7.

2.2 A Description of the Model

In this section we will introduce the first model for resource allocation in annual plants as well as the optimal control framework we used for determining the optimal growth trajectory for maximal fruit growth. We will begin by introducing relevant notation and conceptualizations, followed by the model equations.

2.2.1 Model Setup

We consider an annual plant with three organs - shoots, roots, and fruits. Shoots consist of all above-ground vegetative biomass and roots all below-ground vegetative biomass. Fruits refer to any reproductive biomass, be it in the form of colloquial 'fruits', nuts, seeds, etc. Biomass of each organ is measured in units of 'carbon' where we used 'carbon' as a catch-all term for carbohydrates produced by the shoots. The functions $S(t), R(t)$, and $F(t)$ give the biomass of shoots, roots, and fruits, respectively, at time t throughout a fixed growing season $[0, T]$. We assume that the plant relies on two resources, which for simplicity we refer to as 'carbon' and 'nitrogen,' though as previously mentioned we use these terms loosely to refer to more complicated classes of carbohydrates and soil nutrients. We assume that carbon is fixed by the shoots at a rate of $C(S)$ and nitrogen is absorbed by the roots at a rate of $N(R)$. Note that this is choice we made to simplify the model, as in reality the rate of

carbon fixation depends on both shoots and roots via transpiration. Throughout the growing season, fractions of carbon $u_{1C}(t), u_{2C}(t)$, and $u_{0C}(t)$ are allocated to shoots, roots, and fruits, respectively at time t, and because this model focuses on the case in which fruits are nitrogen-deficient, we also have fractions of nitrogen $u_{1N}(t)$ and $u_{2N}(t)$, which are allocated to the shoots and roots, respectively, at time t. The resources pass through a synthesizing unit (SU) in each organ, where they are converted into biomass. Since we assume that fruits are carbon-only we assume perfect conversion from resources to biomass. For the shoots and roots, we use Kooijman's parallel complementary synthesizing unit (PCSU) function from [6], employing the same simplification seen in [7, Appendix A], given by

$$g(A, B) = \frac{A^2B + AB^2}{A^2 + AB + B^2} \tag{2.1}$$

to provide the rate of tissue production when resources are provided at rates A and B. As the tissue is measured in units of carbon, we need conversion factors ν_S and ν_R which give the fixed stoichiometric C:N ratios in the shoots and roots, respectively. We specify initial conditions $S(0) = S_0, R(0) = R_0$, and $F(0) = 0$ which leads to the following model:

$$\frac{dS}{dt} = g(u_{1C}C, \nu_S u_{1N} N), \quad S(0) = S_0 \tag{2.2}$$

$$\frac{dR}{dt} = g(u_{2C}C, \nu_R u_{2N} N), \quad R(0) = R_0 \tag{2.3}$$

$$\frac{dF}{dt} = u_{0C}C, \qquad F(0) = 0 \tag{2.4}$$

Note that we suppress the arguments in most functions for convenience. This model is shown schematically in Figure 2.1,

Because the functions $u_{0C}, u_{1C}, u_{2C}, u_{1N}$, and u_{2N} represent fractions of carbon

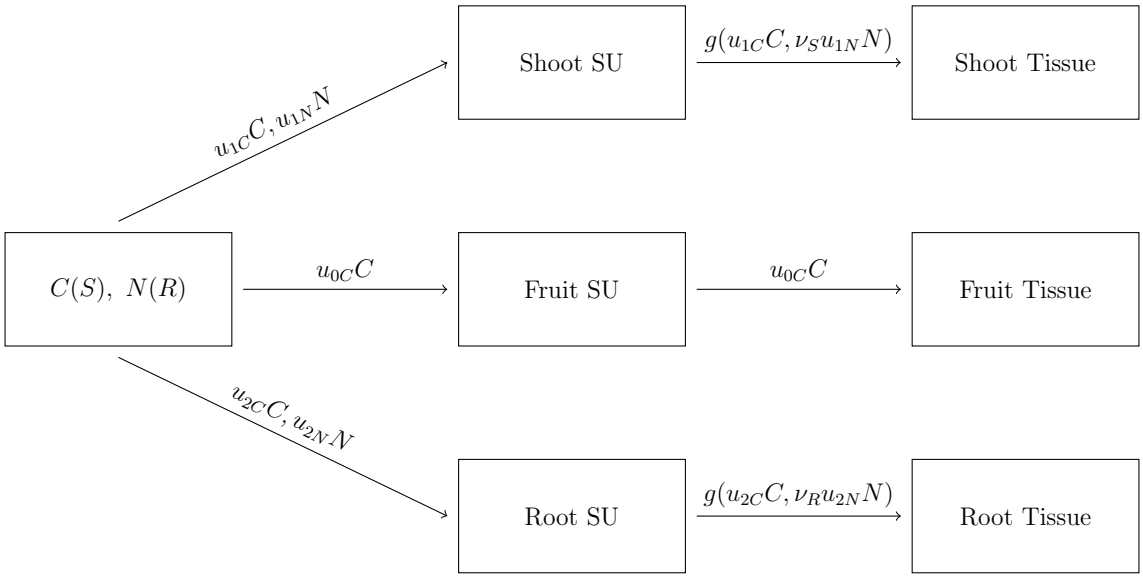

Figure 2.1: Model Schematic

and nitrogen, respectively, we impose several restrictions. First, we require that each function be piecewise continuous and bounded between 0 and 1. Furthermore, as we assume full utilization of each resource, we assume that

$$u_{0C} + u_{1C} + u_{2C} = 1 = u_{1N} + u_{2N} \tag{2.5}$$

at all times $t \in [0, T]$. Additionally, to be biologically realistic, we assume that the plant's capacity to 'collect' resources increases continuously with biomass, meaning that we require both C and N to be continuously differentiable, and

$$\frac{dC}{dS} > 0 \quad \text{and} \quad \frac{dN}{dR} > 0. \tag{2.6}$$

We also assume that the plant experiences possibly diminishing returns with increased biomass, meaning that

$$\frac{d^2C}{dS^2} \leq 0 \quad \text{and} \quad \frac{d^2N}{dR^2} \leq 0.$$

2.2.2 The PCSU

Before we introduce the optimal control framework we will use for determining the optimal resource allocation strategy, there are several important features of the PCSU that merit discussion. As introduced in (2.1), for resource fluxes A and B, the PCSU function

$$g(A,B) = \frac{A^2B + AB^2}{A^2 + AB + B^2}, \qquad \text{((2.1) revisited)}$$

gives the rate of tissue production. Note that when both resources are received by the SU at the same rate, we have

$$g(A,A) = \frac{A^3 + A^3}{A^2 + A^2 + A^2} = \frac{2A^3}{3A^2} = \frac{2}{3}A, \tag{2.7}$$

in which case we observe that the output rate is 2/3rds the input rate. We call this the efficiency of the SU. Furthermore, note that when only one resource is present, the tissue production rate is zero as $g(A,0) = 0 = g(0,B)$ for non-zero A and B, respectively. Additionally, this is still true when neither resource is present. Indeed, using the polar transformations $A = r\cos(\theta)$ and $B = r\sin(\theta)$ we see

$$\begin{aligned}
\lim_{(A,B)\to(0,0)} g(A,B) &= \lim_{r\to 0} \frac{r^3\cos^2(\theta)\sin(\theta) + r^3\cos(\theta)\sin^2(\theta)}{r^2 + r^2\cos(\theta)\sin(\theta)} \\
&= \lim_{r\to 0} r\frac{\cos^2(\theta)\sin(\theta) + \cos(\theta)\sin^2(\theta)}{1 + \frac{1}{2}\sin(2\theta)} \\
&= 0.
\end{aligned} \tag{2.8}$$

Because of this, we will take g to be the continuous extension of this function to the origin such that $g(0,0) = 0$.

Now, by changing variables to either $z_B = \frac{B}{A}$ or $z_A = \frac{A}{B}$, we can rewrite g as

$$g(A, B) = AG(z_B) = BG(z_A), \tag{2.9}$$

where

$$G(z) = \frac{z(1+z)}{1+z+z^2}. \tag{2.10}$$

The fact that we can represent the PCSU function in this manner will aid in our analysis by restricting the nonlinearities present to a function of a single variable. By the above discussion, we observe that although z_B and z_A may be undefined when either A or B or both are zero, both $AG(z_B)$ and $BG(z_A)$ are zero when at least one of A or B is zero. It is important to note, however, that

$$\lim_{(A,B)\to(0,0)} G(z_B) \quad \text{and} \quad \lim_{(A,B)\to(0,0)} G(z_A)$$

are both undefined without the additional factor of A or B, respectively. Additionally, it will facilitate late analysis of the model to note that

$$G'(z) = \frac{1+2z}{(1+z+z^2)^2}, \tag{2.11}$$

and as before neither

$$\lim_{(A,B)\to(0,0)} G'(z_B) \quad \text{nor} \quad \lim_{(A,B)\to(0,0)} G'(z_A)$$

exists. However, a similar argument to (2.8) can be used to verify that both $AG'(z_B)$ and $BG'(z_A)$ are zero when either A or B is zero. Furthermore, it is worth noting

that both G and G' are bounded between 0 and 1 and that

$$G(0) = 0, \quad G'(0) = 1, \quad \lim_{z\to\infty} G(z) = 1, \quad \lim_{z\to\infty} G'(z) = 0 \tag{2.12}$$

2.2.2.1 PCSU Identities

There are two identities related to the PCSU function and its derivatives that will be repeatedly cited in later analysis, and so bear mentioning here. We will start with the following notation:

$$G_2(z) = G'(1/z) = \frac{z^3(2+z)}{(1+z+z^2)^2}. \tag{2.13}$$

Employing this notation, we have the first identity.

Proposition 2.1.

$$G(z) - zG'(z) = G_2(z). \tag{2.14}$$

Proof. Using (2.10), (2.11), and (2.13) we have

$$\begin{aligned}
G(z) - zG'(z) &= \frac{z(1+z)}{1+z+z^2} - \frac{z(1+2z)}{(1+z+z^2)^2} \\
&= \frac{(z+z^2)(1+z+z^2) - z - 2z^2}{1+z+z^2+z+z^2+z^3+z^2+z^3+z^4} \\
&= \frac{z+z^2+z^3+z^2+z^3+z^4-z-2z^2}{1+2z+3z^2+2z^3+z^4} \\
&= \frac{2z^3+z^4}{1+2z+3z^2+2z^3+z^4} \\
&= \frac{z^3(2+z)}{(1+z+z^2)^2} \\
&= G_2(z).
\end{aligned}$$

□

The second useful identity is as follows.

Proposition 2.2.

$$G_2'(z) = -zG''(z). \tag{2.15}$$

Proof. Differentiating (2.14) from Proposition 2.1, we have

$$G_2'(z) = \frac{d}{dz}\left(G(z) - zG'(z)\right) = G'(z) - zG''(z) - G'(z) = -zG''(z).$$

□

2.2.3 Optimal Control Problem

In this section we will translate the original problem of finding the growth trajectory to maximize fruit biomass at the end of the growing season to an optimal control problem. Noting that

$$F(T) = \int_0^T \frac{dF}{dt}\, dt + F(0) = \int_0^T u_{0C}(t)C(S)\, dt \tag{2.16}$$

we can reframe our goal as maximizing the right hand side of (2.16). By imposing the previously discussed constraints that the fractions of each resource, henceforth referred to as the controls, be bounded in $[0, 1]$ and fractions of the same resource sum to unity, along with the differential equations for the biomass of each vegetative organ, we can associate the optimal growth trajectory with the solution to the following optimal control problem.

$$\begin{aligned} &\max_{\vec{u}} \int_0^T u_{0C} C \, dt \\ \text{subject to:} \quad & u_{iC} \geq 0, \ u_{jN} \geq 0 \text{ for } i = 0, 1, 2, \ j = 1, 2 \\ & u_{0C} + u_{1C} + u_{2C} = 1 = u_{1N} + u_{2N} \\ & \frac{dS}{dt} = g(u_{1C} C, \nu_S u_{1N} N), \quad S(0) = S_0 \\ & \frac{dR}{dt} = g(u_{2C} C, \nu_R u_{2N} N), \quad R(0) = R_0 \end{aligned} \tag{2.17}$$

Note that here we only require that the controls be non-negative, because implicit in the combination of those constraints and the equality constraints we recover the requirement that each control must also be less than one.

2.2.4 Necessary Conditions

We will solve the optimal control problem (2.17) using a set of necessary conditions that must be satisfied by the solution. The presence of the two equality constraints makes this problem non-standard, so we take the time to derive the necessary conditions for this type of problem. This derivation is found in Appendix A. Since we can essentially think of the carbon controls and the nitrogen controls separately, the necessary conditions for (2.17) are a combination of the conditions for the two types of problems discussed in Appendix A. We begin by forming a Hamiltonian with two piecewise differentiable adjoints, $\lambda_1(t)$ and $\lambda_2(t)$:

$$H = u_{0C} C + \lambda_1 g(u_{1C} C, \nu_S u_{1N} N) + \lambda_2 g(u_{2C} C, \nu_R u_{2N} N). \tag{2.18}$$

The necessary conditions for optimality are as follows:

$$\begin{cases} u_{iC} = 0 & \text{if} \quad \frac{\partial H}{\partial u_{iC}} < \frac{\partial H}{\partial u_{jC}}, \text{ for all } j \neq i \\ u_{iC} = 1 & \text{if} \quad \frac{\partial H}{\partial u_{iC}} > \frac{\partial H}{\partial u_{jC}}, \text{ for all } j \neq i \\ 0 \leq u_{iC} \leq 1 & \text{if} \quad \frac{\partial H}{\partial u_{iC}} = \frac{\partial H}{\partial u_{jC}}, \text{ for any } j \neq i \\ u_{iN} = 0, u_{jN} = 1 & \text{if} \quad \frac{\partial H}{\partial u_{iN}} < \frac{\partial H}{\partial u_{jN}} \\ 0 \leq u_{1N}, u_{2N} \leq 1 & \text{if} \quad \frac{\partial H}{\partial u_{1N}} = \frac{\partial H}{\partial u_{2N}} \end{cases} \tag{2.19}$$

What this essentially says is that all the carbon is being allocated to the organ such that the partial derivative of the Hamiltonian with respect to that organ's carbon control is larger than the partial derivatives of the Hamiltonian with respect to the other two carbon controls. Likewise for the nitrogen controls. Note also that by [8] the Hamiltonian must be constant along the optimal trajectory because the optimal control problem (2.17) is autonomous (t is not explicit in the integrand or any of the constraints).

As we mentioned in Section 2.2.2, it's possible to rewrite the synthesizing unit function g via a change of variables. In particular, we can rewrite (2.2), (2.3), and (2.4) via (2.9) and the substitutions

$$z_{1C} = \frac{u_{1C}C}{\nu_S u_{1N} N} \tag{2.20}$$

$$z_{2C} = \frac{u_{2C}C}{\nu_R u_{2N} N}. \tag{2.21}$$

This results in the following system

$$\frac{dS}{dt} = \nu_S u_{1N} N G(z_{1C}), \quad S(0) = S_0 \tag{2.22}$$

$$\frac{dR}{dt} = \nu_R u_{2N} N G(z_{2C}), \quad R(0) = R_0 \tag{2.23}$$

$$\frac{dF}{dt} = u_{0C} C, \qquad F(0) = 0 \tag{2.24}$$

where G is given by (2.10). Therefore, the Hamiltonian can be rewritten as

$$H = u_{0C} C + \lambda_1 \nu_S u_{1N} N G(z_{1C}) + \lambda_2 \nu_R u_{2N} N G(z_{2C}). \tag{2.25}$$

Now, as we previously mentioned, we can essentially think of the carbon controls and nitrogen controls separately. The above formulation provides an avenue for restricting the appearance of the carbon controls to the argument of G by letting us write

$$g(\text{Carbon Flux}, \text{Nitrogen Flux}) = (\text{Nitrogen Flux}) \cdot G\left(\frac{\text{Carbon Flux}}{\text{Nitrogen Flux}}\right).$$

When working directly with the nitrogen controls, it will be useful to do a similar change of variables and write

$$g(\text{Carbon Flux}, \text{Nitrogen Flux}) = (\text{Carbon Flux}) \cdot G\left(\frac{\text{Nitrogen Flux}}{\text{Carbon Flux}}\right).$$

With this in mind, we can make the substitutions

$$z_{1N} = \frac{\nu_S u_{1N} N}{u_{1C} C} \tag{2.26}$$

$$z_{2N} = \frac{\nu_R u_{2N} N}{u_{2C} C}, \tag{2.27}$$

and rewrite (2.2), (2.3), and (2.4) as

$$\frac{dS}{dt} = u_{1C}CG(z_{1N}), \quad S(0) = S_0 \tag{2.28}$$
$$\frac{dR}{dt} = u_{2C}CG(z_{2N}), \quad R(0) = R_0 \tag{2.29}$$
$$\frac{dF}{dt} = u_{0C}C, \qquad F(0) = 0. \tag{2.30}$$

As before, the Hamiltonian can be rewritten as

$$H = u_{0C}C + \lambda_1 u_{1C}CG(z_{1N}) + \lambda_2 u_{2C}CG(z_{2N}). \tag{2.31}$$

Using (2.25) and (2.31), we can compute the following partial derivatives:

$$\frac{\partial H}{\partial u_{0C}} = \frac{\partial}{\partial u_{0C}} u_{0C}C = C \tag{2.32}$$
$$\frac{\partial H}{\partial u_{1C}} = \frac{\partial}{\partial u_{1C}} \lambda_1 \nu_S u_{1N} NG(z_{1C}) = \lambda_1 \nu_S u_{1N} NG'(z_{1C}) \frac{\partial}{\partial u_{1C}} z_{1C} = \lambda_1 CG'(z_{1C}) \tag{2.33}$$
$$\frac{\partial H}{\partial u_{2C}} = \frac{\partial}{\partial u_{2C}} \lambda_2 \nu_R u_{2N} NG(z_{2C}) = \lambda_2 \nu_R u_{2N} NG'(z_{2C}) \frac{\partial}{\partial u_{2C}} z_{2C} = \lambda_2 CG'(z_{2C}) \tag{2.34}$$
$$\frac{\partial H}{\partial u_{1N}} = \frac{\partial}{\partial u_{1N}} \lambda_1 u_{1C} CG(z_{1N}) = \lambda_1 u_{1C} CG'(z_{1N}) \frac{\partial}{\partial u_{1N}} z_{1N} = \lambda_1 \nu_S NG'(z_{1N}) \tag{2.35}$$
$$\frac{\partial H}{\partial u_{2N}} = \frac{\partial}{\partial u_{2N}} \lambda_2 u_{2C} CG(z_{2N}) = \lambda_2 u_{2C} CG'(z_{2N}) \frac{\partial}{\partial u_{2N}} z_{2N} = \lambda_2 \nu_R NG'(z_{2N}). \tag{2.36}$$

It is important to recall that $\lim_{(A,B)\to(0,0)} G'(A/B)$ is undefined, so these partial derivatives are not necessarily defined in cases when several controls go to zero simultaneously. However, $G'(z)$ is bounded between 0 and 1, so this bound is preserved in the limit as well.

The last piece of the optimal control framework concerns the adjoints λ_1 and λ_2.

By the derivation of the necessary conditions in Appendix A, we have that

$$\lambda_1' = -\frac{\partial H}{\partial S}, \quad \lambda_1(T) = 0 \tag{2.37}$$

$$\lambda_2' = -\frac{\partial H}{\partial R}, \quad \lambda_2(T) = 0. \tag{2.38}$$

Making use of (2.25) and (2.31) we can express (2.37) and (2.38) as follows.

$$\begin{aligned}
\lambda_1' &= -\frac{\partial H}{\partial S} \\
&= -\frac{\partial}{\partial S}\left[u_{0C}C + \lambda_1\nu_S u_{1N} N G(z_{1C}) + \lambda_2\nu_R u_{2N} N G(z_{2C})\right] \\
&= -\left[u_{0C}C_S + \lambda_1\nu_S u_{1N} N G'(z_{1C})\frac{\partial z_{1C}}{\partial S} + \lambda_2\nu_R u_{2N} N G'(z_{2C})\frac{\partial z_{2C}}{\partial S}\right] \\
&= -\left[u_{0C}C_S + \lambda_1 u_{1C} C_S G'(z_{1C}) + \lambda_2 u_{2C} C_S G'(z_{2C})\right] \\
&= -C_S\left[u_{0C} + \lambda_1 u_{1C} G'(z_{1C}) + \lambda_2 u_{2C} G'(z_{2C})\right]
\end{aligned} \tag{2.39}$$

$$\begin{aligned}
\lambda_2' &= -\frac{\partial H}{\partial R} \\
&= -\frac{\partial}{\partial R}\left[u_{0C}C + \lambda_1 u_{1C} C G(z_{1N}) + \lambda_2 u_{2C} C G(z_{2N})\right] \\
&= -\left[\lambda_1 u_{1C} C G'(z_{1N})\frac{\partial z_{1N}}{\partial R} + \lambda_2 u_{2C} C G'(z_{2N})\frac{\partial z_{2N}}{\partial R}\right] \\
&= -\left[\lambda_1\nu_S u_{1N} N_R G'(z_{1N}) + \lambda_2\nu_R u_{2N} N_R G'(z_{2N})\right] \\
&= -N_R\left[\lambda_1\nu_S u_{1N} G'(z_{1N}) + \lambda_2\nu_R u_{2N} G'(z_{2N})\right].
\end{aligned} \tag{2.40}$$

2.3 Four-Phase Structure

In this section we will discuss the structure of the solution to (2.17). Generally speaking, the solution exhibits a four-phase structure. First, the plant experiences an initial phase of vegetative growth in which it addresses deficiencies in either shoots

or roots. Second, the plant undergoes a phase of balanced growth growth when both the shoots and the roots are growing. Third, there is a phase of mixed vegetative/reproductive growth, during which the shoots continue to grow and the fruits begin growing simultaneously. Lastly, the plant completes the growing season with a period of reproductive growth, during which only the fruits are growing.

We will begin with the analysis of the four-phase structure of the optimal solution. Because we know that the adjoints, λ_1 and λ_2, vanish at the end of the growing season, we use that as the starting point for our analysis and proceed in reverse.

2.3.1 Final Interval - Reproductive Growth

We begin by showing that there exists a switching time after which all carbon is allocated to fruit production. We will refer to the interval between the switching time and T the final interval. Note that by (2.37) and (2.38), we have that $\lambda_1(T) = 0 = \lambda_2(T)$. Furthermore, recall that G' is bounded. Therefore, using the shorthand $C(S(T)) = C^*$, we have by (2.32), (2.33), and (2.34) that

$$\frac{\partial H}{\partial u_{0C}}(T) = C^* > 0 \tag{2.41}$$

$$\frac{\partial H}{\partial u_{1C}}(T) = 0 \tag{2.42}$$

$$\frac{\partial H}{\partial u_{2C}}(T) = 0. \tag{2.43}$$

This implies that

$$\frac{\partial H}{\partial u_{0C}}(t) > \frac{\partial H}{\partial u_{1C}}(T) \quad \text{and} \quad \frac{\partial H}{\partial u_{0C}}(t) > \frac{\partial H}{\partial u_{2C}}(T) \tag{2.44}$$

at $t = T$. Therefore, by (2.19), we have that $u_{0C}(T) = 1$ and $u_{1C}(T) = 0 = u_{2C}(T)$. That is, at the end of the growing season the plant is allocating all of the available

carbon to the fruits, which in turn means that only the fruits are growing at this time.

Now, because both λ_1 and λ_2 are continuous, $\lambda_1(T) = 0 = \lambda_2(T)$, and G' is bounded, there must be some $\varepsilon > 0$ such that for all t in $[T - \varepsilon, T]$ we have that (2.44) still holds. So, we have the existence of a (potentially small) interval of fruit-only growth. Now, writing $C_S(S(T)) = C_S^*$, during this interval we have by (2.39) that

$$\lambda_1' = -C_S^* \tag{2.45}$$

because here we have $u_{0C} = 1, u_{1C} = 0 = u_{2C}$ and G' is bounded. Because $\lambda_1(T) = 0$ this implies that during this interval

$$\lambda_1(t) = C_S^* \cdot (T - t). \tag{2.46}$$

Additionally, because during this interval we have a scenario where for $i = 1, 2$ either only u_{iC} is zero or both u_{iC} and u_{iN} are zero, we have by (2.39) that

$$\lambda_2' = 0. \tag{2.47}$$

This is because either u_{iC} alone is zero, in which case $z_{iN} \to \infty$ and by (2.12) we have that $G'(z_{iN}) \to 0$, or in the case that both u_{iC} and u_{iN} are zero then $u_{iN}G'(z_{iN}) = 0$ because G' is bounded. In either case, we get (2.47). So, because $\lambda_2(T) = 0$, we have that the following holds throughout the interval:

$$\lambda_2(t) = 0. \tag{2.48}$$

Therefore, because G' is bounded, we have by (2.36) that

$$\frac{\partial H}{\partial u_{2N}} = 0.$$

Since (2.35) is non-negative in this interval, it must be the case that

$$\frac{\partial H}{\partial u_{2N}} \leq \frac{\partial H}{\partial u_{1N}}. \tag{2.49}$$

If this inequality were strict, then by (2.19) we would have $u_{1N} = 1$. In this case, because $u_{1C} = 0$, we have that $z_{1N} \to \infty$, and so by (2.12) we have that $G'(z_{1N}) \to 0$. This, however, implies that

$$\frac{\partial H}{\partial u_{1N}} = 0 = \frac{\partial H}{\partial u_{2N}},$$

and so the inequality (2.49) cannot be strict, and both partial derivatives are zero in this interval. In particular, $G'(z_{1N}) = 0$, which means that $z_{1N} \to \infty$ and so $z_{1C} = 0$, which by (2.12) implies that $G'(z_{1C}) = 1$. In summary then, during this interval we have

$$\frac{\partial H}{\partial u_{0C}} = C^*, \quad \frac{\partial H}{\partial u_{1C}} = \lambda_1 C^*, \quad \frac{\partial H}{\partial u_{2C}} = \frac{\partial H}{\partial u_{1N}} = \frac{\partial H}{\partial u_{2N}} = 0. \tag{2.50}$$

Now, since λ_1 is decreasing by (2.45), we see from (2.50) that the partial derivatives will maintain the same ordering as long as $\lambda_1 < 1$. We define the switching point t^* to be the time when the plant switches to fruit-only growth. We can identify the switching point as the time when $\lambda_1(t) = 1$, and we can use (2.46) to define t^* implicitly by the equation

$$C_S^* \cdot (T - t^*) = 1. \tag{2.51}$$

By solving for t^* in (2.51) we can obtain

$$t^* = T - \frac{1}{C_S^*}. \tag{2.52}$$

In doing so, note that we have extended this period of fruit-only growth from the interval $[T-\varepsilon, T]$ to $[t^*, T]$. Note also that during the final interval we have by (2.18) that

$$H = C,$$

and because the Hamiltonian must be constant along the optimal trajectory, we have that

$$H = C^* \tag{2.53}$$

throughout the growing season.

2.3.2 Penultimate Interval - Mixed Vegetative/Reproductive Growth

We will now continue backwards to show that, prior to the final interval of fruit-only growth, we have a period of mixed vegetative/reproductive growth, during which both the shoots and fruits grow simultaneously. This interval will be referred to as the penultimate interval. Recall that at the beginning of the final interval we have that $\lambda_1 = 1$, and so (2.50) becomes

$$\frac{\partial H}{\partial u_{0C}} = \frac{\partial H}{\partial u_{1C}} = C, \quad \frac{\partial H}{\partial u_{2C}} = \frac{\partial H}{\partial u_{1N}} = \frac{\partial H}{\partial u_{2N}} = 0. \tag{2.54}$$

In order to apply (2.19) we need to determine the ordering of the partial derivatives of H in an interval immediately prior to t^*. We begin with the following inequality.

Lemma 2.3. $\frac{\partial H}{\partial u_{2C}} < \frac{\partial H}{\partial u_{0C}}$ *in an open interval immediately prior to* t^*.

Proof. Because $\lambda_2(t^*) = 0$, λ_2 is continuous, and G' is bounded, there exists $\varepsilon > 0$ such that for all t in $(t^* - \varepsilon, t^*)$ we have

$$\lambda_2 G'(z_{2C})C < C,$$

that is

$$\frac{\partial H}{\partial u_{2C}} < \frac{\partial H}{\partial u_{0C}}$$

as desired. □

Note that, regardless of where $\frac{\partial H}{\partial u_{1C}}$ falls in the order of partial derivatives, the strict inequality in Lemma 2.3 means that there is no root growth during this period of time immediately prior to t^*. Next, we will show that during this penultimate interval we have $\frac{\partial H}{\partial u_{1C}} = \frac{\partial H}{\partial u_{0C}}$. We will prove this in two steps, the first of which indicates the allocation to shoots at this point in the growing season is at least as important to overall fitness as allocation to fruits.

Lemma 2.4. $\frac{\partial H}{\partial u_{1C}} \geq \frac{\partial H}{\partial u_{0C}}$ *in an open interval immediately prior to* t^*.

Proof. Suppose to the contrary that for some $\varepsilon > 0$ we have for all t in $(t^* - \varepsilon, t^*)$ that

$$\frac{\partial H}{\partial u_{1C}} < \frac{\partial H}{\partial u_{0C}}.$$

By Lemma 2.3 we have that

$$\frac{\partial H}{\partial u_{2C}} < \frac{\partial H}{\partial u_{0C}}$$

as well, and so by (2.19) this means that $u_{0C} = 1$ and $u_{1C} = 0 = u_{2C}$. As in the final interval, this yields

$$\frac{\partial H}{\partial u_{0C}} = C, \quad \frac{\partial H}{\partial u_{1C}} = \lambda_1 C.$$

However, as $\lambda_1(t^*) = 1$ and by (2.39) we have that $\lambda_1' < 0$, it must be the case that $\lambda_1 > 1$. This then means that

$$\frac{\partial H}{\partial u_{1C}} > \frac{\partial H}{\partial u_{0C}},$$

a contradiction. Therefore we have shown that during this interval we have

$$\frac{\partial H}{\partial u_{1C}} \geq \frac{\partial H}{\partial u_{0C}}.$$

□

Lemma 2.5. $\frac{\partial H}{\partial u_{1C}} = \frac{\partial H}{\partial u_{0C}}$ *in an open interval immediately prior to* t^*.

Proof. First, note that by Lemma 2.4 we have that

$$\frac{\partial H}{\partial u_{1C}} \geq \frac{\partial H}{\partial u_{0C}}$$

in an open interval immediately prior to t^*. Suppose now that the inequality is strict, and that in fact we have

$$\frac{\partial H}{\partial u_{1C}} > \frac{\partial H}{\partial u_{0C}}.$$

Now, combining this supposition with Lemma 2.3 gives us

$$\frac{\partial H}{\partial u_{1C}} > \frac{\partial H}{\partial u_{0C}} > \frac{\partial H}{\partial u_{2C}},$$

which by (2.19) means that

$$u_{1C} = 1, \quad u_{0C} = 0 = u_{2C}.$$

This then gives us

$$z_{1N} = \frac{\nu_S u_{1N} N}{C},$$

which, for fixed C and N, is bounded between 0 and $\frac{\nu_S N}{C}$. Therefore, we have that $G'(z_{1N}) \neq 0$. Furthermore, recall that we know that $\lambda_2(t^*) = 0$ and G' is bounded. This means that there is a (potentially smaller) open interval immediately prior to t^* in which we have

$$\lambda_1 \nu_S N G'(z_{1N}) > \lambda_2 \nu_R N G'(z_{2N}),$$

or in terms of the Hamiltonian,

$$\frac{\partial H}{\partial u_{1N}} > \frac{\partial H}{\partial u_{2N}}.$$

By (2.19), this means that $u_{1N} = 1$ and $u_{2N} = 0$ during this interval. With this configuration of the controls, we have that $N = N^*$ is constant because $R' = 0$ and this interval reaches this final interval. So, we have that

$$z_{1C} = \frac{C}{\nu_S N^*} > 0.$$

This then means that $G'(z_{1C}) < 1$, and so as $\lambda_1(t^*) = 1$ and both z_{1C} and λ_1 are continuous here there exists $\varepsilon > 0$ such that for all t in $(t^* - \varepsilon, t^*)$ we have

$$\lambda_1 G'(z_{1C}) < 1$$

that is

$$\frac{\partial H}{\partial u_{1C}} < \frac{\partial H}{\partial u_{0C}},$$

a contradiction. Therefore, we have

$$\frac{\partial H}{\partial u_{1C}} = \frac{\partial H}{\partial u_{0C}}$$

in an open interval immediately prior to t^*, as desired. □

Putting together Lemmas 2.3 and 2.5, we get

$$\frac{\partial H}{\partial u_{2C}} < \frac{\partial H}{\partial u_{1C}} = \frac{\partial H}{\partial u_{0C}}. \tag{2.55}$$

By (2.19), this means that $u_{2C} = 0$ and by (2.32) and (2.33) we obtain

$$\lambda_1 G'(z_{1C}) = 1 \tag{2.56}$$

during this interval. Lastly, we will verify that here $u_{1N} = 1$ and $u_{1C} > 0$, which in turn will be used to show that this interval is indeed marked by simultaneous fruit and shoot growth.

Lemma 2.6. *$u_{1N} = 1$ and $u_{1C} > 0$ in an open interval immediately prior to t^*.*

Proof. We will first show that u_{1N} is non-zero. Recall that by (2.55) and (2.19), we have that $u_{2C} = 0$ in an open interval immediately prior to t^*. Suppose furthermore that $u_{1N} = 0$ during this interval as well. By (2.2) and (2.3), this means that both $S' = 0$ and $R' = 0$. As this interval is followed by a period of fruit-only growth, in order to maximize the integral in (2.17) it must be the case that $u_{0C} = 1$. We therefore have

$$u_{0C} = 1, \quad u_{1C} = 0 = u_{2C}, \quad u_{1N} = 0, \quad u_{2N} = 1.$$

In particular, because here we are again in a case where $u_{0C} = 1$, $u_{1C} = 0 = u_{2C}$, and $\lambda_2(t^*) = 0$ we again arrive at a situation where

$$\frac{\partial H}{\partial u_{0C}} = C, \quad \frac{\partial H}{\partial u_{1C}} = \lambda_1 C$$

following the same argument as in the justification for the existence of a final interval of fruit-only growth. However, as we know that here $\lambda_1 > 1$ we have that

$$\frac{\partial H}{\partial u_{0C}} < \frac{\partial H}{\partial u_{1C}},$$

which regardless of the relationship between $\frac{\partial H}{\partial u_{2C}}$ with the other two partial derivatives contradicts the fact that $u_{0C} = 1$. Hence, $u_{1N} > 0$.

Now, because $\lambda_1 > 1$ we know by (2.56) that $z_{1C} > 0$ so that $G'(z_{1C}) < 1$. Because $u_{1N} > 0$, this allows us to conclude that $u_{1C} > 0$ as well without any concern regarding ambiguous limiting behavior with $G'(z_{1C})$ when one or both controls are zero. This in turn implies that

$$\frac{\partial H}{\partial u_{1N}} = \lambda_1 \nu_S N G'(z_{1N}) > 0.$$

Now, if $u_{1N} < 1$ we would also have $u_{2N} > 0$. Because $u_{2C} = 0$, this would imply that $z_{2N} \to \infty$, and so by (2.12) we would have $G'(z_{2N}) = 0$. This would then imply that

$$\frac{\partial H}{\partial u_{1N}} > \frac{\partial H}{\partial u_{2N}},$$

which by (2.19) means that $u_{1N} = 1$, a contradiction. Therefore, it must have been the case that $u_{1N} = 1$ in addition to $u_{1C} > 0$ in an open interval immediately prior to t^*, as desired. □

At this point, note that we have only showed that

$$u_{1C} > 0, \quad u_{2C} = 0, \quad u_{1N} = 1, \quad u_{2N} = 0.$$

Lastly then, as $\lambda_1 > 1$ for this open interval before t^*, $\lambda_1(t^*) = 1$, and λ_1 is continuous, we have that

$$1 = \lim_{t \to t^{*-}} \lambda_1 = \lim_{t \to t^{*-}} \frac{1}{G'(z_{1C})} \implies \lim_{t \to t^{*-}} z_{1C} = 0. \tag{2.57}$$

As we have already verified that $u_{1N} = 1$ this means that

$$\lim_{t \to t^{*-}} u_{1C} = 0, \tag{2.58}$$

and so u_{1C} is continuous at t^*. This also means that $0 < u_{1C} < 1$ during an open interval immediately prior to t^*, so here $u_{0C} > 0$. Therefore, we have established the existence of a penultimate interval, during which

$$u_{0C}, u_{1C} > 0, \quad u_{2C} = 0, \quad u_{1N} = 1, \quad u_{2N} = 0,$$

that is fruits and shoots alone grow simultaneously.

2.3.3 Balanced Growth - Mixed Vegetative Growth

At this point, we have established the existence of a final interval during which only the fruits are growing, and a period before this when both the shoots and fruits are growing together. Continuing backwards, we have arrived at a juncture where there are seemingly several potential options. Due to the fact that z_{2C} is not defined during the penultimate interval it is not possible to determine directly from the necessary conditions (2.19) what exactly constitutes the stage immediately prior. We will,

however, rule out several possibilities and make an educated guess based on Iwasa and Roughgarden's work in [5].

Prior to this period of mixed shoot-fruit growth, there are several possibilities. These are: fruit growth, root-fruit growth, root-shoot growth, root-shoot-fruit growth, shoot growth, or root growth. Because we assume that fruits are only carbon-dependent, we can eliminate the possibilities that this phase consists of fruit-only growth, because such a phase can only be optimal if the shoots are done growing. We can also rule out a phase of root-fruit growth because the only reason to grow roots in this model is to increase shoot biomass and subsequent capacity for carbon fixation. If the roots are not fully utilizing the available carbon, it would be better for the shoots to utilize the excess than the fruits. We will show in Appendix B.1 that mixed root/shoot/fruit growth is also impossible here.

This then leaves the possibility that this phase consists of shoot-only or root-only growth, or a combination of both. Now, we rule out root-only growth here because this would result in a case where the plant is increasing the capacity for shoot production without actively growing shoots. We also rule out shoot-only growth here because if no further root growth is going to occur the shoots will be nitrogen-limited and it would be better for the excess carbon to be directed to the fruits. Additionally, Iwasa and Roughgarden [5] found in the single-resource case that 'balanced growth,' a phase of mixed root-shoot growth, was optimal, so we will assume that is the case here as well.

2.3.4 Initial Phase - Shoot or Root Growth

It is not optimal to grow fruits in any capacity prior to the balanced growth phase for arguments similar to those outlined above. Therefore, the initial phase of growth consists of either shoot growth or root growth. Here the plant addresses deficiencies

present in either vegetative organ depending on the initial conditions. Note that this is also the first stage that we see in the single-resource case described in [5].

2.4 Phase Dynamics and Transitions

In this section, we will present the basic equations governing the dynamics in each phase of the solution to (2.19) as well as show that z_{1C} is continuous between any two consecutive phases with shoot growth, and z_{2C} is continuous between any two consecutive phases with root growth. Note that this last statement only applies to the case where the initial phase consists of root growth. To keep this section from being too cluttered or disjoint, we will go through the dynamics for each phase in chronological order and then go on to discuss the transitions between phases. We will also limit this section to include only phase-specific versions of the differential equations for states and adjoints, relevant algebraic constraints due to the necessary conditions, and any additional equations used for understanding the transitions between phases. When we discuss the numerical scheme, we will make use of several additional equations which govern dynamics in various stages but are not relevant here. In what follows, we will also make frequent use of the notation $G_2(x) = G'(1/x)$.

2.4.1 Initial Phase: Shoot-Only Growth

During shoot-only growth we have

$$u_{0C} = 0, \quad u_{1C} = 1, \quad u_{2C} = 0, \quad u_{1N} = 1, \quad u_{2N} = 0 \tag{2.59}$$

and only S, λ_1, and λ_2 are changing. The differential equations in time for these three during this stage are given by

$$S' = \nu_S N G\left(z_{1C}\right) \tag{2.60}$$

$$\lambda_1' = -\lambda_1 C_S G'\left(z_{1C}\right) \tag{2.61}$$

$$\lambda_2' = -\lambda_1 N_R \nu_S G_2\left(z_{1C}\right) \tag{2.62}$$

where here

$$z_{1C} = \frac{C}{\nu_S N}. \tag{2.63}$$

Note also that in this stage both N and N_R are constant because R is constant. Furthermore, because we know by (2.53) that $H = C^*$, we can rewrite the Hamiltonian (2.25) during this phase as

$$C^* = \lambda_1 \nu_S N G(z_{1C}) \tag{2.64}$$

and so by solving (2.64) for λ_1 and then making use of (2.14) we have two additional expressions for λ_1 during this phase:

$$\lambda_1 = \frac{C^*}{\nu_S N G(z_{1C})} \tag{2.65}$$

$$\lambda_1 = \frac{C^*}{\nu_S N G_2(z_{1C}) + C G'(z_{1C})}. \tag{2.66}$$

2.4.2 Initial Phase: Root-Only Growth

During root-only growth we have

$$u_{0C} = 0, \quad u_{1C} = 0, \quad u_{2C} = 1, \quad u_{1N} = 0, \quad u_{2N} = 1 \tag{2.67}$$

and only R, λ_1, and λ_2 are changing. The differential equations in time for these three during this stage are given by

$$R' = \nu_R N G(z_{2C}) \tag{2.68}$$

$$\lambda_1' = -\lambda_2 C_S G'(z_{2C}) \tag{2.69}$$

$$\lambda_2' = -\lambda_2 N_R \nu_R G_2\left(z_{2C}\right) \tag{2.70}$$

where here

$$z_{2C} = \frac{C}{\nu_R N}. \tag{2.71}$$

Note also that in this stage both C and C_S are constant because S is constant. Furthermore, because we know by (2.53) that $H = C^*$, we can rewrite the Hamiltonian (2.25) during this phase as

$$C^* = \lambda_2 \nu_R N G(z_{2C}) \tag{2.72}$$

and so by solving (2.72) for λ_2 and then making use of (2.14) we have two additional expressions for λ_2 during this phase:

$$\lambda_2 = \frac{C^*}{\nu_R N G(z_{2C})} \tag{2.73}$$

$$\lambda_2 = \frac{C^*}{\nu_R N G_2(z_{2C}) + C G'(z_{2C})}. \tag{2.74}$$

2.4.3 Balanced Growth - Shoot/Root Growth

During balanced growth we have

$$u_{0C} = 0, \quad 0 \leq u_{1C} \leq 1, \quad u_{2C} = 1 - u_{1C}, \quad 0 \leq u_{1N} \leq 1, \quad u_{2N} = 1 - u_{1N} \tag{2.75}$$

and so by (2.19) we have that

$$\lambda_1 G'(z_{1C}) = \lambda_2 G'(z_{2C}) > 1 \tag{2.76}$$

and

$$\lambda_1 \nu_S G_2(z_{1C}) = \lambda_2 \nu_R G_2(z_{2C}). \tag{2.77}$$

During this stage S, R, λ_1, and λ_2 are changing. The differential equations in time for these four during this phase are given by

$$S' = \nu_S u_{1N} N G(z_{1C}) \tag{2.78}$$

$$R' = \nu_R (1 - u_{1N}) N G(z_{2C}) \tag{2.79}$$

$$\lambda_1' = -C_S \lambda_1 G'(z_{1C}) = -C_S \lambda_2 G'(z_{2C}) \tag{2.80}$$

$$\lambda_2' = -N_R \lambda_1 \nu_S G_2(z_{1C}) = -N_R \lambda_2 \nu_R G_2(z_{2C}). \tag{2.81}$$

Furthermore, again taking advantage of the fact that (2.53) gives us $H = C^*$, we can rewrite the Hamiltonian (2.25) during this phase as

$$C^* = \lambda_1 \nu_S u_{1N} N G(z_{1C}) + \lambda_2 \nu_R u_{2N} N G(z_{2C}). \tag{2.82}$$

Rewriting this using (2.14) we get

$$C^* = \lambda_1 \nu_S u_{1N} N \left[G_2(z_{1C}) + z_{1C} G'(z_{1C}) \right] + \lambda_2 \nu_R u_{2N} N \left[G_2(z_{2C}) + z_{2C} G'(z_{2C}) \right].$$

Simplifying, and using (2.76) and (2.77) to write everything in terms of z_{1C}, gives us

$$\begin{aligned}
C^* &= \lambda_1 \nu_S u_{1N} N \left[G_2(z_{1C}) + \frac{u_{1C} C}{\nu_S u_{1N} N} G'(z_{1C}) \right] \\
&\quad + \lambda_2 \nu_R u_{2N} N \left[G_2(z_{2C}) + \frac{u_{2C} C}{\nu_R u_{2N} N} G'(z_{2C}) \right] \\
&= \lambda_1 \nu_S u_{1N} N \left[G_2(z_{1C}) + \frac{u_{1C} C}{\nu_S u_{1N} N} G'(z_{1C}) \right] \\
&\quad + \lambda_1 \nu_S u_{2N} N \left[G_2(z_{1C}) + \frac{u_{2C} C}{\nu_S u_{2N} N} G'(z_{1C}) \right] \\
&= \lambda_1 \left[\nu_S N G_2(z_{1C})(u_{1N} + u_{2N}) + C G'(z_{1C})(u_{1C} + u_{2C}) \right] \\
&= \lambda_1 \left[\nu_S N G_2(z_{1C}) + C G'(z_{1C}) \right]. \qquad (2.83)
\end{aligned}$$

Solving for λ_1 gives us

$$\lambda_1 = \frac{C^*}{\nu_S N G_2(z_{1C}) + C G'(z_{1C})}. \qquad (2.84)$$

Using (2.76) and (2.77) to rewrite (2.83) in terms of z_{2C} instead leads to

$$C^* = \lambda_2 \left[\nu_R N G_2(z_{2C}) + C G'(z_{2C}) \right], \qquad (2.85)$$

which upon solving for λ_2 gives us

$$\lambda_2 = \frac{C^*}{\nu_R N G_2(z_{2C}) + C G'(z_{2C})}. \qquad (2.86)$$

Note that (2.84) and (2.86) are consistent with (2.66) during shoot-only growth, and (2.74) during root-only growth. While we do not assume that either z_{1C} or z_{2C} is continuous between phases (because the controls need not be), we will use this consistency later to show continuity between the initial and balanced growth phases.

2.4.4 Penultimate Interval - Shoot/Fruit Growth

During the penultimate interval, we have

$$0 \leq u_{0C} \leq 1, \quad u_{1C} = 1 - u_{0C}, \quad u_{2C} = 0, \quad u_{1N} = 1, \quad u_{2N} = 0 \tag{2.87}$$

and so by (2.19) we have that

$$\lambda_1 G'(z_{1C}) = 1. \qquad \text{((2.56) revisited)}$$

Note that here we have

$$z_{1C} = \frac{u_{1C} C}{\nu_S N}. \tag{2.88}$$

During this interval S, F, λ_1, and λ_2 are all changing. The differential equations in time for these four during this stage are

$$S' = \nu_S N G(z_{1C}) \tag{2.89}$$

$$F' = u_{0C} C \tag{2.90}$$

$$\lambda_1' = -C_S \tag{2.91}$$

$$\lambda_2' = -N_R \lambda_1 \nu_S G_2(z_{1C}) \tag{2.92}$$

Furthermore, again taking advantage of the fact that (2.53) gives us $H = C^*$, we can rewrite the Hamiltonian (2.25) during this phase as

$$C^* = u_{0C} C + \lambda_1 \nu_S N G(z_{1C}). \tag{2.93}$$

Using (2.14) we get

$$\begin{aligned} C^* &= u_{0C}C + \lambda_1\nu_S N^* \left[G_2(z_{1C}) + z_{1C}G'(z_{1C})\right] \\ &= u_{0C}C + \lambda_1\nu_S N \left[G_2(z_{1C}) + \frac{u_{1C}C}{\nu_S N}G'(z_{1C})\right] \\ &= u_{0C}C + \lambda_1\nu_S N G_2(z_{1C}) + \lambda_1 G'(z_{1C})u_{1C}C \end{aligned}$$

which by (2.56) becomes

$$\begin{aligned} C^* &= u_{0C}C + u_{1C}C + \lambda_1\nu_S N G_2(z_{1C}) \\ &= C + \lambda_1\nu_S N G_2(z_{1C}). \end{aligned} \tag{2.94}$$

Now, at this point we can either solve for λ_1 and obtain

$$\lambda_1 = \frac{C^* - C}{\nu_S N G_2(z_{1C})}, \tag{2.95}$$

or we can use (2.56) to rewrite (2.94) as

$$C^* = C\lambda_1 G'(z_{1C}) + \lambda_1\nu_S N G_2(z_{1C})$$

and solve to obtain the familiar expression

$$\lambda_1 = \frac{C^*}{\nu_S N G_2(z_{1C}) + CG'(z_{1C})}. \tag{2.96}$$

As we commented earlier, the reappearance of (2.96) suggests that z_{1C} may be continuous between the balanced growth phase. We will prove this later with the help of (2.95).

Another important result in the penultimate interval which we prove here is that

z_{1C} is monotonically decreasing throughout this phase. This will be especially important because, as we will see in Section 2.5.3, it will be advantageous for us to think of z_{1C}, rather than t, as the integration variable for the numerical scheme during the penultimate interval. In particular, we will see that numerically solving the differential equations for this phase in t requires integrating a singularity, whereas solving the differential equations in z_{1C} does not.

Lemma 2.7. *z_{1C} is monotonically decreasing during the penultimate interval.*

Proof. Recall that during the penultimate interval we have the relationship

$$\lambda_1 G'(z_{1C}) = 1.$$

Differentiating both sides with respect to t yields

$$\lambda_1' G'(z_{1C}) + \lambda_1 G''(z_{1C}) \frac{dz_{1C}}{dt} = 0.$$

Using the fact that $\lambda_1' = -C_S$, we have

$$\frac{dz_{1C}}{dt} = \frac{-\lambda_1' G'(z_{1C})}{\lambda_1 G''(z_{1C})} = \frac{C_S \left[G'(z_{1C})\right]^2}{G''(z_{1C})} = -\frac{C_S \left(1 + 2z_{1C}\right)^2}{6z_{1C}\left(1 + z_{1C}\right)\left(1 + z_{1C} + z_{1C}^2\right)} < 0. \quad (2.97)$$

Therefore, z_{1C} is monotonically decreasing throughout the penultimate interval. □

2.4.5 Final Interval - Fruit-Only Growth

During the final interval of fruit-only growth we have

$$u_{0C} = 1, \quad u_{1C} = 0, \quad u_{2C} = 0, \quad u_{1N} = 1, \quad u_{2N} = 0 \quad (2.98)$$

and only F and λ_1 are changing. The differential equations in time for these two during this phase are

$$F' = C^* \tag{2.99}$$

$$\lambda_1' = -C_S^*, \tag{2.100}$$

Because $\lambda_1(T) = 0$ we also have

$$\lambda_1(t) = C_S^* \cdot (T - t). \qquad \text{((2.46) revisited)}$$

Furthermore, recall that because $\lambda_2(T) = 0$ and $\lambda_2' = 0$ during this phase we have that $\lambda_2 = 0$ throughout this interval. Now that we have established the main dynamics in each phase, we will turn our attention to the transitions between phases. Again, we will proceed in chronological order from the beginning.

2.4.6 Initial Phase to Balanced Growth Transition

We can use the fact that the states and adjoints are continuous at the boundary between the initial phase and the balanced growth phase to characterize this transition. In particular, we will show that z_{1C} is continuous at this transition when the first stage consists of shoot-only growth and z_{2C} is continuous at this transition when the first stage consists of root-only growth. In the first case this will mean that the ratio of the shoot carbon flux to shoot nitrogen flux is continuous across this boundary, and in the second case the ratio of root carbon flux to root nitrogen flux is continuous.

Due to the symmetry between equations (2.84) and (2.66) and equations (2.86) and (2.74) we will streamline the following argument by considering a 'generalized'

versions of these equations:

$$\lambda = \frac{C^*}{\nu N G_2(z) + C G'(z)}, \tag{2.101}$$

where here (λ, ν, z) is either $(\lambda_1, \nu_S, z_{1C})$ or $(\lambda_2, \nu_R, z_{2C})$ depending on whether the initial stage is shoot-only growth or root-only growth, respectively. We will call this transition point $t = \widetilde{t}$, let $x = \lim_{t \to \widetilde{t}^+} z$, and note that $\lim_{t \to \widetilde{t}^-} z = \frac{C}{\nu N}$. Now, taking limits of (2.101) from both sides, we obtain

$$\lim_{t \to \widetilde{t}^+} \lambda(t) = \widetilde{\lambda}^+ = \frac{C^*}{\nu N G_2(x) + C G'(x)} \tag{2.102}$$

$$\lim_{t \to \widetilde{t}^-} \lambda(t) = \widetilde{\lambda}^- = \frac{C^*}{\nu N G_2\left(\frac{C}{\nu N}\right) + C G'\left(\frac{C}{\nu N}\right)} \tag{2.103}$$

Note that while we don't know the particular values of C and N at $t = \widetilde{t}$, in what follows it will only matter that they are the same in both (2.102) and (2.103). At $t = \widetilde{t}$ we have that $\widetilde{\lambda}^- = \widetilde{\lambda}^+$ and so

$$G_2\left(\frac{C}{\nu N}\right) + \frac{C}{\nu N} G'\left(\frac{C}{\nu N}\right) = G_2(x) + \frac{C}{\nu N} G'(x). \tag{2.104}$$

Let

$$f(x) := G_2(x) + \frac{C}{\nu N} G'(x). \tag{2.105}$$

It is clear from (2.104) that $x = \frac{C}{\nu N}$ is a solution to

$$f(x) = G_2\left(\frac{C}{\nu N}\right) + \frac{C}{\nu N} G'\left(\frac{C}{\nu N}\right). \tag{2.106}$$

Here we will show that this solution is unique. To this end we will consider $f'(x)$:

$$\begin{aligned} f'(x) &= G_2'(x) + \frac{C}{\nu N} G''(x) \\ &= -xG''(x) + \frac{C}{\nu N} G''(x) \qquad \text{by (2.15)} \\ &= G''(x)\left(\frac{C}{\nu N} - x\right). \end{aligned}$$

Note that because $G''(x) \leq 0$ for $x \geq 0$, we have that

$$\operatorname{sgn}\left(f'(x)\right) = \operatorname{sgn}\left(x - \frac{C}{\nu N}\right) \text{ for } x \geq 0. \tag{2.107}$$

Now, as f is continuous and, by (2.107), we have that f is decreasing on $\left(0, \frac{C}{\nu N}\right)$ and increasing on $\left(\frac{C}{\nu N}, \infty\right)$, it must be the case that the only solution to (2.106) is at $x = \frac{C}{\nu N}$. Therefore, we have that

$$\lim_{t \to \tilde{t}^+} z = \lim_{t \to \tilde{t}^-} z,$$

which verifies our claim that z_{1C} is continuous at the transition from shoot-only growth to balanced growth and z_{2C} is continuous at the transition from root-only growth to balanced growth.

2.4.7 Balanced Growth to Penultimate Interval Transition

As in Section 2.4.6, we can use the fact that the states and adjoints are continuous, along with the various formulations for λ_1, to show that z_{1C} is continuous at the boundary between the balanced growth and penultimate intervals. To simplify what

follows, we call this transition point $t = \hat{t}$ and let

$$x = \lim_{t \to \hat{t}^-} z_{1C}, \quad y = \lim_{t \to \hat{t}^+} z_{1C}.$$

We wish to show that $x = y$. Because λ_1 must be continuous at $\hat{t}$, we have in particular that the following are true:

$$\frac{C^* - C}{\nu_S N G_2(y)} = \frac{C^*}{\nu_S N G_2(x) + C G'(x)} \tag{2.108}$$

$$\frac{1}{G'(y)} = \frac{C^*}{\nu_S N G_2(x) + C G'(x)}. \tag{2.109}$$

Note that these come from equations (2.84), (2.95), and (2.56). It will be useful to rewrite these equations as

$$C^* \nu_S N G_2(y) = (C^* - C)(\nu_S N G_2(x) + C G'(x)) \tag{2.110}$$

$$C^* G'(y) = \nu_S N G_2(x) + C G'(x). \tag{2.111}$$

It is important to note that because both G_2 and G' are one-to-one, we have that y is a function of x in both (2.110) and (2.111). It will also be useful to equate the left-hand sides of (2.108) and (2.109) to obtain:

$$\frac{\nu_S N G_2(y)}{C^* - C} = G'(y). \tag{2.112}$$

We will break the argument up into a sequence of lemmas.

Lemma 2.8. *For any given values of C^*, C, ν_S, and N, equation* (2.112) *has only one positive real solution.*

Proof. Setting $k = \frac{\nu_S N}{C^* - C}$, equation (2.112) gives us

$$\begin{aligned} kG_2(y) &= G'(y) \\ k\frac{y^3(2+y)}{(1+y+y^2)^2} &= \frac{1+2y}{(1+y+y^2)^2} \\ ky^3 &= \frac{1+2y}{2+y}. \end{aligned}$$

Note that since $C^* - C \geq 0$ we have that $k > 0$. For any positive choice of k it is a simple matter to see that the graphs of

$$x = ky^3 \text{ and } x = \frac{1+2y}{2+y}$$

have only one positive intersection point. □

Note that Lemma 2.8 tells us that $G'(y)/G_2(y)$ is invertible, and in particular that there is only one admissible value of y (depending on C) which solves equations (2.110) and (2.111). Next we will show that $x = y$ is a solution to equations (2.110) and (2.111).

Lemma 2.9. *$x = y$ is a solution to equations* (2.110) *and* (2.111)*.*

Proof. We will first show that $x = y$ is a solution to (2.110). Evaluating (2.110) at $x = y$:

$$
\begin{aligned}
C^*\nu_S N G_2(y) &= (C^* - C)(\nu_S N G_2(y) + CG'(y)) \\
C^*\nu_S N G_2(y) &= C^*\nu_S N G_2(y) + C^* CG'(y) - C\nu_S N G_2(y) - C^2 G'(y) \\
0 &= C^* CG'(y) - C\nu_S N G_2(y) - C^2 G'(y) \\
\nu_S N G_2(y) &= (C^* - C)G'(y) \\
\frac{\nu_S N G_2(y)}{C^* - C} &= G'(y)
\end{aligned}
$$

This last equality is true by (2.112). We will next show that $x = y$ is a solution to (2.111). As before, evaluating the equation at $x = y$:

$$
\begin{aligned}
C^* G'(y) &= \nu_S N G_2(y) + CG'(y) \\
(C^* - C)G'(y) &= \nu_S N G_2(y) \\
G'(y) &= \frac{\nu_S N G_2(y)}{C^* - C}.
\end{aligned}
$$

Again, this last equality is (2.112). Therefore $x = y$ is a solution to equations (2.110) and (2.111). □

We will eventually show that $x = y$ is in fact the only solution to (2.110) and (2.111), which by Lemma 2.8 is a single point depending on C. In doing so we will use $\frac{dy}{dx}$ for both equations (2.110) and (2.111) as well the PCSU identity (2.15). We proceed with finding $\frac{dy}{dx}$ for (2.110), and so as to avoid confusion we will refer y in this equation as y_A:

$$\frac{d}{dx}C^*\nu_S N G_2(y_A) = \frac{d}{dx}(C^* - C)(\nu_S N G_2(x) + CG'(x))$$
$$C^*\nu_S N G_2'(y_A)\frac{dy_A}{dx} = (C^* - C)\left(\nu_S N G_2'(x) + CG''(x)\right)$$

and so

$$\begin{aligned}\frac{dy_A}{dx} &= \frac{(C^* - C)\left(\nu_S N G_2'(x) + CG''(x)\right)}{C^*\nu_S N G_2'(y_A)} \\ &= \frac{(C^* - C)\left(CG''(x) - \nu_S N x G''(x)\right)}{C^*\nu_S G_2'(y_A)} \\ &= -\frac{(C^* - C)\nu_S N G''(x)\left(x - \frac{C}{\nu_S N}\right)}{C^*\nu_S G_2'(y_A)} \\ &= \frac{(C^* - C)NG''(x)}{C^* y_A G''(y_A)}\left(x - \frac{C}{\nu_S N}\right). \end{aligned} \tag{2.113}$$

Now,

$$G''(x) = \frac{d}{dx}\frac{1+2x}{(1+x+x^2)^2} = -\frac{6x(1+x)}{(1+x+x^2)^3} \leq 0 \text{ for } x \geq 0$$

and so

$$\frac{(C^* - C)NG''(x)}{C^* y_A G''(y_A)} \geq 0 \text{ for } x, y_A \geq 0.$$

In particular, this means that

$$\operatorname{sgn}\left(\frac{dy_A}{dx}\right) = \operatorname{sgn}\left(x - \frac{C}{\nu_S N}\right). \tag{2.114}$$

Next, we do the same thing for (2.111), here using y_B to refer to y:

$$\frac{d}{dx}C^*G'(y_B) = \frac{d}{dx}\left(\nu_S N G_2(x) + CG'(x)\right)$$
$$C^*G''(y_B)\frac{dy_B}{dx} = \nu_S N G_2'(x) + CG''(x)$$

and so

$$\begin{aligned}\frac{dy_B}{dx} &= \frac{\nu_S N G_2'(x) + CG''(x)}{C^*G''(y_B)} \\ &= \frac{CG''(x) - \nu_S N x G''(x)}{C^*G''(y_B)} \\ &= -\frac{\nu_S N G''(x)}{C^*G''(y_B)}\left(x - \frac{C}{\nu_S N}\right).\end{aligned} \tag{2.115}$$

As before,

$$\frac{\nu_S N G''(x)}{C^*G''(y_B)} \geq 0 \text{ for } x, y_B \geq 0,$$

so in particular we have that

$$\operatorname{sgn}\left(\frac{dy_B}{dx}\right) = -\operatorname{sgn}\left(x - \frac{C}{\nu_S N}\right) \tag{2.116}$$

which means that

$$\operatorname{sgn}\left(\frac{dy_A}{dx}\right) = -\operatorname{sgn}\left(\frac{dy_B}{dx}\right). \tag{2.117}$$

Additionally, note that both derivatives vanish at $x = \frac{C}{\nu_S N}$. Finally, we will show that in fact the point $x = y$ where y is the unique solution to (2.112) is the only admissible solution to equations (2.110) and (2.111).

Lemma 2.10. *$x = y$ is the only admissible solution to equations* (2.110) *and* (2.111).

Proof. By Lemma 2.9 we have that $x = y$ solves equations (2.110) and (2.111).

Furthermore, as $y = \lim_{t\to\hat{t}^+} z_{1C}$ we have that

$$y = \lim_{t\to\hat{t}^+} \frac{u_{1C}C}{\nu_S N}$$

and as $u_{1C} \leq 1$ it must be the case that

$$y \leq \frac{C}{\nu_S N},$$

that is the $y = x$ intersection point must be at or before the point where both $\frac{dy_A}{dx}$ and $\frac{dy_B}{dx}$ are zero. There are two possible cases to consider here: either

$$\lim_{t\to\hat{t}^+} u_{1C}(t) = 1 \text{ or } \lim_{t\to\hat{t}^+} u_{1C}(t) < 1.$$

We will use the notation

$$\hat{u}_{1C}^+ := \lim_{t\to\hat{t}^+} u_{1C}(t).$$

1. Case 1: $\hat{u}_{1C}^+ = 1$

   In this case the $y = x$ intersection point happens at the point $\left(\frac{C}{\nu_S N}, \frac{C}{\nu_S N}\right)$, where $\frac{dy_A}{dx} = 0 = \frac{dy_B}{dx}$. We have from equations (2.114) and (2.116) that this $x = \frac{C}{\nu_S N}$ is the unique global minimizer for y_A and the unique global maximizer for y_B, meaning that there are no additional intersection points.

2. Case 2: $\hat{u}_{1C}^+ < 1$

   In this case the $y = x$ intersection point occurs at the point $\left(\frac{\hat{u}_{1C}^+ C}{\nu_S N}, \frac{\hat{u}_{1C}^+ C}{\nu_S N}\right)$, before $x = \frac{C}{\nu_S N}$ where $\frac{dy_A}{dx} = 0 = \frac{dy_B}{dx}$. Now, we have from (2.114) and (2.116) that y_A strictly decreases until $x = \frac{C}{\nu_S N}$ and then strictly increases after that point, whereas y_B strictly increases until $x = \frac{C}{\nu_S N}$ and then strictly decreases after that point, implying the existence of exactly one more intersection point at

some $x > \frac{C}{\nu_S N}$. Now, as we established in Lemma 2.8 there is only one solution for y, namely $y = \frac{\hat{u}_{1C}^{+} C}{\nu_S N}$. Therefore, in this case there is another intersection point (x, y) where $x > y$. Note that this is potentially admissibly because

$$x = \lim_{t \to \hat{t}^-} z_{1C} = \lim_{t \to \hat{t}^-} \frac{u_{1C} C}{u_{1N} \nu_S N}$$

which for admissible choices of the controls could be greater than $\frac{\hat{u}_{1C}^{+} C}{\nu_S N}$. As we will see, however, this is impossible. By equation (2.56) we have that

$$\lambda_1(\hat{t}) G'(y) = 1$$

and as G' is decreasing we have that for $x > y$ that

$$G'(x) < G'(y),$$

which in turn means that

$$\lambda_1(\hat{t}) G'(x) < \lambda_1(\hat{t}) G'(y) = 1,$$

and in particular that

$$\lambda_1(\hat{t}) G'(x) < 1.$$

However, by equations (2.32) and (2.33), this means that at this point

$$\frac{dH}{du_{1C}} < \frac{dH}{du_{0C}}.$$

which cannot happen during the balanced growth phase. Therefore this second intersection point is not admissible.

In both cases we have shown that the only admissible intersection point for the curves defined by equations (2.110) and (2.111) is at the single intersection point where $y = x$. □

Recalling that $x = \lim_{t\to\hat{t}^-} z_{1C}$ and $y = \lim_{t\to\hat{t}^+} z_{1C}$, we have shown that z_{1C} must be continuous at the boundary between the balanced growth phase and the penultimate interval.

2.4.8 Penultimate Interval to Final Interval Transition

Note that in the final interval of fruit-only growth, neither z_{1C} nor z_{2C} is defined, so we won't be addressing continuity of either one here. We will, however, analyze the controls at this transition and conclude that at this particular junction we actually have continuity of the controls.

First, recall that in Section 2.3.2, we showed that $\lim_{t\to t^{*-}} u_{1C}(t) = 0$. Because during the penultimate interval $u_{0C} = 1 - u_{1C}$ we have that $\lim_{t\to t^{*-}} u_{0C}(t) = 1$ as well. Note also that during the penultimate interval we have from (2.87) that $u_{2C} = 0, u_{1N} = 1$, and $u_{2N} = 0$, and by (2.98) we have that all the controls maintain their values in the final interval as well. Therefore, all the controls are continuous at $t = t^*$, the boundary between the penultimate interval and the final interval of fruit-only growth.

Next, we will prove the following lemma which shows that the transition to fruit-only growth occurs rapidly at the end of the penultimate interval.

Lemma 2.11.

$$\lim_{t\to t^{*-}} \frac{du_{1C}}{dt}(t) = -\infty.$$

Proof. First, recall that by (2.57), $z_{1C} \to 0^+$ as $t \to t^{*^-}$. Also, by (2.97) we have that

$$\frac{dz_{1C}}{dt} \sim -\frac{C_S}{6z_{1C}}, \text{ as } \quad z_{1C} \to 0,$$

which means that

$$\lim_{t \to t^{*^-}} \frac{dz_{1C}}{dt} = \lim_{z_{1C} \to 0^+} -\frac{C_S}{6z_{1C}} = -\infty. \tag{2.118}$$

Furthermore, by (2.88) and the fact that here $N = N^*$, we have that

$$u_{1C} = \frac{\nu_S z_{1C} N^*}{C}. \tag{2.119}$$

Differentiating (2.119) with respect to t,

$$\begin{aligned} \frac{du_{1C}}{dt} &= \frac{\nu_S N^* C \frac{dz_{1C}}{dt} - \nu_S N^* z_{1C} C_S \frac{dS}{dt}}{C^2} \\ &= \frac{\nu_S N^*}{C} \frac{dz_{1C}}{dt} - \left(\frac{\nu_S N^*}{C}\right)^2 C_S G(z_{1C}) \end{aligned}$$

Now, using the fact that S and C are continuous, C^* and C_S^* are finite and non-zero, $G(0) = 0$, and (2.118), we have that

$$\begin{aligned} \lim_{t \to t^{*^-}} \frac{du_{1C}}{dt} &= \lim_{z_{1C} \to 0^+} \frac{\nu_S N^*}{C} \frac{dz_{1C}}{dt} - \left(\frac{\nu_S N^*}{C}\right)^2 C_S G(z_{1C}) \\ &= \frac{\nu_S N^*}{C^*} \lim_{z_{1C} \to 0^+} \frac{dz_{1C}}{dt} - \left(\frac{\nu_S N^*}{C^*}\right)^2 C_S^* \lim_{z_{1C} \to 0^+} G(z_{1C}) \\ &= -\infty. \end{aligned}$$

□

2.5 Numerical Scheme

In this section, we will outline the numerical scheme we developed for solving the optimal control problem (2.17). Because the solution exhibits a multi-phase structure, standard methods of solving optimal control problems, such as forward-backward sweep or shooting [8] aren't well-suited. Broadly speaking, our approach is to construct a numerical scheme for solving the problem backward in time. There are two primary components to the numerical scheme. In the first, we obtain the map

$$(S^*, R^*) \mapsto (S, R, F, \vec{u}, \lambda_1, \lambda_2). \tag{2.120}$$

In the second component we use MATLAB's built-in nonlinear equation solver `fsolve` to find the map

$$(S_0, R_0) \mapsto (S^*, R^*).$$

Ultimately, then, we obtain the map

$$(S_0, R_0) \mapsto (S, R, F, \vec{u}, \lambda_1, \lambda_2).$$

We will now direct out attention the numerical scheme for finding (2.120). We find the solution in the penultimate interval, final interval, balanced growth phase, and initial phase, respectively. During each phase, we use the fourth-order Runge-Kutta method (RK4) to solve differential equations which govern the phase dynamics, and use derived algebraic equations to update the controls. Taking this approach allows us to avoid having to update the controls iteratively. We will use the fact that the Hamiltonian is constant along the optimal trajectory, as well as the information we have about transitions from Section 2.4, to find the boundaries between phases

and ultimately stitch the four phases together to form one complete solution. We will discuss how the numerical scheme works without getting into the fine numerical details. The actual MATLAB code is included in Appendix C.1.

2.5.1 Penultimate Interval

We begin implementing this stage by finding the end of the penultimate interval (t^*) using equation (2.52). Now, as the dynamics during the penultimate interval depend on z_{1C}, we would ideally use (2.97) and (2.89) to solve for z_{1C} and S simultaneously and use (2.119) to update the controls. However, recall that by the proof of Lemma 2.11, in particular equation 2.118, z_{1C} approaches a vertical tangent as $t \to t^{*^-}$, which makes solving for z_{1C} numerically difficult. Note, however, that by the same reasoning we have that $\frac{dt}{dz_{1C}}$ approaches 0 as $z_{1C} \to 0^+$. So, to remedy this numerical difficulty, we take advantage of this fact, along with the fact that z_{1C} is monotonically decreasing in time throughout the penultimate interval, to think of z_{1C} as our 'time' during the penultimate interval and derive differential equations for t, S, and λ_2 in terms of z_{1C}. This, combined with the fact that during the penultimate interval R is constant, λ_1 is algebraically related to z_{1C} via (2.56), and F depends on the value of $\hat{t}$, gives us everything we need to find the solution during this phase numerically. Note that because z_{1C} decreases as t increases, solving forward in z_{1C} is tantamount to solving backward in time.

To this end, note that by (2.97) we have that

$$\frac{dt}{dz_{1C}} = \frac{G''(z_{1C})}{C_S \left[G'(z_{1C})\right]^2}, \quad t(0) = t^*. \tag{2.121}$$

Furthermore, using (2.121) we obtain

$$\frac{d\lambda_2}{dz_{1C}} = \frac{d\lambda_2}{dt}\frac{dt}{dz_{1C}} = -\frac{N_R\nu_S G'\left(1/z_{1C}\right)G''(z_{1C})}{C_S\left[G'(z_{1C})\right]^3}, \quad \lambda_2(0) = 0 \tag{2.122}$$

$$\frac{dS}{dz_{1C}} = \frac{dS}{dt}\frac{dt}{dz_{1C}} = \frac{\nu_S N G(z_{1C})G''(z_{1C})}{C_S\left[G'(z_{1C})\right]^2}, \qquad S(0) = S^*. \tag{2.123}$$

We use RK4 to solve (2.121), (2.122), and (2.123) forward in z_{1C}, and use (2.119) to update u_{1C}. Also, recall that $\lambda_1 = 1/G'(z_{1C})$, $u_{0C} = 1 - u_{1C}$, and the other three controls are constant by (2.87). Upon reordering by t, this gives us all of the states, adjoints, and controls during the penultimate interval, with the exception of fruits, which we will determine after we identify the correct transition point from balanced growth to the penultimate interval ($\hat{t}$).

We stop solving forward in z_{1C} when u_{1C} becomes unbounded, as this gives us an upper bound on the value of z_{1C} (lower bound for the value of t) for which the transition from balanced growth to the penultimate interval can occur. Furthermore, because during the balanced growth phase we have by (2.76) that $\lambda_2 G'(z_{2C}) > 1$, it must be the case that $\lambda_2 \geq 1$ at the balanced growth-penultimate interval boundary because G' is bounded between 0 and 1. Finding where $\lambda_2 = 1$ gives us a lower bound on the value of z_{1C} (upper bound for the value of t) for which the transition from balanced growth to the penultimate interval can occur. This gives us an interval $[z_{1C}^{\min}, z_{1C}^{\max}]$ in which to search for $\hat{t}$.

2.5.2 Locating the Start of the Penultimate Interval

This portion of the numerical scheme begins with the interval $[z_{1C}^{\min}, z_{1C}^{\max}]$ identified in Section 2.5.1. Since we know that the Hamiltonian must be constant along the optimal trajectory, we use the balanced-growth specific formulation of H in terms of

λ_2 and z_{2C}, given by (2.85):

$$H = C^* = \lambda_2 \left[\nu_R N G_2(z_{2C}) + C G'(z_{2C})\right]. \qquad \text{((2.85) revisited)}$$

Starting with $[z_{1C}^{\min}, z_{1C}^{\max}]$, we use a binary search to locate the smallest interval, relative to current RK4 step size, which contains the point at which (2.85) is satisfied. At this point we use a smaller RK4 step size, and use the same procedure discussed in Section 2.5.1 to increase the resolution of the solution and repeat the binary search until we find a point that satisfies (2.85) to within some specified tolerance. We call this point $\hat{t}$, and record the values of the states, adjoints, and controls.

One crucial feature on this binary search which we have not mentioned so far is the fact that since z_{2C} is not defined in the penultimate interval, we need to compute its limit from balanced growth at every iteration in the binary search. This is made possible by the fact we established in Section 2.4.7, that z_{1C} is continuous at $\hat{t}$. Therefore, we can use equations (2.76) and (2.77) to obtain

$$\frac{G'(z_{2C})}{G_2(z_{2C})} = \frac{\nu_R}{\nu_S} \frac{G'(z_{1C})}{G_2(z_{1C})}. \tag{2.124}$$

Using MATLAB's built-in nonlinear equation solver `fsolve`, we use the known value of z_{1C} at a particular point in the penultimate interval to calculate what z_{2C} would be if the balanced growth phase ended at that point. This value of z_{2C} is then used to evaluate the right-hand side of (2.85) in the binary search.

2.5.3 Fruits - Penultimate Interval

Recall that in Section 2.5.1 we were unable to solve for F because we didn't know the value of $\hat{t}$. This was resolved in Section 2.5.2, so at this point we use RK4 to solve

(2.90) forward in time during the penultimate interval. This completes the numerical solution for the penultimate interval.

2.5.4 Final Interval

Before continuing backward in time we take advantage of the fact that, by the process described in Section 2.5.3, we now know the value of F at the beginning of the final interval. Here the controls are constant by (2.98), S, R, and λ_2 are constant, and λ_1 is given by (2.46). Lastly, using the value of $F(t^*)$ obtained as described in Section 2.5.3, and the fact that F' is constant during the final interval by (2.99), we have that

$$F(t) = F(t^*) + C^* \cdot (t - t^*). \tag{2.125}$$

Therefore, this portion of the numerical scheme consists only of defining the states, adjoints, and controls as defined above between time t^* and time T.

2.5.5 Balanced Growth

Upon locating $\hat{t}$ as discussed in Section 2.5.2, we have the value of the states, adjoints, z_{1C}, and z_{2C} at the end of the balanced growth stage. Recall that the controls need not be continuous, so we do not immediately know their values at the end of balanced growth. We can, however, start with the equations for z_{1C} and z_{2C} during balanced growth

$$z_{1C} = \frac{u_{1C}C}{\nu_S u_{1N} N} \tag{2.126}$$

$$z_{2C} = \frac{(1 - u_{1C})C}{\nu_R (1 - u_{1N}) N} \tag{2.127}$$

and solve for u_{1N} and u_{1C}:

$$u_{1N} = \frac{\frac{C}{N} - \nu_R z_{2C}}{\nu_S z_{1C} - \nu_R z_{2C}} \tag{2.128}$$

$$u_{1C} = \frac{\nu_S z_{1C} N u_{1N}}{C}. \tag{2.129}$$

We derive differential equations for z_{1C} and z_{2C} during this phase, and ultimately use (2.128) and (2.129) to obtain the controls. The differential equations in time for z_{1C} and z_{2C} are given below, and the derivation is included in Appendix B.2.

$$\frac{dz_{1C}}{dt} = \frac{N_R \nu_S \nu_R G(z_{2C}) G_2(z_{1C}) - C_S G'(z_{1C}) \left[\nu_S G_2(z_{1C}) + z_{2C} \nu_R G'(z_{1C})\right]}{G''(z_{1C}) \left[\nu_S z_{1C} - \nu_R z_{2C}\right]} \tag{2.130}$$

$$\frac{dz_{2C}}{dt} = \frac{N_R \nu_S G_2(z_{1C}) \left[G'(z_{2C})\right]^2 - C_S \left[G'(z_{1C})\right]^2 G'(z_{2C}) + G''(z_{1C}) G'(z_{2C}) \frac{dz_{1C}}{dt}}{G'(z_{1C}) G''(z_{2C})}. \tag{2.131}$$

Beginning at $\hat{t}$, we use RK4 to simultaneously solve equations (2.78), (2.79), (2.80), (2.81), (2.130), and (2.131) backward in time to obtain $S, R, \lambda_1, \lambda_2, z_{1C}$, and z_{2C}, respectively. We use (2.129) and (2.128) to eliminate u_{1C} and u_{1N} in (2.78) and (2.79) so that these six differential equations are expressed exclusively in terms of these six variables. We then use (2.129) and (2.128) to obtain u_{1C} and u_{1N}, as well as the rest of the controls during this phase via (2.75). As with the penultimate interval, we continue backward in time until one of the controls leaves the interval $[0, 1]$. This gives us the earliest time for the transition point between the initial and balanced growth phases.

2.5.6 Locating the Start of Balanced Growth

Here we will discuss the portion of the numerical scheme involved in finding time $t = \bar{t}$, the start of balanced growth. Recall that by Section 2.4.6, z_{1C} is continuous

at the boundary between an initial phase of shoot growth and the balanced growth phase, and z_{2C} is continuous at the boundary between an initial phase of root growth and the balanced growth phase. In either case, the transition must occur at a point where

$$u_{1C} - u_{1N} = 0. \tag{2.132}$$

This is because in the case where the initial phase consists of shoot growth we have

$$\lim_{t \to \bar{t}^+} z_{1C} = \lim_{t \to \bar{t}^+} \frac{u_{1C} C}{\nu_S u_{1N} N} = \lim_{t \to \bar{t}} \frac{C}{\nu_S N} \implies \lim_{t \to \bar{t}^+} \frac{u_{1C}}{u_{1N}} = 1$$

and in the case where the initial phase consists of root growth we have

$$\lim_{t \to \bar{t}^+} z_{1C} = \lim_{t \to \bar{t}^+} \frac{(1 - u_{1C}) C}{\nu_S (1 - u_{1N}) N} = \lim_{t \to \bar{t}} \frac{C}{\nu_S N} \implies \lim_{t \to \bar{t}^+} \frac{(1 - u_{1C})}{(1 - u_{1N})} = 1.$$

We use essentially the same procedure employed in Section 2.5.2 for refining the transition point between the balanced growth and the penultimate intervals, with the exception that finding $\bar{t}$ doesn't require computing the limits of any quantities from the earlier phase, as we had to do with z_{2C} at $\hat{t}$. We use a binary search to locate the smallest interval about which (2.132) is satisfied, and then use RK4 to solve equations (2.78), (2.79), (2.80), (2.81), (2.130), and (2.131) backward in time on a smaller integration mesh. We repeat this process until we have found a point at which (2.132) is met to within some specified tolerance. We call this point $\bar{t}$.

It is important to note that if no such point exists then the plant begins in the balanced growth phase, in which case there are only three phases instead of four. If there is such a point $\bar{t}$, the next step in the numerical scheme is to determine whether the initial phase consists of shoot growth or root growth. Using (2.65) and (2.73), we

compute the following at $\bar{t}$:

$$\left|\lambda_1 \nu_S N G\left(\frac{C}{\nu_S N}\right) - C^*\right| \quad \text{and} \quad \left|\lambda_2 \nu_R N G\left(\frac{C}{\nu_R N}\right) - C^*\right|.$$

If the first is smaller then the initial phase consists of shoot growth, and if the second is smaller the initial phase consists of root growth.

2.5.7 Initial Phase

2.5.7.1 Shoot-Only Growth

In the case that the first phase consists of shoot-only growth, we use RK4 to solve (2.60), (2.61), and (2.62) simultaneously backward in time until we reach $t = 0$. This gives us S, λ_1, and λ_2, respectively. The controls are constant here and given by (2.59), and R is a constant determined by it's value at the end of the initial phase.

2.5.7.2 Root-Only Growth

In the case that the first phase consists of root-only growth, we use RK4 to solve (2.68), (2.69), and (2.70) simultaneously backward in time until we reach $t = 0$. This gives us R, λ_1, and λ_2, respectively. The controls are constant here and given by (2.67), and S is a constant determined by it's value at the end of the initial phase.

2.6 Numerical Results

In this section, we will present some of the primary results that are apparent from the numerical simulations of the model. In particular, we will look at several 'representative' simulation results that showcase some of the different strategies a plant can employ to maximize fruit production, as well as some results which help us understand

the relationship between initial and terminal conditions. In all of the simulations we will discuss, we have made the simplifying assumption that $C(S) = S$ and $N(R) = R$, and have chosen $\nu_R = 1, \nu_S = 1/3$, and $T = 10$. The choices of C and N ignore any possibility of self-shading as the plant grows. We chose $\nu_R = 1$ for convenience, and chose $\nu_S = 1/3$. Note that this results in $\frac{\nu_R}{\nu_S} = 3$, which is likely a bit higher than data suggests (see [4, 10]), but had the effect of exaggerating the lengths of the different phases, resulting in easier-to-interpret plots. We chose $T = 10$ because in the case that $C(S) = S$, we have by (2.52) that $t^* = 9$, which again facilitated our interpretation of the numerical results. Furthermore, it can be shown that with these choices of C and N, the length of the penultimate interval is fixed for given choices of the stoichiometric ratios. We will begin with four examples of different optimal growth strategies which all reach the same optimal value of fruits at time T.

2.6.1 Initial Shoot Growth

Here we chose terminal conditions $S^* = 223.20$ and $R^* = 112.86$. This results in $F(T) = 900$, $S_0 = 17$, and $R_0 = 84$. In this simulation we see the full four-stage structure of the solution. There is an initial phase of shoot growth, followed by a period of balanced growth between shoots and roots, a penultimate interval of shoot and fruit growth, and finally a period of fruit-only growth at the end of the growing season. The states and controls are shown in Figure 2.2.

The lower two plots in Figure 2.2 show the carbon and nitrogen allocation strategies for this plant. Of particular note, we see that following the initial period of shoot growth, there is a short period during which u_{2C} and u_{2N} are increasing, signifying a period of increasing root production. Shortly after time $t = 2$, we see that u_{2C} and u_{2N} begin to decrease again, signifying that although it is still advantageous to be investing resources into roots, the plant ultimately needs to prioritize shoot production

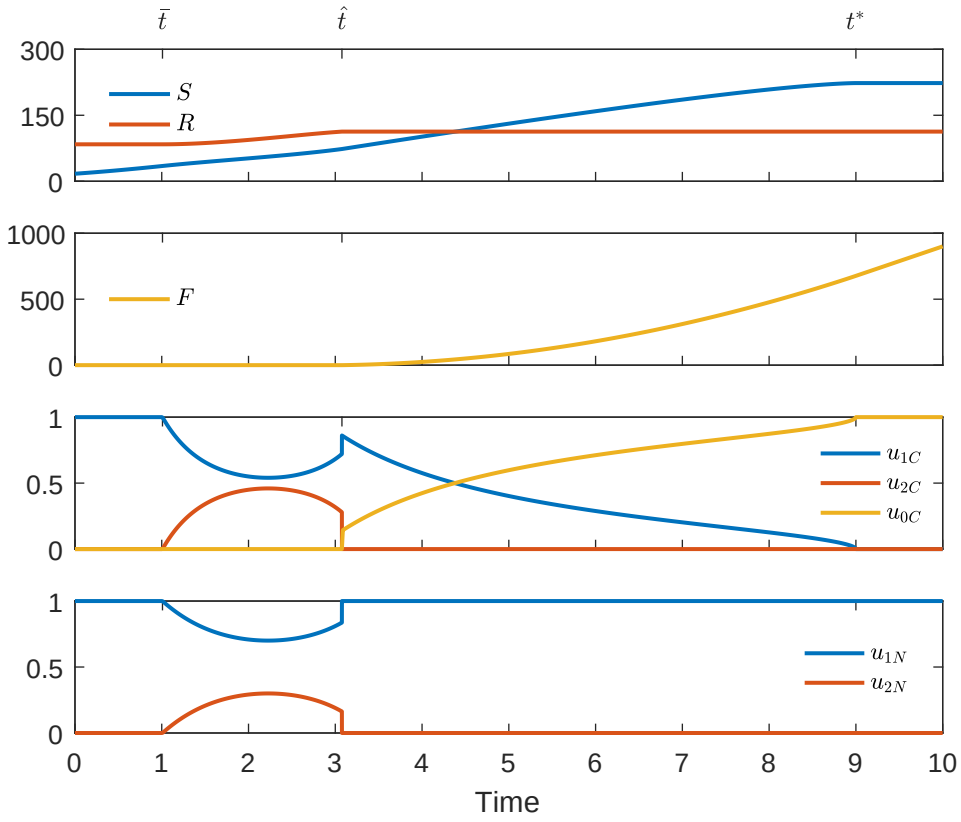

Figure 2.2: States and controls illustrating initial shoot growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

again to prepare for the penultimate interval of mixed shoot/fruit growth. During the penultimate interval we see the plant gradually stop investing in shoots, before switching to fruit-only production at time $t = 9$.

2.6.2 Initial Root Growth

Here we chose terminal conditions $S^* = 222.11$ and $R^* = 111.55$. This results in $F(T) = 900$, $S_0 = 48.9$, and $R_0 = 28.9$. In this simulation we again see the four-stage structure as in Figure 2.2, however here we see an initial phase of root growth instead of the initial phase of shoot growth we saw previously. The initial conditions of the two simulations are quite different, but the terminal conditions are nearly the same.

This, as we will discuss later, suggests that the model predicts that initial transients tend to go away in the first stage of growth. The states and controls are shown in Figure 2.3.

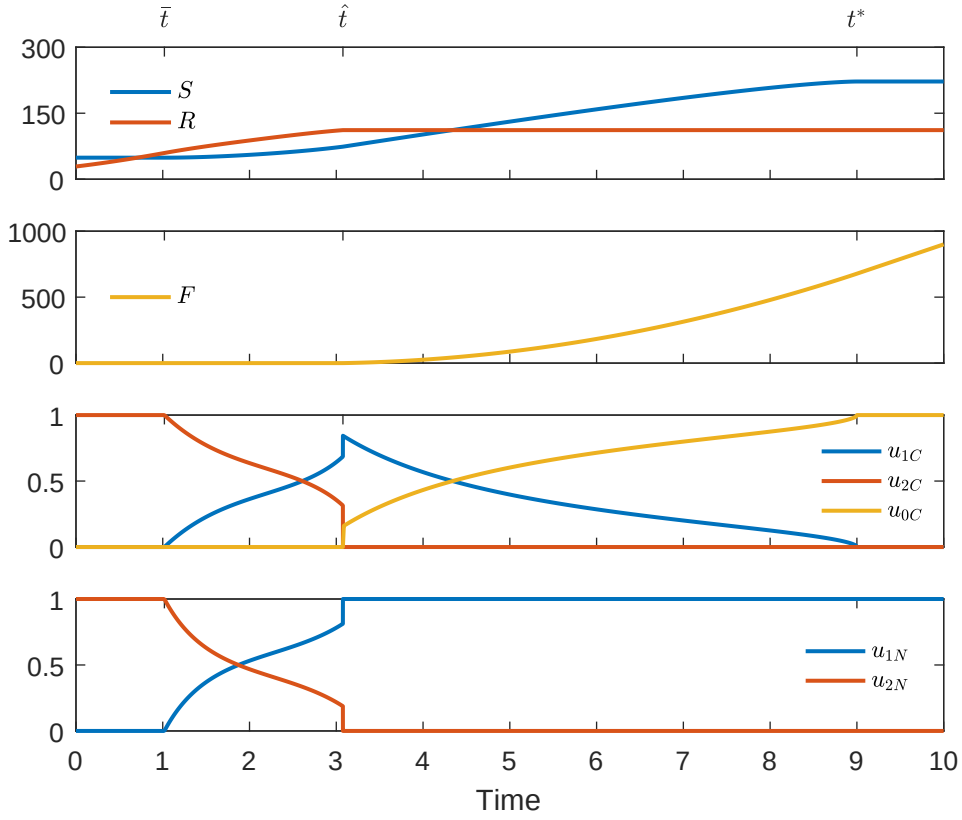

Figure 2.3: States and controls illustrating initial root growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

The two graphs at the bottom of Figure 2.3 show a different allocation strategy than appeared in the case of initial shoot growth in Figure 2.2. In particular, following the initial period of root growth, we see a decline in both carbon and nitrogen allocation to the roots throughout the entire balanced growth phase, and the steady increase in allocation to shoots. Following the balanced growth phase we again see a similar penultimate interval of mixed shoot-fruit growth before the final interval of fruit growth beginning at $t = 9$.

2.6.3 Balanced Growth First - Type S

Here we chose terminal conditions $S^* = 222.71$ and $R^* = 112.27$. This results in $F(T) = 900$, $S_0 = 30.3$, and $R_0 = 59.74$. The states and controls are shown in Figure 2.4.

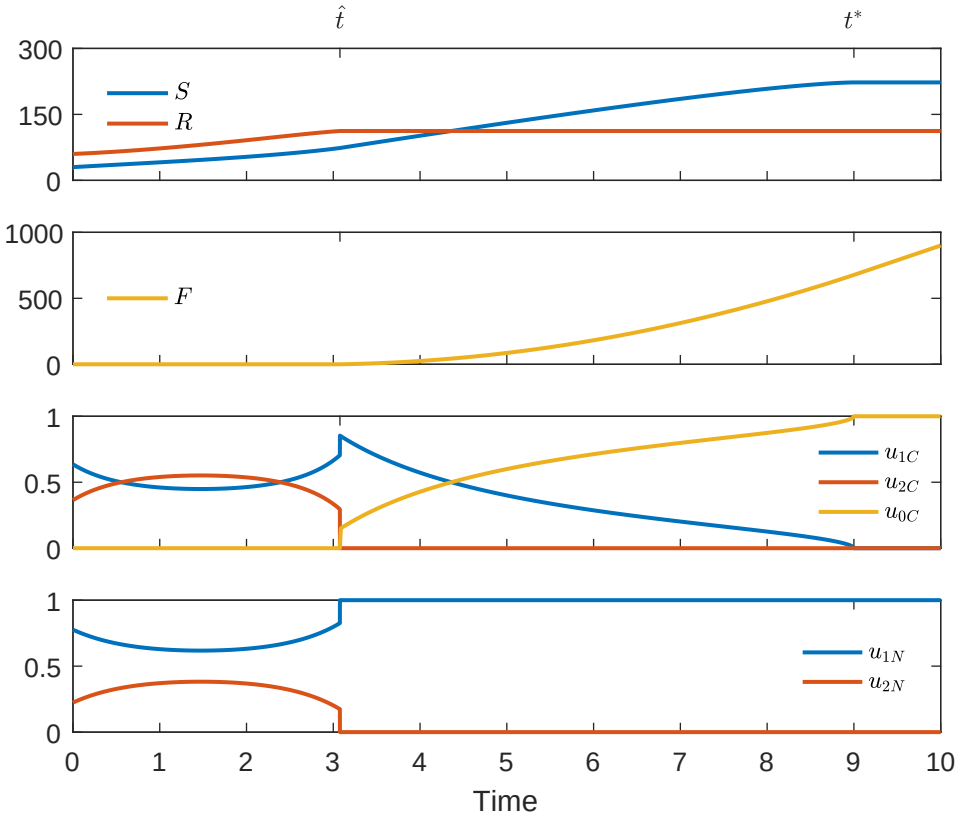

Figure 2.4: States and controls illustrating Type S initial balanced growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

In this simulation, we see an example of a plant which starts in balance, and therefore forgoes the initial phase of shoot-only or root-only growth. That said, comparing Figure 2.4 to Figure 2.2, we see a similar allocation strategy during balanced growth. In particular, both show a gradual decrease in allocation to shoots followed by a gradual increase in allocation to shoots throughout the balanced growth phase. This

behavior corresponds to initial conditions from which the plant begins in balanced growth, but are more biased toward starting shoot-deficient than root-deficient. For this reason, we refer to this type of initially balanced growth as 'Type S.'

2.6.4 Balanced Growth First - Type R

Here we chose terminal conditions $S^* = 222.60$ and $R^* = 112.14$. This results in $F(T) = 900$, $S_0 = 36.16$, and $R_0 = 49.58$. The states and controls are shown in Figure 2.5.

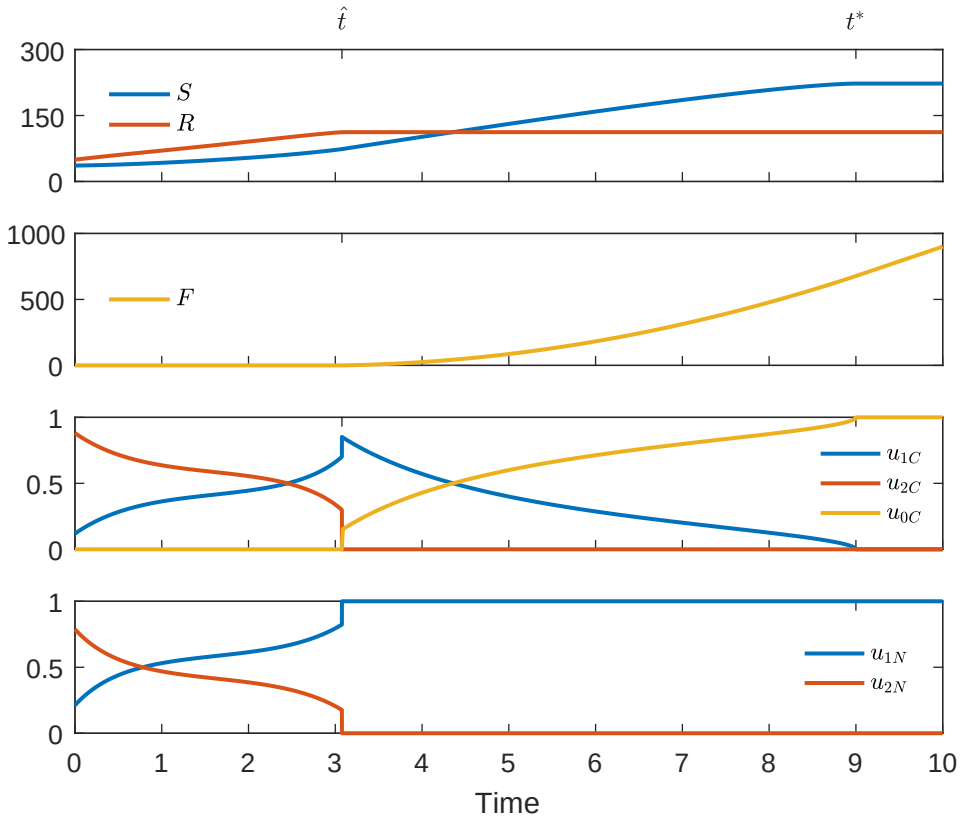

Figure 2.5: States and controls illustrating Type R initial balanced growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

Figure 2.5 provides us with another example of a plant that begins in balanced growth, skipping over the initial phase. Unlike Figure 2.4, the balanced growth phase

in this simulation has a similar structure to the balanced growth phase in Figure 2.3, where the plant had an initial phase of root growth. In both this simulation and the one shown in Figure 2.3, we see that the balanced growth phase consists of steadily increasing the allocation to shoots throughout the interval while simultaneously decreasing the allocation to roots. In this case the plant is initially more biased toward being root-deficient than shoot-deficient. So, while the initial phase of root-only growth is unnecessary, the early growth sees a greater investment in roots than in shoots. For this reason we refer to this type of initially balanced growth as 'Type R.'

2.6.5 Final Fruits Value Contours

In order to better understand the relationship between initial conditions and the optimal final value of fruits, we looked for points in the (S_0, R_0)-plane for which the final value of fruits was 700, 800, or 900. For a given value of S_0, we used MATLAB's built-in nonlinear equation solver `fsolve` to find the appropriate value of R_0 for which the numerical scheme outlined in Section 2.5 would yield either 700, 800, or 900 for $F(T)$. We plotted the resulting contours in the initial condition plane as seen in Figure 2.6.

There are several key features of this plot to notice. First is the wide range of initial conditions for which a particular final value of fruits is optimal. Depending on the allocation strategy, different plants may be able to reach the same level of fruit yield even though they may start with very different initial conditions. Additionally, we can see from Figure 2.6 how much more initial shoots or roots would be necessary at a particular point along a contour to move the plant to a contour with a higher fruit yield. Looking at the ends of the contours, it takes relatively little in the way of additional initial structure to move from one contour to the next, whereas in the middle of the contours we see that a larger increase in initial biomass is required to

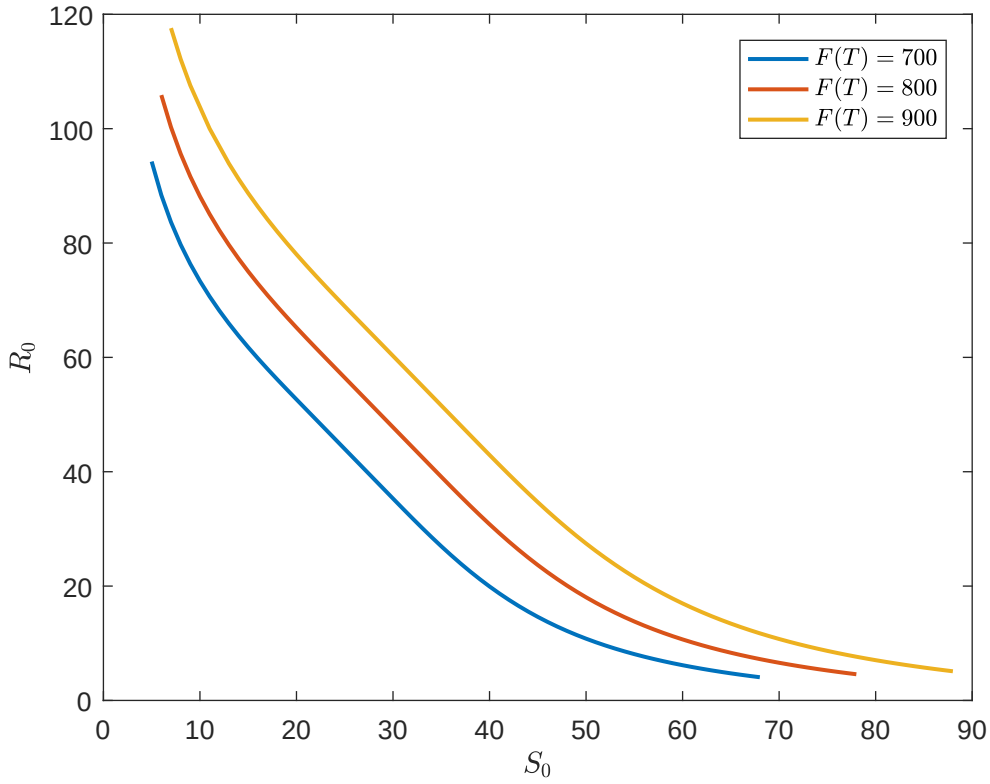

Figure 2.6: Contours depicting which initial conditions correspond to final fruit values of 700, 800, and 900. Shoots, roots, and fruits are given in units of carbon

move to the next contour. As we will see in the next section, these middle regions contain initial conditions from which the plant begins in balance. Nearer to the ends of the contours, where the concavity becomes more pronounced, we see that for a plant to recover from an initial imbalance it must be more biased toward the vegetative organ with the most biomass.

2.6.6 900 Fruit Contour

To get more of an understanding of the results we are seeing in Figure 2.6, we take a closer look at the contour along which each initial condition results in $F(T) = 900$, and in particular we considered 80 values of S_0 between 7 and 88. We will begin by identifying which of the four types of growth we identified in Sections 2.6.1, 2.6.2,

2.6.3 and 2.6.4 each initial condition corresponds too. The $F(T) = 900$ contour is plotted again in Figure 2.7. The coloration indicates the initial phase of growth.

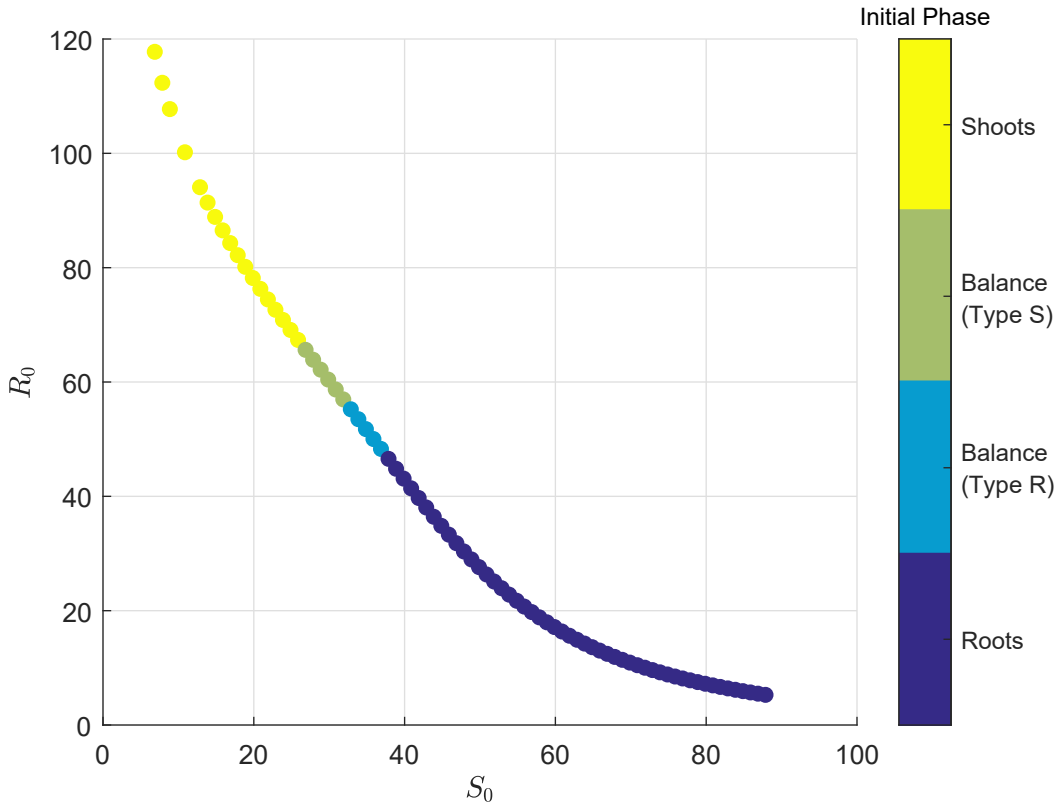

Figure 2.7: $F(T) = 900$ contour with coloration indicating the initial phase of growth. Shoots, roots, and fruits are given in units of carbon

We next consider the terminal conditions (S^*, R^*) which each point along the contour shown in Figure 2.7 corresponds to. In Figure 2.8, we see two different plots of the terminal conditions corresponding to the points along the $F(T) = 900$ contour. The plot on the left shows the terminal conditions following the same color scheme used in Figure 2.7, and the plot on the right shows the frequency of points along that line through the (S^*, R^*) plane. The frequency plot was generated using the MATLAB function `hist3` with 2.5×2.5 bins in the (S^*, R^*) plane.

Note that the right-most plot in Figure 2.8 shows that the vast majority of terminal

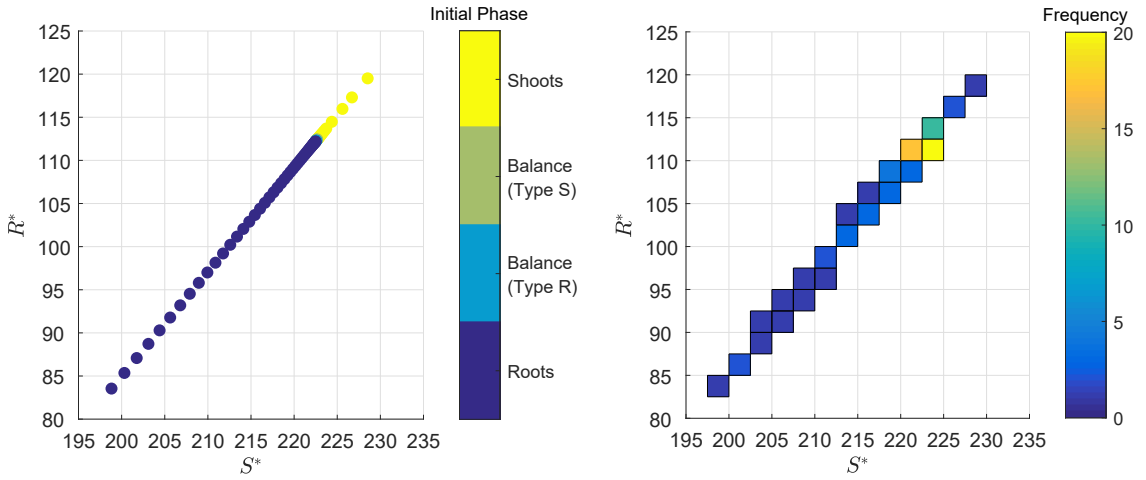

Figure 2.8: The plot on the left shows the terminal conditions corresponding to initial conditions along the $F(T) = 900$ contour. The plot on the right shows the corresponding frequency of points along the line on the left with points grouped into 2.5×2.5 bins. Shoots, roots, and fruits are given in units of carbon.

conditions are concentrated in a small region of the (S^*, R^*)-plane, which suggests that optimal growth for most of the points along the $F(T) = 900$ contour means reaching a common final configuration of biomass. Additionally, looking at the left-most plot in Figure 2.8 along the $F(T) = 900$ contour, the final amount of roots varies linearly with the final value of shoots. Furthermore, looking at the coloration on this plot, we see that plants which began with a phase of root-only growth required less overall biomass to reach the $F(T) = 900$ contour than those which began with shoot-only growth. The region of the terminal condition plane which corresponds to plants which begin in balanced growth corresponds to a very small region where the terminal conditions are most dense.

It is worth pointing out here that Figures 2.2, 2.4, 2.5, and 2.3 (in that order) represent 'snapshots' along a spectrum of outcomes as we move left to right along the $F(T) = 900$ contour in Figure 2.7. We can visualize this by overlaying each of the plots of either shoots vs. time or roots vs. time for each of the initial conditions along

the $F(T) = 900$ contour, and using a color scale to indicate the progression along the contour. This is shown in Figure 2.9, which shows $S(t)$ and $R(t)$ for each of the initial conditions along the $F(T) = 900$ contour. As the initial conditions progress from left to right along the contour, the color of the plots of $S(t)$ and $R(t)$ transition from blue to red. That is, the solution curves based on the initial condition on the far left of the contour are solid blue, and the solution curves based on the initial conditions on the far right of the contour are solid red, and as the initial conditions progress from left to right, the color of the corresponding solution curves gradually changes from blue to red.

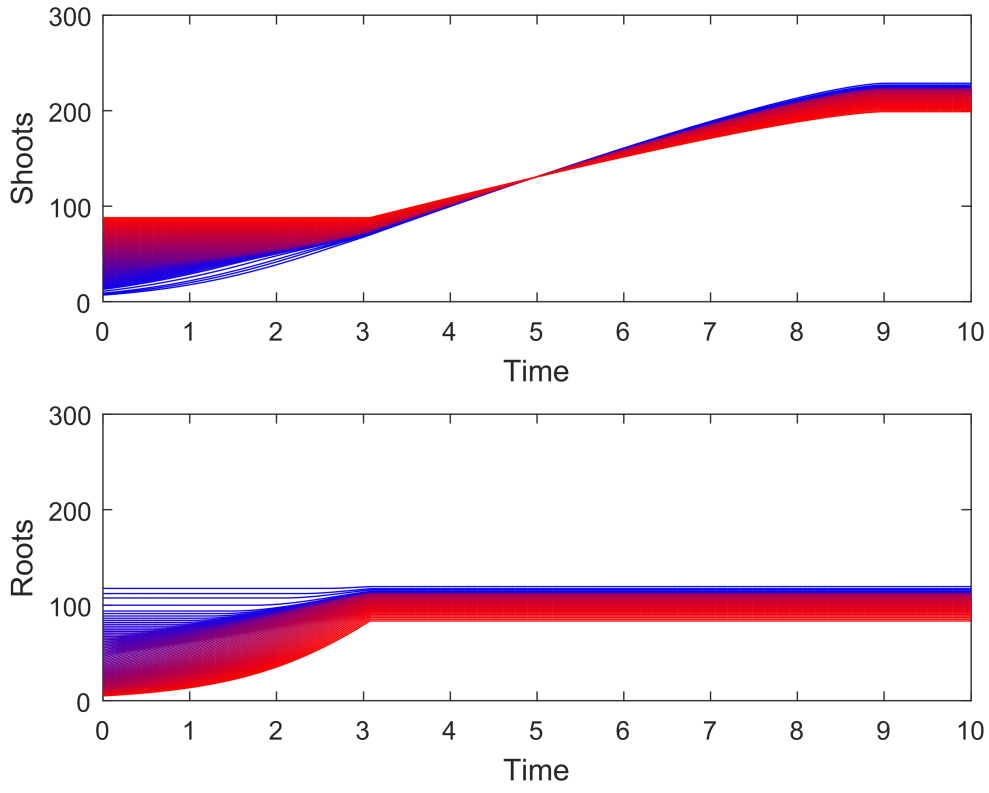

Figure 2.9: Progression of solutions curves of $S(t)$ and $R(t)$ along the $F(T) = 900$ contour. Going from blue to red corresponds to moving from left to right along the contour. Shoots, roots, and fruits are given in units of carbon.

Notice that although there is a clear gradient from blue to red in the initial phases,

this transition is much sharper once the roots stop growing. This reinforces some of what we have seen thus far. In particular, we see that plants that begin with an excess of shoots compared to roots require less biomass overall to achieve the same outcome in regard to fruits. We also see that, in the top plot, the plants which begin with the least amount of initial shoots need to make up for it by producing the most shoots by the end of the growing season to reach the optimal amount of fruits. Furthermore, this figure reinforces the idea that the common final value of fruits we see along the $F(T) = 900$ contour comes about by driving a wide range of initial conditions toward a similar point by the end of balanced growth.

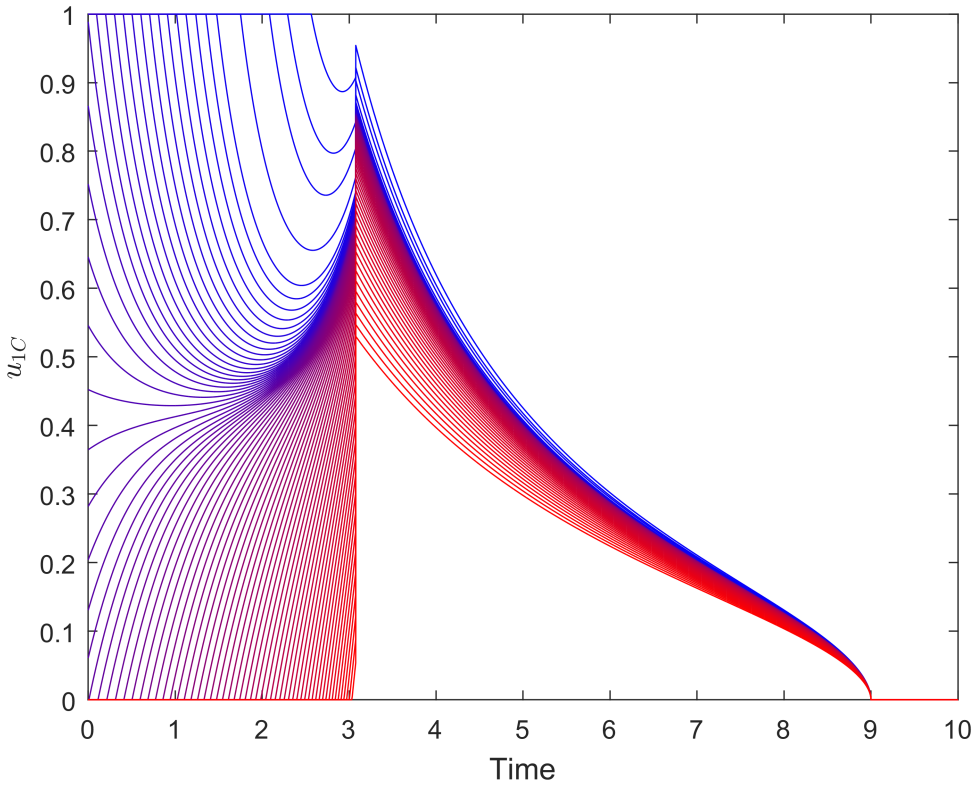

Figure 2.10: Progression of solution curves of $u_{1C}(t)$ (fraction of carbon allocated to shoots) as the initial conditions (S_0, R_0) move along the $F(T) = 900$ contour. Going from blue to red corresponds to moving from left to right along the contour.

To delve a little bit further into the different allocation strategies employed by

plants with initial conditions along the $F(T) = 900$ contour, we will look at how the trajectory of carbon allocation to shoots (u_{1C}) changes along this contour. We again use a color gradient from blue to red to show how the $u_{1C}(t)$ solution curves changes as the initial conditions progress from left to right along the contour in Figure 2.7. These solutions are shown in Figure 2.10.

We see in Figure 2.10 a gradual transition from the initial stage consisting of shoot-only growth to the initial stage consisting of root-only growth. Additionally, we again see the difference in the level of variation between the early phases of growth and the later phases of growth. There is a substantial amount of variation in the initial and balanced growth phases, but relatively little in the penultimate interval. As we move toward the region of the $F(T) = 900$ curve that consists of initial conditions which have an excess of shoots we begin to see allocation of carbon to shoots decrease at the beginning of the penultimate interval, but for the majority of initial conditions the penultimate interval shows only minimal variation in the optimal allocation strategy.

2.7 Discussion

By incorporating nitrogen into the model, but keeping the fruits solely reliant on carbon, we have essentially considered a case in which the C:N ratio in fruits is infinite. While this isn't biologically reasonable for most annual plants, it does provide a mathematically approachable framework in which to begin to analyze the dynamics of allocation of two resources in annual plants. It is worth noting, however, that if we were to exchange the assumption that fruits are solely carbon with the assumption that the fruits are solely nitrogen, we would expect the model to predict a reversal in the roles of shoots and roots. The assumption that fruits are carbon-only, however, seems a more natural extension of Iwasa and Roughgarden's work in [5].

An obvious outcome of our model, not present in [5], is the penultimate interval, during which the shoots and fruits grow together. Here the C:N ratio in fruits is greater than the C:N ratios in either shoots or roots, and so we see a phase during which shoot production is overall more important to eventual fruit yield, but an increased capacity to assimilate nitrogen is not useful so any excess carbon is invested in fruits. This observation will play a key role in Chapter 3 in how we approach our extension of this model to the case where fruits require both resources.

The theoretical results obtained in Section 2.4 confer a degree of biological relevance to the model that arguably adds credence to the model despite the narrow scope imposed by the assumption that fruits are built solely from carbon. In particular, we showed that z_{1C}, z_{2C}, and u_{0C} are continuous between any two phases in which they are defined and non-zero. Note that while z_{1C} and z_{2C} are dimensionless, they are multiples of the ratio of carbon flux to nitrogen flux in both shoots in roots. So, the fact that z_{1C} and z_{2C} are continuous at these junctions means that these ratios are continuous in both the shoots and roots. This is somewhat striking, given that, while these individual fluxes are continuous at the beginning of balanced growth, they are markedly discontinuous at the end of balanced growth in Figures 2.2, 2.3, 2.4, and 2.5. What this says is that, so long as either the shoots or roots is growing, the amount of allocated carbon per unit of allocated nitrogen varies continuously, which is reasonable to expect from biochemical processes. Furthermore, by equation 2.24, the fact that u_{0C} is continuous between the penultimate and final intervals means that the rate of fruit growth is continuous between these phases.

The results presented in Section 2.6 provide several avenues for drawing more general conclusions about the nature of plant growth that optimizes fruit yield. First, we can see in Figures 2.2, 2.3, 2.4, and 2.5 that the allocation strategy is a balancing act between preventing limitations due to nutrient deficiency and investing in the

organ which most directly contributes to increased fruit yield. This is evidenced by the fact that the plant invests in the most deficient organ until it can efficiently invest in both shoots and roots, which it is does in such a way that by the end of balanced growth we see an increase in carbon flux to shoots. During the penultimate interval, then, fruits are a better investment than roots, but fruits still benefit from increased carbon flux. Therefore, we see a phase during which the plant still invests in shoots, but gradually transitions to fruit-only growth by the beginning of the final interval.

Recall that Figures 2.2, 2.3, 2.4, and 2.5 represent 'snapshots' of optimal growth along the $F(T) = 900$ contour represented in Figures 2.6 and 2.7. While the general trends discussed above hold across the contour, there is a broad spectrum of optimal strategies. What is striking here is the high level of variation in solutions with initial conditions along the contour, and the fact that these initial transients are absent by the end of balanced growth. In Figure 2.8, we see that most initial conditions along the contour correspond to tightly clustered terminal conditions. In Figures 2.9 and 2.10, we again see the high degree of variation in the growth and resource allocation strategies employed in the initial stages, and the relatively little variation in both growth patterns and allocation strategies after the completion of balanced growth. In some sense, then, we might think of optimal growth under this model as being an equalizing agent that reduces initial variance in a population.

CHAPTER 3

SECOND MODEL - CARBON/NITROGEN FRUITS

3.1 Introduction

The second model for resource allocation in annual plants we will discuss is an extension of the first model, discussed in Chapter 2, to the case where the fruits require both nitrogen and carbon. This is a particularly important addition because, as we pointed out in Section 2.1, many annual plants do not pack seeds in large carbon-rich fruits, so this extension allows the model to encompass a much broader class of plants.

This additional level of complexity makes the resulting optimal control problem unwieldy, so after introducing the model in Section 3.2 we will use the results of our first model to make an ansatz about the structure of the solution to this second model, in a particular case regarding the ordering of C:N ratios between the three organs. Letting ν_F be the C:N ratio in fruits, we will assume that $\nu_F < \nu_S < \nu_R$ (see [4, 10]). This is based on the assumption that shoots and roots each need a higher proportion of the imported resource, and fruits require more nitrogen per unit carbon than either of the other two organs.

This chapter will follow the same structure as in Chapter 2. We will begin with a description of the model and optimal control problem in Section 3.2, followed by two sections on mathematical results, Sections 3.3 and 3.4. Next, we will go through

the numerical scheme for this model in Section 3.5, and discuss how it differs from the first model. In Section 3.6 we will present numerical results based on simulations, ending with a discussion in Section 3.7.

3.2 A Description of the Model

In this section we will describe our second model for resource allocation in annual plants. This model is in many ways similar to the first model so we will frequently reference Chapter 2 rather than reiterate details common to both models.

3.2.1 Model Setup

We consider an extension of the model discussed in Chapter 2 in which we replace the assumption that fruits are carbon-only with the assumption that fruits require both carbon and nitrogen. Furthermore, as with the shoots and roots, we use the parallel complementary synthesizing unit function given by 2.10 to provide the rate of fruit production given carbon and nitrogen fluxes to the fruits. This requires introducing an additional control, $u_{0N}(t)$, the fraction of nitrogen allocated to the fruits at time t, as well as the fixed C:N ratio in fruits, ν_F. Aside from these additions, this model is otherwise identical to the model discussed in Chapter 2. The differential equations for this model are

$$\frac{dS}{dt} = g(u_{1C}C, \nu_S u_{1N} N), \quad S(0) = S_0 \tag{3.1}$$

$$\frac{dR}{dt} = g(u_{2C}C, \nu_R u_{2N} N), \quad R(0) = R_0 \tag{3.2}$$

$$\frac{dF}{dt} = g(u_{0C}C, \nu_F u_{0N} N), \quad F(0) = 0 \tag{3.3}$$

and the model is shown schematically in Figure 3.1.

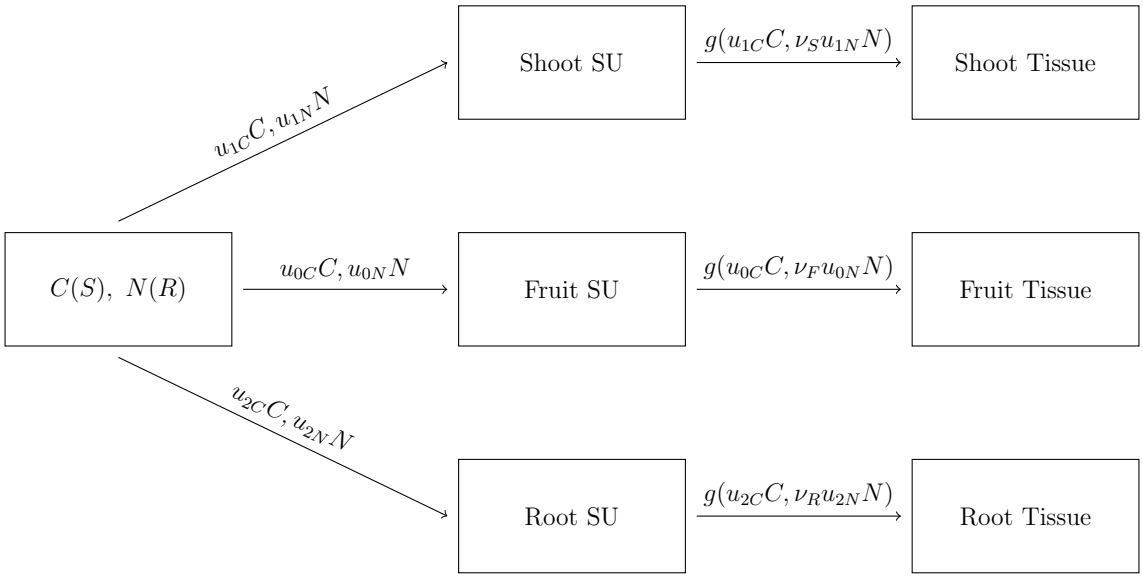

Figure 3.1: Model Schematic

3.2.2 Optimal Control Problem

As with the first model, the goal here is to find the growth trajectory that maximizes fruit biomass at time T. Following the same procedure employed in Section 2.2.3, we obtain the following optimal control problem.

$$\begin{aligned} & \max_{\vec{u}} \int_0^T g(u_{0C}C, \nu_F u_{0N}N)\, dt \\ \text{subject to:} \quad & u_{iC} \geq 0,\ u_{iN} \geq 0 \text{ for } i = 0, 1, 2 \\ & u_{0C} + u_{1C} + u_{2C} = 1 = u_{0N} + u_{1N} + u_{2N} \\ & \frac{dS}{dt} = g(u_{1C}C, \nu_S u_{1N}N), \quad S(0) = S_0 \\ & \frac{dR}{dt} = g(u_{2C}C, \nu_R u_{2N}N), \quad R(0) = R_0 \end{aligned} \tag{3.4}$$

Note that the only real difference in the formulations between (2.17) and (3.4) is that the integrand is different and we have the addition of u_{0N} in the algebraic constraints.

3.2.3 Necessary Conditions

As with the first model, we will use a set of necessary conditions to describe the dynamics within each phase of the solution. Since the new model essentially consists of two 3-control problems embedded in (3.4), we again rely on the derivation in Appendix A to obtain the necessary conditions. We begin by forming a Hamiltonian with two piecewise differentiable adjoints, $\lambda_1(t)$ and $\lambda_2(t)$:

$$H = g(u_{0C}C, \nu_F u_{0N} N) + \lambda_1 g(u_{1C}C, \nu_S u_{1N} N) + \lambda_2 g(u_{2C}C, \nu_R u_{2N} N). \tag{3.5}$$

The necessary conditions for optimality are as follows:

$$\begin{cases} u_{iC} = 0 & \text{if} \quad \frac{\partial H}{\partial u_{iC}} < \frac{\partial H}{\partial u_{jC}}, \text{ for all } j \neq i \\ u_{iC} = 1 & \text{if} \quad \frac{\partial H}{\partial u_{iC}} > \frac{\partial H}{\partial u_{jC}}, \text{ for all } j \neq i \\ 0 \leq u_{iC} \leq 1 & \text{if} \quad \frac{\partial H}{\partial u_{iC}} = \frac{\partial H}{\partial u_{jC}}, \text{ for any } j \neq i \\ u_{iN} = 0 & \text{if} \quad \frac{\partial H}{\partial u_{iN}} < \frac{\partial H}{\partial u_{jN}}, \text{ for all } j \neq i \\ u_{iN} = 1 & \text{if} \quad \frac{\partial H}{\partial u_{iN}} > \frac{\partial H}{\partial u_{jN}}, \text{ for all } j \neq i \\ 0 \leq u_{iN} \leq 1 & \text{if} \quad \frac{\partial H}{\partial u_{iN}} = \frac{\partial H}{\partial u_{jN}}, \text{ for any } j \neq i \end{cases} \tag{3.6}$$

Note that these are essentially the same necessary conditions we saw in Section 2.2.4, with the exception that since there are now three nitrogen controls, the necessary conditions for the nitrogen controls take on the same form as those for the carbon controls.

As in Section 2.2.4, we change variables to simplify the differential equations, and

ultimately the necessary conditions. We make the substitutions

$$z_{0C} = \frac{u_{0C}C}{\nu_F u_{0N} N} \tag{3.7}$$

$$z_{1C} = \frac{u_{1C}C}{\nu_S u_{1N} N} \tag{3.8}$$

$$z_{2C} = \frac{u_{2C}C}{\nu_R u_{2N} N} \tag{3.9}$$

and use (2.9) to rewrite (3.1), (3.2), and (3.3) as

$$\frac{dS}{dt} = \nu_S u_{1N} N G(z_{1C}), \quad S(0) = S_0 \tag{3.10}$$

$$\frac{dR}{dt} = \nu_R u_{2N} N G(z_{2C}), \quad R(0) = R_0 \tag{3.11}$$

$$\frac{dF}{dt} = \nu_F u_{0N} N G(z_{0C}), \quad F(0) = 0 \tag{3.12}$$

Therefore, the Hamiltonian (3.5) can be rewritten as

$$H = \nu_F u_{0N} N G(z_{0C}) + \lambda_1 \nu_S u_{1N} N G(z_{1C}) + \lambda_2 \nu_R u_{2N} N G(z_{2C}). \tag{3.13}$$

Alternatively, we can make the substitutions

$$z_{0N} = \frac{\nu_F u_{0N} N}{u_{0C} C} \tag{3.14}$$

$$z_{2N} = \frac{\nu_R u_{2N} N}{u_{2C} C} \tag{3.15}$$

$$z_{1N} = \frac{\nu_S u_{1N} N}{u_{1C} C} \tag{3.16}$$

and rewrite (3.1), (3.2), and (3.3) as

$$\frac{dS}{dt} = u_{1C}CG(z_{1N}), \quad S(0) = S_0 \tag{3.17}$$
$$\frac{dR}{dt} = u_{2C}CG(z_{2N}), \quad R(0) = R_0 \tag{3.18}$$
$$\frac{dF}{dt} = u_{0C}CG(z_{0N}), \quad F(0) = 0 \tag{3.19}$$

The Hamiltonian can then be rewritten as

$$H = u_{0C}CG(z_{0N}) + \lambda_1 u_{1C}CG(z_{1N}) + \lambda_2 u_{2C}CG(z_{2N}). \tag{3.20}$$

Using (3.13) and (3.20), we can compute the following partial derivatives:

$$\frac{\partial H}{\partial u_{0C}} = \frac{\partial}{\partial u_{0C}}\nu_F u_{0N}NG(z_{0C}) = \nu_F u_{0N}NG'(z_{0C})\frac{\partial}{\partial u_{0C}}z_{0C} = CG'(z_{0C}) \tag{3.21}$$
$$\frac{\partial H}{\partial u_{1C}} = \frac{\partial}{\partial u_{1C}}\lambda_1\nu_S u_{1N}NG(z_{1C}) = \lambda_1\nu_S u_{1N}NG'(z_{1C})\frac{\partial}{\partial u_{1C}}z_{1C} = \lambda_1 CG'(z_{1C}) \tag{3.22}$$
$$\frac{\partial H}{\partial u_{2C}} = \frac{\partial}{\partial u_{2C}}\lambda_2\nu_R u_{2N}NG(z_{2C}) = \lambda_2\nu_R u_{2N}NG'(z_{2C})\frac{\partial}{\partial u_{2C}}z_{2C} = \lambda_2 CG'(z_{2C}) \tag{3.23}$$
$$\frac{\partial H}{\partial u_{0N}} = \frac{\partial}{\partial u_{0N}}u_{0C}CG(z_{0N}) = u_{0C}CG'(z_{0N})\frac{\partial}{\partial u_{0N}}z_{0N} = \nu_F NG'(z_{0N}) \tag{3.24}$$
$$\frac{\partial H}{\partial u_{1N}} = \frac{\partial}{\partial u_{1N}}\lambda_1 u_{1C}CG(z_{1N}) = \lambda_1 u_{1C}CG'(z_{1N})\frac{\partial}{\partial u_{1N}}z_{1N} = \lambda_1\nu_S NG'(z_{1N}) \tag{3.25}$$
$$\frac{\partial H}{\partial u_{2N}} = \frac{\partial}{\partial u_{2N}}\lambda_2 u_{2C}CG(z_{2N}) = \lambda_2 u_{2C}CG'(z_{2N})\frac{\partial}{\partial u_{2N}}z_{2N} = \lambda_2\nu_R NG'(z_{2N}). \tag{3.26}$$

Lastly, recall that by Appendix A we have a characterization of the adjoints in terms of the Hamiltonian:

$$\lambda_1' = -\frac{\partial H}{\partial S}, \quad \lambda_1(T) = 0 \tag{3.27}$$
$$\lambda_2' = -\frac{\partial H}{\partial R}, \quad \lambda_2(T) = 0. \tag{3.28}$$

Making use of (3.13) and (3.20), we can express (3.27) and (3.28) as follows.

$$\begin{aligned}
\lambda_1' &= -\frac{\partial H}{\partial S} \\
&= -\frac{\partial}{\partial S}\left[\nu_F u_{0N} N G(z_{0C}) + \lambda_1 \nu_S u_{1N} N G(z_{1C}) + \lambda_2 \nu_R u_{2N} N G(z_{2C})\right] \\
&= -\left[\nu_F u_{0N} N G'(z_{0C})\frac{\partial z_{0C}}{\partial S} + \lambda_1 \nu_S u_{1N} N G'(z_{1C})\frac{\partial z_{1C}}{\partial S} + \lambda_2 \nu_R u_{2N} N G'(z_{2C})\frac{\partial z_{2C}}{\partial S}\right] \\
&= -\left[u_{0C} C_S G'(z_{0C}) + \lambda_1 u_{1C} C_S G'(z_{1C}) + \lambda_2 u_{2C} C_S G'(z_{2C})\right] \\
&= -C_S\left[u_{0C} G'(z_{0C}) + \lambda_1 u_{1C} G'(z_{1C}) + \lambda_2 u_{2C} G'(z_{2C})\right] \qquad (3.29)
\end{aligned}$$

$$\begin{aligned}
\lambda_2' &= -\frac{\partial H}{\partial R} \\
&= -\frac{\partial}{\partial R}\left[u_{0C} C G(z_{0N}) + \lambda_1 u_{1C} C G(z_{1N}) + \lambda_2 u_{2C} C G(z_{2N})\right] \\
&= -\left[u_{0C} C G'(z_{0N})\frac{\partial z_{0N}}{\partial R} + \lambda_1 u_{1C} C G'(z_{1N})\frac{\partial z_{1N}}{\partial R} + \lambda_2 u_{2C} C G'(z_{2N})\frac{\partial z_{2N}}{\partial R}\right] \\
&= -\left[\nu_F u_{0N} N_R G'(z_{0N}) + \lambda_1 \nu_S u_{1N} N_R G'(z_{1N}) + \lambda_2 \nu_R u_{2N} N_R G'(z_{2N})\right] \\
&= -N_R\left[\nu_F u_{0N} G'(z_{0N}) + \lambda_1 \nu_S u_{1N} G'(z_{1N}) + \lambda_2 \nu_R u_{2N} G'(z_{2N})\right] \qquad (3.30)
\end{aligned}$$

3.3 Four-Phase Structure

The mathematical results for this model will be broadly split into two sections. In the first section we will prove that, as with the first, simpler model, we again see a phase of fruit-only growth at the end of the growing season. The second section of results will concern results we have been able to obtain with the help of Ansatz 3.1 below. We are going to focus specifically on the case where $\nu_F < \nu_S < \nu_R$, which means that roots require the most units of carbon per unit of nitrogen, and fruits require the most nitrogen per unit of carbon. We expect, though do not prove here, that the solution in this case consists of the same four-stage structure seen in the first model

(Section 2.3), with the exception that the penultimate interval consists of root/fruit growth rather than shoot/fruit growth, because here nitrogen is more important to fruit production than carbon. To make the problem tenable we make the following ansatz:

Ansatz 3.1. *When the C:N ratios are ordered $\nu_F < \nu_S < \nu_R$, the optimal trajectory consists of growth stages in the following order starting at either stage 0 or 1 depending on whether the plant is initially balanced:*

0. Initial Phase: shoot-only or root-only growth

1. Balanced Growth: mixed shoot/root growth

2. Penultimate Interval: mixed root/fruit growth

3. Final Interval: fruit-only growth.

3.3.1 Final Interval

In this section we will prove that, as with the first model, the solution to the the optimal control problem (3.4) for this model includes a final interval of fruit-only growth. First, note that by (3.27) and (3.28) we have that $\lambda_1(T) = 0 = \lambda_2(T)$, and since G is bounded we have by (3.22), (3.23), (3.25), and (3.26) that

$$\frac{\partial H}{\partial u_{1C}}(T) = 0 \tag{3.31}$$

$$\frac{\partial H}{\partial u_{2C}}(T) = 0 \tag{3.32}$$

$$\frac{\partial H}{\partial u_{1N}}(T) = 0 \tag{3.33}$$

$$\frac{\partial H}{\partial u_{2N}}(T) = 0. \tag{3.34}$$

Furthermore, because we assume that $C^* > 0$ and $N^* > 0$ and since $G' > 0$ we have at $t = T$ that

$$\frac{\partial H}{\partial u_{0C}}(t) > \frac{\partial H}{\partial u_{1C}}(T), \frac{\partial H}{\partial u_{2C}}(T) \tag{3.35}$$

$$\frac{\partial H}{\partial u_{0N}}(t) > \frac{\partial H}{\partial u_{1N}}(T), \frac{\partial H}{\partial u_{2N}}(T). \tag{3.36}$$

By (3.6) then we have that $u_{0C}(T) = 1 = u_{0N}(T)$, and therefore

$$\frac{\partial H}{\partial u_{0C}}(T) = C^* G' \left(\frac{C^*}{\nu_F N^*} \right) \tag{3.37}$$

$$\frac{\partial H}{\partial u_{0N}}(T) = \nu_F N^* G' \left(\frac{\nu_F N^*}{C^*} \right). \tag{3.38}$$

Now, because both λ_1 and λ_2 are continuous, and G' is bounded between 0 and 1, there exists an $\varepsilon > 0$ such that (3.35) and (3.36) still hold for all t in $(T - \varepsilon, T]$. By (3.6) this means there exists an interval of fruit-only growth at the end of the growing season.

Furthermore, by (3.13) and (3.20), we have that during this stage

$$H = \nu_F N^* G \left(\frac{C^*}{\nu_F N^*} \right) = C^* G \left(\frac{\nu_F N^*}{C^*} \right). \tag{3.39}$$

As with the first model, the optimal control problem (3.4) for this model is autonomous, and so the Hamiltonian is constant along the optimal trajectory. Therefore, (3.39) must hold for all $t \in [0, T]$.

3.4 Phase Dynamics and Transition

As we did in Section 2.4 for (2.17), we will again present the basic equations governing the dynamics in each phase of the solution to (3.4) as assumed in Ansatz 3.1, as well

as show that z_{0C} and z_{1C} are continuous between any two phases in which they are defined, and z_{2C} is continuous between a phase of root-only growth and balanced growth. It is still an open question whether z_{2C} is necessarily continuous at the boundary between balanced growth and the penultimate intervals, though numerical evidence seems to suggest that this is the case. Recall that these quantities z_{0C}, z_{1C}, and z_{2C}, represent the ratios carbon flux to nitrogen flux in units of carbon to the fruits, shoots, and roots, respectively, and so their continuity means that the amount of carbon allocated to an organ per unit of nitrogen varies continuously so long as the organ is growing. The definitions are revisited below.

$$z_{0C} = \frac{u_{0C}C}{\nu_F u_{0N} N} \qquad \text{((3.7) revisited)}$$

$$z_{1C} = \frac{u_{1C}C}{\nu_S u_{1N} N} \qquad \text{((3.8) revisited)}$$

$$z_{2C} = \frac{u_{2C}C}{\nu_R u_{2N} N} \qquad \text{((3.9) revisited)}$$

As in Section 2.4, we will discuss the phases in chronological order before discussing the transition in chronological order. We will begin with the initial stage of either shoot or root growth.

3.4.1 Initial Phase: Shoot-Only Growth

During shoot-only growth we have

$$u_{0C} = 0, \quad u_{1C} = 1, \quad u_{2C} = 0, \quad u_{0N} = 0, \quad u_{1N} = 1, \quad u_{2N} = 0 \qquad (3.40)$$

and only S, λ_1, and λ_2 are changing. The differential equations in time for these three during this stage are given by

$$S' = \nu_S N G\left(z_{1C}\right) \tag{3.41}$$

$$\lambda_1' = -\lambda_1 C_S G'\left(z_{1C}\right) \tag{3.42}$$

$$\lambda_2' = -\lambda_1 N_R \nu_S G_2\left(z_{1C}\right) \tag{3.43}$$

where here

$$z_{1C} = \frac{C}{\nu_S N}. \tag{3.44}$$

Note also that in this stage both N and N_R are constant because R is constant. Furthermore, because we know by (3.39) that $H = C^* G\left(\frac{\nu_F N^*}{C^*}\right)$ we can rewrite the Hamiltonian (3.13) during this phase as

$$C^* G\left(\frac{\nu_F N^*}{C^*}\right) = \lambda_1 \nu_S N G(z_{1C}) \tag{3.45}$$

and so by solving (3.45) for λ_1 and then making use of (2.14) we have two additional expressions for λ_1 during this phase:

$$\lambda_1 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_S N G(z_{1C})} \tag{3.46}$$

$$\lambda_1 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_S N G_2(z_{1C}) + C G'(z_{1C})}. \tag{3.47}$$

3.4.2 Initial Phase: Root-Only Growth

During root-only growth we have

$$u_{0C} = 0, \quad u_{1C} = 0, \quad u_{2C} = 1, \quad u_{0N} = 0, \quad u_{1N} = 0, \quad u_{2N} = 1 \tag{3.48}$$

and only R, λ_1, and λ_2 are changing. The differential equations in time for these three during this stage are given by

$$R' = \nu_R N G(z_{2C}) \tag{3.49}$$

$$\lambda_1' = -\lambda_2 C_S G'(z_{2C}) \tag{3.50}$$

$$\lambda_2' = -\lambda_2 N_R \nu_R G_2\left(z_{2C}\right) \tag{3.51}$$

where here

$$z_{2C} = \frac{C}{\nu_R N}. \tag{3.52}$$

Note also that in this stage both C and C_S are constant because S is constant. Furthermore, because we know by (3.39) that $H = C^* G\left(\frac{\nu_F N^*}{C^*}\right)$ we can rewrite the Hamiltonian (3.13) during this phase as

$$C^* G\left(\frac{\nu_F N^*}{C^*}\right) = \lambda_2 \nu_R N G(z_{2C}) \tag{3.53}$$

and so by solving (3.53) for λ_2 and then making use of (2.14) we have two additional expressions for λ_2 during this phase:

$$\lambda_2 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_R N G(z_{2C})} \tag{3.54}$$

$$\lambda_2 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_R N G_2(z_{2C}) + C G'(z_{2C})}. \tag{3.55}$$

3.4.3 Balanced Growth - Shoot/Root Growth

During balanced growth we have

$$u_{0C} = 0, \quad 0 \le u_{1C} \le 1, \quad u_{2C} = 1 - u_{1C}, \quad u_{0N} = 0, \quad 0 \le u_{1N} \le 1, \quad u_{2N} = 1 - u_{1N} \tag{3.56}$$

and so by (3.6) we have that

$$\lambda_1 G'(z_{1C}) = \lambda_2 G'(z_{2C}) \tag{3.57}$$

and

$$\lambda_1 \nu_S G_2(z_{1C}) = \lambda_2 \nu_R G_2(z_{2C}). \tag{3.58}$$

During this stage S, R, λ_1, and λ_2 are changing. The differential equations in time for these four during this phase are given by

$$S' = \nu_S u_{1N} N G(z_{1C}) \tag{3.59}$$

$$R' = \nu_R (1 - u_{1N}) N G(z_{2C}) \tag{3.60}$$

$$\lambda_1' = -C_S \lambda_1 G'(z_{1C}) = -C_S \lambda_2 G'(z_{2C}) \tag{3.61}$$

$$\lambda_2' = -N_R \lambda_1 \nu_S G_2(z_{1C}) = -N_R \lambda_2 \nu_R G_2(z_{2C}). \tag{3.62}$$

Furthermore, taking advantage of the fact that (3.39) gives us $H = C^* G\left(\frac{\nu_F N^*}{C^*}\right)$, we can rewrite the Hamiltonian (3.13) during this phase as

$$C^* G\left(\frac{\nu_F N^*}{C^*}\right) = \lambda_1 \nu_S u_{1N} N G(z_{1C}) + \lambda_2 \nu_R u_{2N} N G(z_{2C}). \tag{3.63}$$

Noting that as (3.63) and (2.82) only differ by a constant on the left-hand side, we can use the same procedure outlined in Section 2.4.3 to obtain characterizations for λ_1 and λ_2 during the balanced growth phase:

$$\lambda_1 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_S N G_2(z_{1C}) + C G'(z_{1C})} \tag{3.64}$$

$$\lambda_2 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_R N G_2(z_{2C}) + C G'(z_{2C})}. \tag{3.65}$$

3.4.4 Penultimate Interval - Root/Fruit Growth

During the penultimate interval we have

$$0 \leq u_{0C} \leq 1, \quad u_{1C} = 0, \quad u_{2C} = 1 - u_{0C}, \quad 0 \leq u_{0N} \leq 1, \quad u_{1N} = 0, \quad u_{2N} = 1 - u_{0N} 4 \tag{3.66}$$

and so by (3.6) we have that

$$G'(z_{0C}) = \lambda_2 G'(z_{2C}) \tag{3.67}$$

and

$$\nu_F G_2(z_{0C}) = \lambda_2 \nu_R G_2(z_{2C}). \tag{3.68}$$

During this interval R, F, λ_1, and λ_2 are all changing. The differential equations in time for these four during this stage are

$$R' = \nu_R (1 - u_{0N}) N G(z_{2C}) \tag{3.69}$$

$$F' = \nu_F u_{0N} N G(z_{0C}) \tag{3.70}$$

$$\lambda_1' = -C_S \lambda_2 G'(z_{2C}) = -C_S G'(z_{0C}) \tag{3.71}$$

$$\lambda_2' = -N_R \lambda_2 \nu_R G_2(z_{2C}) = -N_R \nu_F G_2(z_{0C}) \tag{3.72}$$

Furthermore, taking advantage of the fact that (3.39) gives us $H = C^* G\left(\frac{\nu_F N^*}{C^*}\right)$, we can rewrite the Hamiltonian (3.13) during this phase as

$$C^* G\left(\frac{\nu_F N^*}{C^*}\right) = \nu_F u_{0N} N G(z_{0C}) + \lambda_2 \nu_R u_{2N} N G(z_{2C}). \tag{3.73}$$

Again, by the procedure used in Section 2.4.3, we use (3.73) to obtain the following:

$$\lambda_2 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_R N G_2(z_{2C}) + C G'(z_{2C})} \tag{3.74}$$

$$1 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_F N G_2(z_{0C}) + C G'(z_{0C})}. \tag{3.75}$$

3.4.5 Final Interval - Fruit-Only Growth

During the final interval of fruit-only growth we have

$$u_{0C} = 1, \quad u_{1C} = 0, \quad u_{2C} = 0, \quad u_{0N} = 1, \quad u_{1N} = 1, \quad u_{2N} = 0 \tag{3.76}$$

and F, λ_1, and λ_2 are changing. The differential equations in time for these three during this phase are

$$F' = \nu_F N^* G(z_{0C}) \tag{3.77}$$

$$\lambda_1' = -C_S^* G'(z_{0C}) \tag{3.78}$$

$$\lambda_2' = -N_R^* \nu_F G_2(z_{0C}) \tag{3.79}$$

where here

$$z_{0C} = \frac{C^*}{\nu_F N^*}.$$

Because we also know that $\lambda_1(T) = 0 = \lambda_2(T)$, we can solve for λ_1 and λ_2 during this final interval:

$$\lambda_1 = C_S^* G'\left(\frac{C^*}{\nu_F N^*}\right)(T - t) \tag{3.80}$$

$$\lambda_2 = N_R^* \nu_F G_2\left(\frac{C^*}{\nu_F N^*}\right)(T - t). \tag{3.81}$$

3.4.6 Initial Phase to Balanced Growth Transition

We will now turn our attention to discussing the transitions between the four phases and, in a similar manner to what we did in Section 2.4, we will show that z_{0C} and z_{1C} are continuous between any two phases in which they are defined, and z_{2C} is continuous between a phase of root-only growth and balanced growth. We will, whenever possible, appeal to arguments made in previous sections to avoid repeating work we've already done.

In the same manner discussed in Section 3.4.6, we see that (3.47) and (3.64) are the same and (3.55) and (3.65) are the same. Therefore, we can again consider a 'generalized' versions of these equations:

$$\lambda = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu N G_2(z) + C G'(z)} \tag{3.82}$$

where here (λ, ν, z) is either $(\lambda_1, \nu_S, z_{1C})$ or $(\lambda_2, \nu_R, z_{2C})$ depending on whether the initial stage is shoot-only growth or root-only growth, respectively. We will call this transition point $t = \widetilde{t}$, let $x = \lim_{t \to \widetilde{t}^+} z$, and note that $\lim_{t \to \widetilde{t}^-} z = \frac{C}{\nu N}$. Now, taking limits of (3.82) from both sides, we obtain

$$\lim_{t \to \widetilde{t}^+} \lambda(t) = \widetilde{\lambda}^+ = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu N G_2(x) + C G'(x)} \tag{3.83}$$

$$\lim_{t \to \widetilde{t}^-} \lambda(t) = \widetilde{\lambda}^- = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu N G_2\left(\frac{C}{\nu N}\right) + C G'\left(\frac{C}{\nu N}\right)} \tag{3.84}$$

As λ is necessarily continuous, we have that $\widetilde{\lambda}^- = \widetilde{\lambda}^+$, which gives us

$$G_2\left(\frac{C}{\nu N}\right) + \frac{C}{\nu N} G'\left(\frac{C}{\nu N}\right) = G_2(x) + \frac{C}{\nu N} G'(x). \tag{3.85}$$

Note that this is exactly (2.104), and so the argument proceeds identically to that in

Section 2.4.6 and we arrive at the conclusion that z_{1C} is continuous between an initial phase of shoot-only growth and balanced growth, and z_{2C} is continuous between an initial phase of root-only growth and balanced growth. As it is still unknown whether z_{2C} is always continuous between the balanced growth and penultimate intervals, we will proceed to the transition from the penultimate to final interval.

3.4.7 Penultimate Interval to Final Interval Transition

In this section, we will show that z_{0C} is continuous at the boundary between the penultimate and final intervals. This argument will proceed in a similar fashion to the one previously used to show that either z_{1C} or z_{2C} is continuous between the initial phase and the balanced growth phase. Recall that during the penultimate interval we have

$$1 = \frac{C^* G\left(\frac{\nu_F N^*}{C^*}\right)}{\nu_F N G_2(z_{0C}) + C G'(z_{0C})}. \qquad \text{(3.75 revisited)}$$

Letting $\lim_{t \to t^{*-}} z_{0C} = z_{0C}^-$, we have that as $t \to t^*$, (3.75) becomes

$$\frac{C^*}{\nu_F N^*} G\left(\frac{\nu_F N^*}{C^*}\right) = G_2\left(z_{0C}^-\right) + \frac{C^*}{\nu_F N^*} G'\left(z_{0C}^-\right). \tag{3.86}$$

Note that by (2.9) we can rewrite the left-hand side of (3.86) to obtain

$$G\left(\frac{C^*}{\nu_F N^*}\right) = G_2\left(z_{0C}^-\right) + \frac{C^*}{\nu_F N^*} G'\left(z_{0C}^-\right), \tag{3.87}$$

which by (2.14) becomes

$$G_2\left(\frac{C^*}{\nu_F N^*}\right) + \frac{C^*}{\nu_F N^*} G'\left(\frac{C^*}{\nu_F N^*}\right) = G_2\left(z_{0C}^-\right) + \frac{C^*}{\nu_F N^*} G'\left(z_{0C}^-\right). \tag{3.88}$$

At this point we have again reached (2.104), and so the argument proceeds identically to that in Section 2.4.6, and we arrive at the conclusion that z_{0C} is continuous between the penultimate and final intervals.

3.5 Numerical Scheme

The overall structure of the numerical scheme is similar to that discussed in Section 2.5, in the sense that solving the control problem (3.4) numerically follows the same two-step process. We first develop a numerical scheme for the map

$$(S^*, R^*) \mapsto (S, R, F, \vec{u}, \lambda_1, \lambda_2),$$

and then, in conjunction with MATLAB's built-in nonlinear equation solver `fsolve`, use it to obtain the map

$$(S_0, R_0) \mapsto (S^*, R^*).$$

As before, we ultimately obtain the map

$$(S_0, R_0) \mapsto (S, R, F, \vec{u}, \lambda_1, \lambda_2).$$

In this section, as in Section 2.6, we will focus on the first map, that is solving the problem backward in time for given (S^*, R^*). Additionally, we will again keep the discussion detailed enough to convey how the phases are simulated and pieced together, but general enough so as to avoid going into the fine details of the scheme. The actual MATLAB code is included in Appendix C.2.

While the initial phase and balanced growth phase are nearly the same in both models, the penultimate interval and final interval in the second model are much

different from the first. We will begin with locating t^*, the boundary between the penultimate interval and the final interval, and then proceed along a similar trajectory to that laid out in Section 2.5, constructing the solution to (3.4) during the penultimate interval, final interval, balanced growth phase, and initial phase, respectively.

3.5.1 Locating the Penultimate Interval - Final Interval Boundary

Because $\frac{\partial H}{\partial u_{0C}}$ and $\frac{\partial H}{\partial u_{0N}}$ are the only partial derivatives of H with respect to the controls defined in the final interval, there is more involved in locating t^* than there was with the first model. We will, however, be able to use the fact that z_{0C} is continuous at this boundary to simplify the procedure for locating t^*. First, let $\lim_{t\to t^{*-}} z_{2C} = z_{2C}^-$. Now, because z_{0C} is continuous at $t = t^*$, (3.67) and (3.68) become

$$G'\left(\frac{C^*}{\nu_F N^*}\right) = \lambda_2(t^*)G'(z_{2C}^-) \tag{3.89}$$

$$\nu_F G_2\left(\frac{C^*}{\nu_F N^*}\right) = \lambda_2(t^*)\nu_R G_2(z_{2C}^-). \tag{3.90}$$

Furthermore, because λ_2 is continuous, we can rewrite (3.90) using (3.81) to get

$$\nu_F G_2\left(\frac{C^*}{\nu_F N^*}\right) = N_R^*\nu_F G_2\left(\frac{C^*}{\nu_F N^*}\right)\nu_R G_2(z_{2C}^-)(T - t^*). \tag{3.91}$$

Solving (3.91) for t^* gives us

$$t^* = T - \frac{1}{\nu_R N_R^* G_2(z_{2C}^-)}. \tag{3.92}$$

In order to use (3.92) we first need to calculate z_{2C}^-. To this end, we can use (3.89) and (3.90) to obtain

$$\frac{\nu_R G'\left(\frac{C^*}{\nu_F N^*}\right)}{\nu_F G_2\left(\frac{C^*}{\nu_F N^*}\right)} = \frac{G'(z_{2C}^-)}{G_2(z_{2C}^-)}. \tag{3.93}$$

Now, by (2.11) and (2.13), we have that

$$\frac{G'(z)}{G_2(z)} = \frac{1+2z}{z^3(2+z)},$$

which for $z \geq 0$ is invertible by the proof of Lemma 2.8. This means we can use (3.93) to solve for z_{2C}^-.

To numerically solve for t^*, then, we begin by using MATLAB's built-in nonlinear equation solver `fsolve` to solve (3.93) for z_{2C}^-. Next, we evaluate the right-hand side of (3.92) to obtain t^*. At this point we know when the penultimate interval ends, and we can proceed to the penultimate interval.

3.5.2 Penultimate Interval

Recall that in Section 2.5.1 we solved differential equations in z_{1C} rather than t because of a singularity in $\frac{dz_{1C}}{dt}$ as $t \to t^*$. In this model, due to the presence of z_{0C}, the penultimate interval is more similar to the balanced growth phase than to the penultimate interval in the first model. So, for the penultimate interval in this model, we will employ a strategy that more closely resembles that used in Section 2.5.5 (balanced growth) than the one used in Section 2.5.3 (penultimate interval).

Starting with equations for z_{0C} and z_{2C} during the penultimate interval

$$z_{0C} = \frac{u_{0C}C^*}{\nu_F u_{0N} N} \tag{3.94}$$

$$z_{2C} = \frac{(1-u_{0C})C^*}{\nu_R(1-u_{0N})N} \tag{3.95}$$

we can solve for u_{0N} and u_{0C}:

$$u_{0N} = \frac{\frac{C^*}{N} - \nu_R z_{2C}}{\nu_F z_{0C} - \nu_R z_{2C}} \tag{3.96}$$

$$u_{0C} = \frac{\nu_F z_{0C} N u_{0N}}{C^*}. \tag{3.97}$$

As in Section 2.5.5, we derive differential equations for z_{0C} and z_{2C} and ultimately use (3.96) and (3.97) to obtain the controls. The differential equations in time for z_{0C} and z_{2C} are given below, the derivation is included in Appendix B.3.

$$\frac{dz_2}{dt} = \frac{N_R \nu_R \left[G_2(z_2)\right]^2 G'(z_2) G(z_0)}{G''(z_2)\left[z_0 G'(z_0) G_2(z_2) - z_2 G'(z_2) G_2(z_0)\right]} \tag{3.98}$$

$$\frac{dz_0}{dt} = \frac{G_2(z_0)}{z_0 G''(z_0) G_2(z_2)} \left[z_2 G''(z_2) \frac{dz_2}{dt} + N_R \nu_R \left[G_2(z_2)\right]^2 \right] \tag{3.99}$$

We use RK4 to numerically solve (3.69), (3.71), (3.72), (3.98), and (3.99) backward in time, using (3.96) and (3.97) to eliminate the controls from the differential equations. The differential equations for z_{0C} and z_{2C} are initialized at $t = t^*$ using the fact that z_{0C} is constant during the final interval, and the value of z_{2C} found in the process of finding t^*, as discussed in Section 3.5.1. We initialize λ_1 and λ_2 at t^* by evaluating (3.80) and (3.81), and use R^* to initialize the differential equation for R. Lastly, then, we use (3.96), (3.97), and (3.66) to update the controls. We continue solving backward in time until the controls are no longer bounded in $[0, 1]$. This provides the earliest possible transition point between the balanced growth phase and the penultimate interval.

3.5.3 Locating the Start of the Penultimate Interval

We use a very similar approach to finding $\hat{t}$ to that we used in Section 2.5.2. Broadly speaking, we use a binary search on the current RK4 integration mesh to find a small

interval about which a certain condition is met, simulate the penultimate interval dynamics again on that interval with a finer mesh, and repeat the process until the condition is met to a within a specified tolerance.

We begin with the largest possible interval containing the penultimate interval, starting with the point at which the controls become unbounded and ending at t^*, and use a binary search to identify a small interval about which the following condition is satisfied:

$$H = C^* G\left(\frac{\nu_F N^*}{C^*}\right) = \lambda_1 \left[\nu_S N G_2(z_{1C}) + C G'(z_{1C})\right]. \tag{3.100}$$

Note that (3.100) is derived from (3.64). Since z_{1C} is not defined in the penultimate interval we will need to calculate it as a left-hand limit coming from balanced growth. Since we have not yet proven that z_{2C} is continuous at this boundary, we will use MATLAB's built-in nonlinear equation solver `fsolve` to solve (3.57) and (3.58) simultaneously for z_{1C} and z_{2C} during balanced growth at every step in the binary search. These equations are revisited below:

$$\lambda_1 G'(z_{1C}) = \lambda_2 G'(z_{2C}) \qquad \text{((3.57) revisited)}$$

$$\lambda_1 \nu_S G_2(z_{1C}) = \lambda_2 \nu_R G_2(z_{2C}). \qquad \text{((3.58) revisited)}$$

Once we have found a suitably small interval about which (3.100) is met, we use RK4 to simulate the penultimate interval dynamics on this interval with a finer mesh, and repeat the process until we have found a point at which (3.100) is met to within some tolerance. This gives us $\hat{t}$, the start of the penultimate interval.

3.5.4 Fruits - Penultimate Interval

Now that we have $\hat{t}$, we can solve (3.70) forward in time using RK4. This completes the numerical solution to (3.4) during the penultimate interval.

3.5.5 Final Interval

As we did in Section 2.5.4, we next take advantage of the fact that, by the process described in Section 3.5.4, we now know the value of F at the beginning of the final interval. So, we simulate the dynamics in the final interval before continuing backward to balanced growth. During the final interval the controls are constant by (3.76), S and R are constant, and λ_1 and λ_2 is given by (3.80) and (3.81). Lastly, using the value of $F(t^*)$ obtained as described in Section 3.5.4 and the fact that F' is constant during the final interval by (3.77), we have that

$$F(t) = F(t^*) + \nu_F N^* G\left(\frac{C^*}{\nu_F N^*}\right) \cdot (t - t^*). \tag{3.101}$$

Therefore, as in Section 2.5.4, this portion of the numerical scheme consists only of defining the states, adjoints, and controls, as defined above, between times t^* and T.

3.5.6 Balanced Growth

Note that the only difference between the balanced growth phases of each model is the value of the Hamiltonian constant. This, however, does not directly factor into the solution to either optimal control problem (2.17) or (3.4) during balanced growth, and so here we employ the same numerical scheme outlined in Section 2.5.5.

3.5.7 Locating the Start of Balanced Growth

Recall that, as with the first model, we have by Section 3.4.6 that z_{1C} is continuous between an initial phase of shoot-only growth and balanced growth, and z_{2C} is continuous between an initial phase of root-only growth and balanced growth. This means that we can use the same procedure discussed in Section 2.5.6 to locate the start of balanced growth, $\bar{t}$. As in that section, we also note that it is possible that the plant begins in balance and skips an interval of shoot-only or root-only growth.

The only key distinction between the numerical schemes at this point is in how we determine whether the initial phase consists of shoot-only or root-only growth. Because the Hamiltonian constant is different in this model, here we use (3.46) and (3.54) and evaluate the following at $\bar{t}$:

$$\left|\lambda_1 \nu_S N G\left(\frac{C}{\nu_S N}\right) - C^* G\left(\frac{\nu_F N^*}{C^*}\right)\right| \quad \text{and} \quad \left|\lambda_2 \nu_R N G\left(\frac{C}{\nu_R N}\right) - C^* G\left(\frac{\nu_F N^*}{C^*}\right)\right|.$$

If the first is smaller then the initial phase consists of shoot growth, and if the second is smaller the initial phase consists of root growth.

3.5.8 Initial Phase

As with the balanced growth phase, the dynamics in the initial phase are the same in each model, so we use the same numerical scheme we discussed in Section 2.5.7.

3.6 Numerical Results

In this section we will present some of the main results which are apparent from numerical simulations. We will structure this section in a similar manner to how we structured Section 2.6. As in the numerical simulations conducted with the first

model, we again make the simplifying assumption that the plant does not experience self shading as it grows, and so we choose $C(S) = S$ and $N(R) = R$. We again take $T = 10$ to be the length of the growing season, and pick stoichiometric ratios $\nu_F = 1/9$, $\nu_S = 1/3$, and $\nu_R = 1$. These ratios were chosen to respect the ordering assumed in Ansatz 3.1, as well as to maintain consistency with the values of ν_S and ν_R used in Section 2.6. As with the first model, we again see four types of optimal growth patterns, examples of which are shown below for the case that $F(T) = 600$.

3.6.1 Initial Shoot Growth

Here we chose terminal conditions $S^* = 437.50$ and $R^* = 1470.18$. This results in $F(T) = 600$, $S_0 = 43.02$, and $R_0 = 497.01$. The states and controls are shown in Figure 3.2.

The simulation shown in Figure 3.2 exhibits all four stages of growth laid out in Ansatz 3.1: there is an initial phase of shoot-only growth, followed by a phase of balanced growth between shoots and roots, then a phase of mixed root/fruit growth, and finally a phase of fruit-only growth. Looking at the last two plots of Figure 3.2 we see the carbon and nitrogen allocations strategies employed. Note that following the initial phase of shoot-only growth, we see a gradual decrease in allocation to shoots and an increase of allocation to roots during the balanced growth phase. During the penultimate interval, then, we see the plant reducing allocation to roots while increasing allocation to fruits.

3.6.2 Initial Root Growth

Here we chose terminal conditions $S^* = 439.15$ and $R^* = 1472.9$. This results in $F(T) = 600$, $S_0 = 287.08$, and $R_0 = 74.49$. The states and controls are shown in Figure 3.3.

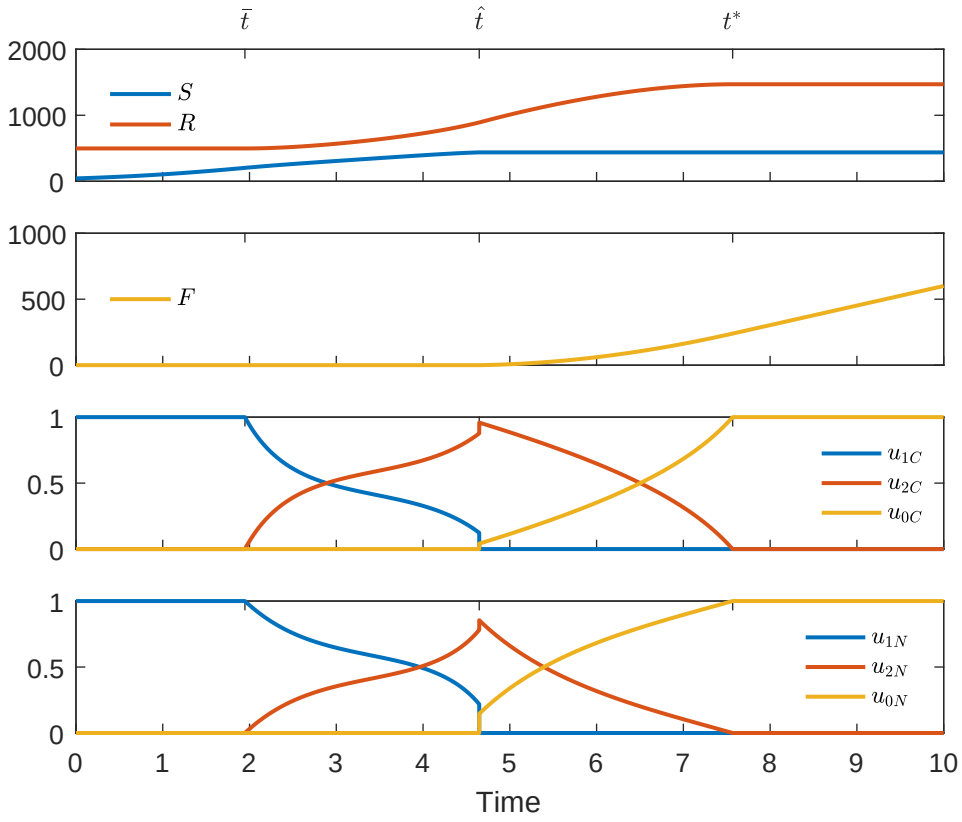

Figure 3.2: States and controls illustrating initial shoot growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

The simulation shown in Figure 3.3 again shows all four stages of growth, though in this case the initial phase of shoot growth is replaced with an initial phase of root growth. Notice that, as with the first model, we see between Figures 3.2 and 3.3 two simulations in which the terminal conditions are very close but the initial conditions are quite different.

Looking at the last two plots in Figure 3.3, we can see the resource allocation strategy. In particular, we see that during balanced growth the plant initially reduces allocation to roots while increasing allocation to shoots. Around time $t = 3.5$ the plant reverses this trend and begins to increase allocation to roots again. During the penultimate interval the allocation strategy resembles that in Figure 3.2.

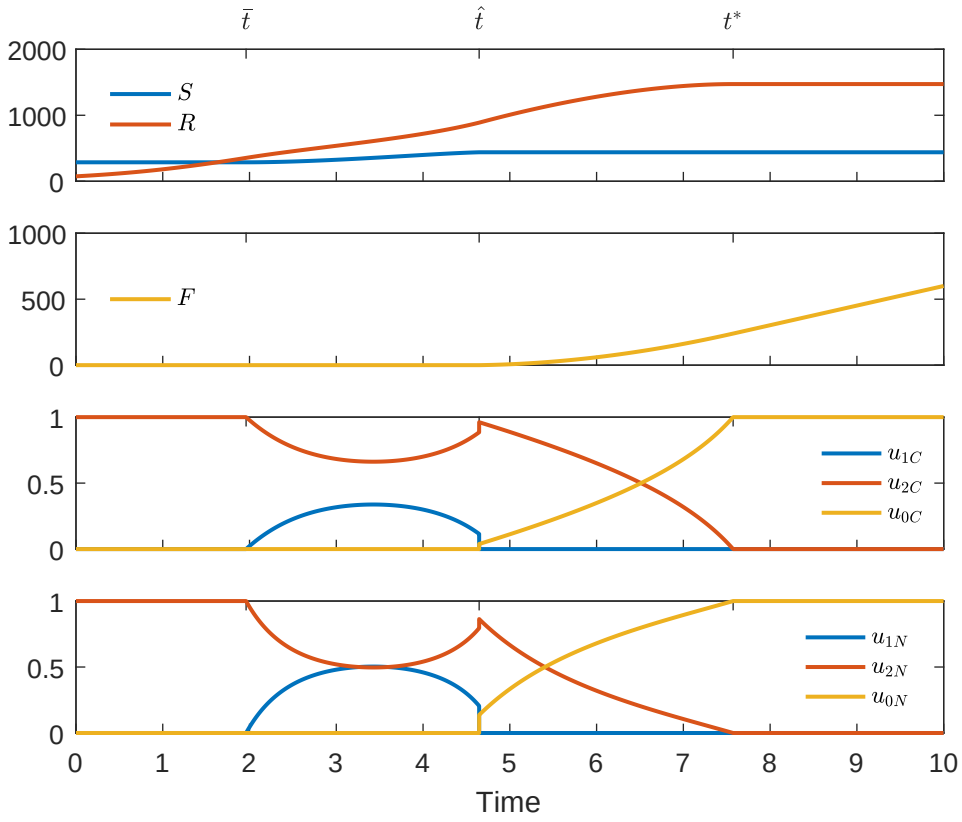

Figure 3.3: States and controls illustrating initial root growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

3.6.3 Balanced Growth First - Type S

Here we chose terminal conditions $S^* = 438.31$ and $R^* = 1471.53$. This results in $F(T) = 600$, $S_0 = 135.10$, and $R_0 = 290.12$. The states and controls are shown in Figure 3.4.

In this case we see an example of a simulation in which the plant begins in balance. Comparing Figure 3.4 with Figure 3.2, however, we see that the plants use similar allocation strategies during balanced growth in both simulations. In particular, balanced growth in each consists of a gradual decrease in allocation to shoots while simultaneously increase allocation to roots. We call this allocation pattern

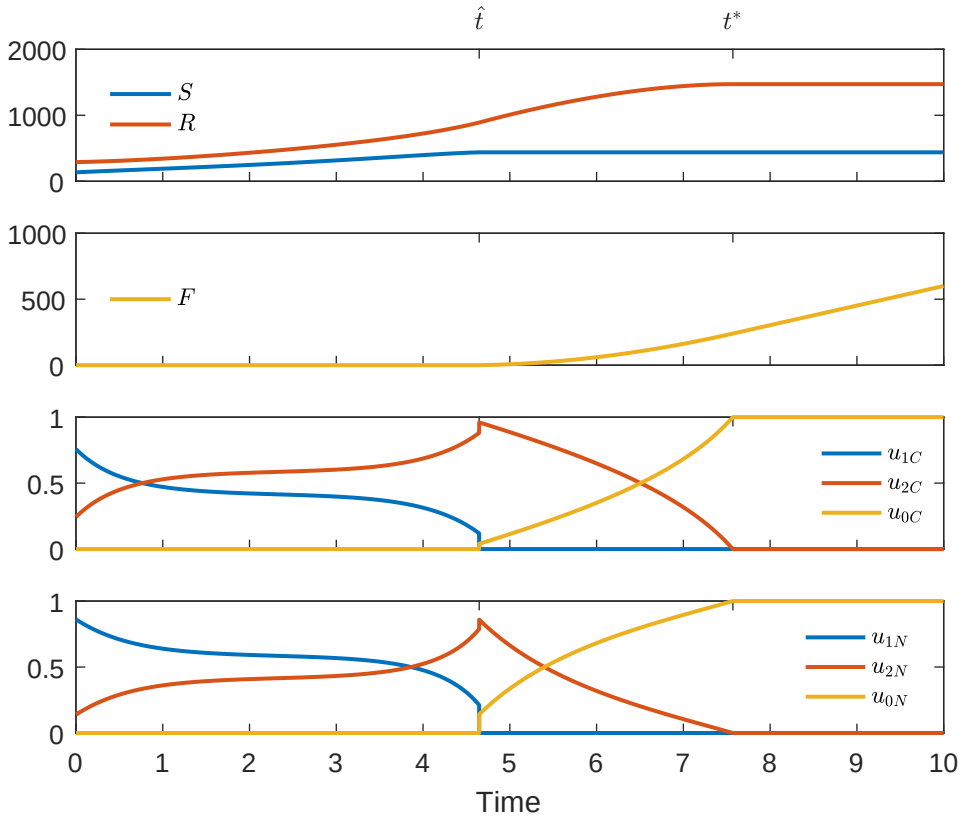

Figure 3.4: States and controls illustrating Type S initial balanced growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

Type S because the balanced growth phase resembles that of a plant that begins with shoot-only growth.

3.6.4 Balanced Growth First - Type R

Here we chose terminal conditions $S^* = 438.33$ and $R^* = 1471.54$. This results in $F(T) = 900$, $S_0 = 164.4$, and $R_0 = 239.89$. The states and controls are shown in Figure 3.5.

As in Figure 3.4, Figure 3.5 shows a simulation in which the plant begins in balance. However, unlike the Type S simulation, we see in Figure 3.5 a balanced growth phase that follows a similar allocation pattern to that seen in simulations

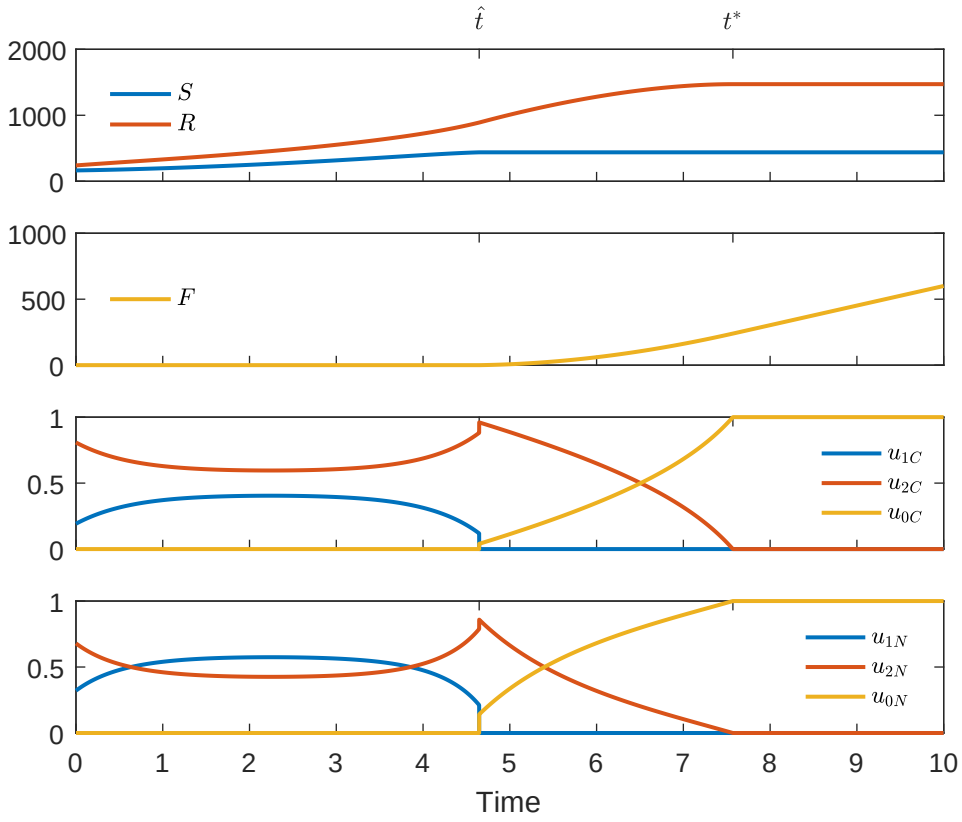

Figure 3.5: States and controls illustrating Type R initial balanced growth. From top to bottom: Shoots and roots, fruits, carbon controls, nitrogen controls. Shoots, roots, and fruits are given in units of carbon and the controls are dimensionless.

which begin with a phase of root-only growth. In particular, balanced growth in each consists of initially decreasing allocation to roots while increasing allocation to shoots, until some point about half-way through the balanced growth phase when this trend reverses and allocation to roots increases again. Therefore, we call this type of allocation pattern Type R.

3.6.5 Final Fruits Value Contours

As we did in Section 2.6.5, we again look for contours in the (S_0, R_0) plane which result in the same value of fruits at time T. For a given value of S_0, we used MATLAB's built-in nonlinear equation solver `fsolve` to find the correspond value of R_0 for which

the numerical scheme we outline in Section 3.5 would yield values of 400, 500, or 600 for $F(T)$. The resulting contours are shown in Figure 3.6.

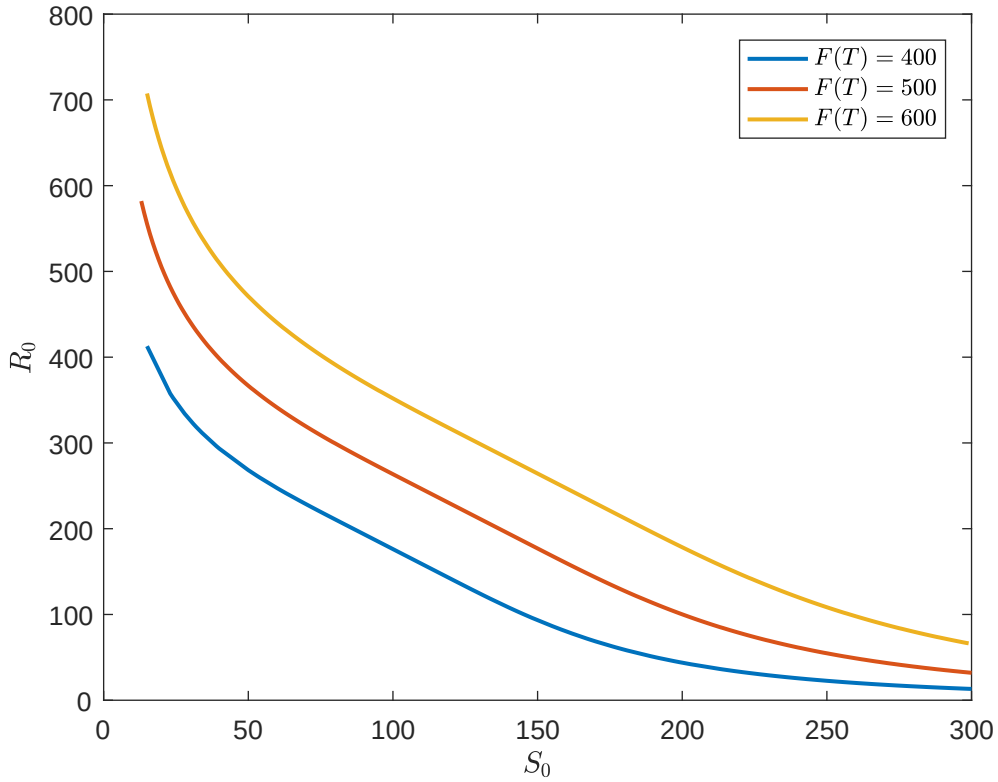

Figure 3.6: Contours depicting which initial conditions correspond to final fruit values of 400, 500, and 600. Shoots, roots, and fruits are given in units of carbon.

There are several key features to note about the contours in Figure 3.6. First, while each contour has a region which is nearly linear in the middle, the concavity is more pronounced at the ends, suggesting that plants which are strongly deficient in either initial roots or shoots will require an abundance of the other to compensate. That said, the contours become closer together near the ends, indicating that a small change in initial conditions in these regions will have a large effect on overall performance. If the plant begins in balance it takes a larger change in the initial conditions to move to a higher-yield contour.

3.6.6 600 Fruit Contour

As we did in Section 2.6.6, we will take a closer look at one of the contours in Figure 3.6. In particular, we will look at simulations with initial conditions along the $F(T) = 600$ contour. These correspond to 280 initial conditions, with S_0 between 15 and 300. As we did previously, we begin by identify which of the four types of growth discussed in Sections 3.6.1, 3.6.2, 3.6.3 and 3.6.4 each initial condition corresponds too. The $F(T) = 600$ contour is plotted again in Figure 3.7, and the coloration indicates the initial phase of growth.

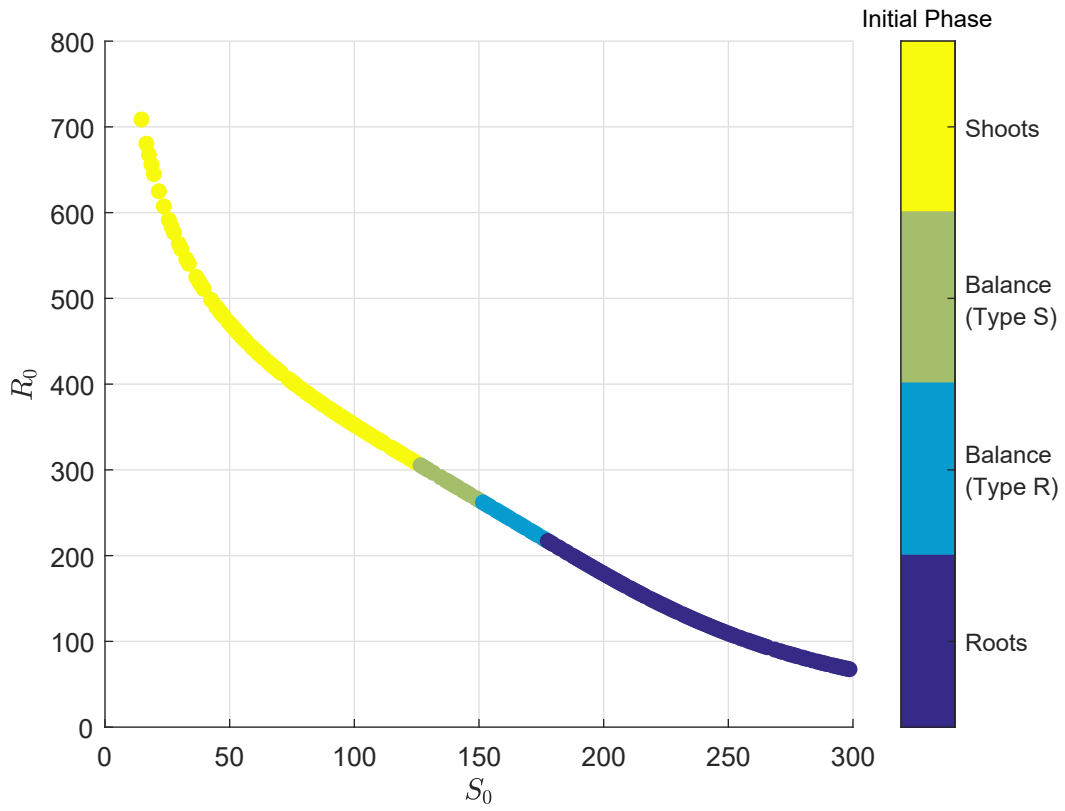

Figure 3.7: $F(T) = 600$ contour with coloration indicating the initial phase of growth. Shoots, roots, and fruits are given in units of carbon.

Additionally, to see the relationship between initial and terminal conditions, we plotted R^* vs. S^* two different ways along the $F(T) = 600$ contour. In the plot

on the left-hand side of Figure 3.8 we see the points in the (S^*, R^*) plane which correspond to points in the (S_0, R_0) plane in Figure 3.7. The coloration here has the same meaning as in Figure 3.7, indicating the initial phase of growth. The plot on the right-hand side of Figure 3.8 shows the frequency of points in 1×1 bins in the (S^*, R^*) plane. This plot was generated in MATLAB using `hist3`.

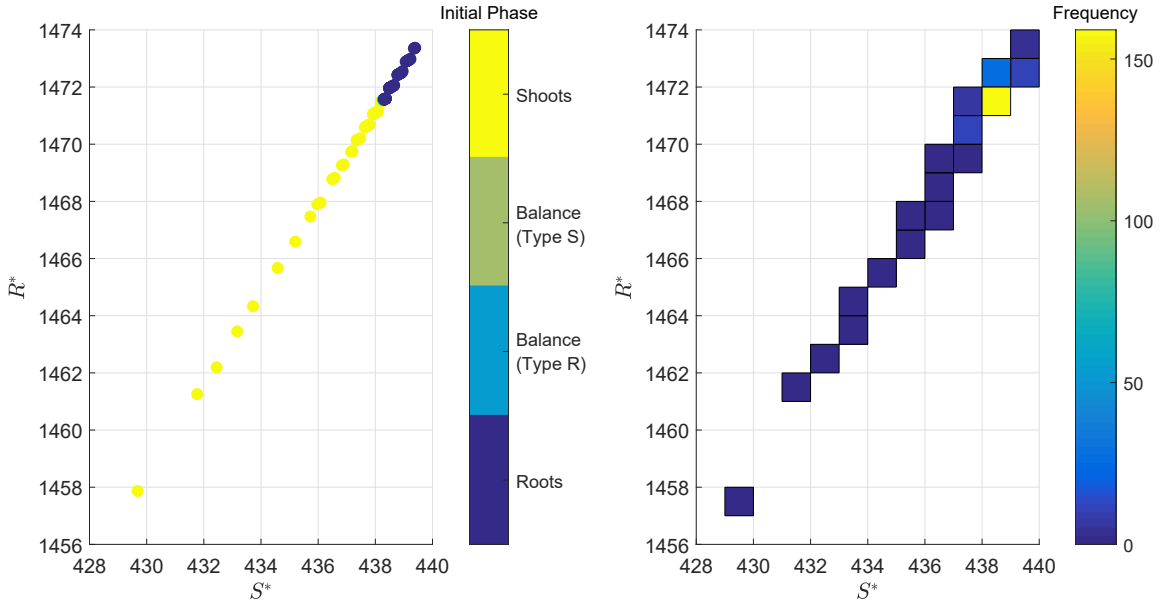

Figure 3.8: The plot on the left shows the terminal conditions corresponding to initial conditions along the $F(T) = 600$ contour. The plot on the right shows the corresponding frequency of points along the line on the left with points grouped into 1×1 bins. Shoots, roots, and fruits are given in units of carbon.

Note that in the left-hand plot in Figure 3.8 we see that plants which began with shoot-only growth reached $F(T) = 600$ with less overall biomass at the end as compared with plants which began with root-only growth. Furthermore, looking at the right-hand plot, we see that, as with the first model, the wide array of initial conditions represented by the $F(T) = 600$ contour result in tightly clustered terminal conditions.

Another way we can see this distinction between the initial and terminal conditions

is in the progression of the plots of $S(t)$ and $R(t)$ as the initial conditions progress along the $F(T) = 600$ contour. Note that we have already seen four plots of snapshots along the $F(T) = 600$ contour in Figures 3.2, 3.3, 3.4, and 3.5. In Figure 3.9 we have plotted the solution curves of $S(t)$ and $R(t)$ with initial conditions on the $F(T) = 600$ contour. The progression of solution curves from blue to red signifies the progression of initial conditions from left to right along the $F(T) = 600$ contour, that is from the low initial shoot/high initial root end to the low initial root/high initial shoot end. Again, notice the how the initial transients tend to go away by the end of balanced growth, when the shoots stop growing.

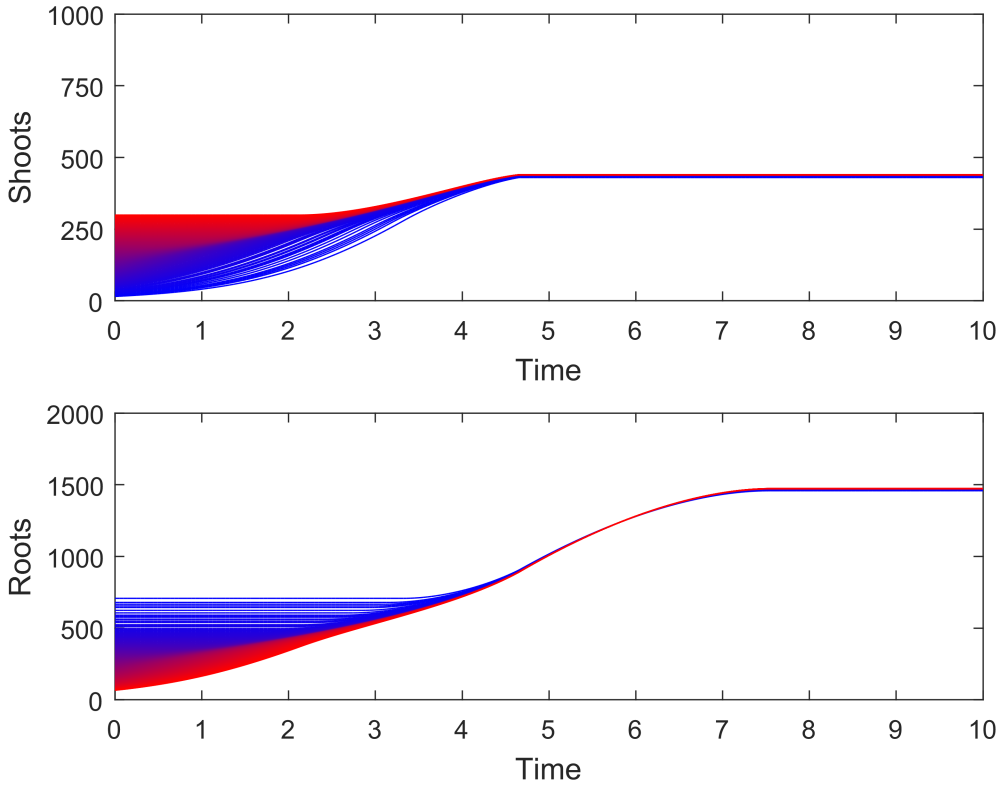

Figure 3.9: Progression of solutions curves of $S(t)$ and $R(t)$ along the $F(T) = 600$ contour. Going from blue to red corresponds to moving from left to right along the contour. Shoots, roots, and fruits are given in units of carbon.

To illustrate this one more 'level' in, we can repeat the same process used to

plot Figure 3.9 with the solution curves to $u_{2C}(t)$ (the fraction of carbon allocated to roots) along the $F(T) = 600$ contour. Again, the progression of solution curves from blue to red signifies the progression of initial conditions from left to right along the $F(T) = 600$ contour. These solution curves are shown in Figure 3.10. As with Figure 3.9, note how the initial transients seem to be eliminated by the end of balanced growth. That said, it is worth pointing out that whereas in the first model the choices of $C(S) = S$ and $N(R) = R$ determines the points at which the last two phases begin, this second model, with the same simplification, does not have this characteristic. While it's difficult to see in Figure 3.10, there is some small variation in when the last two phases begin, which is too large to be due to numerical error.

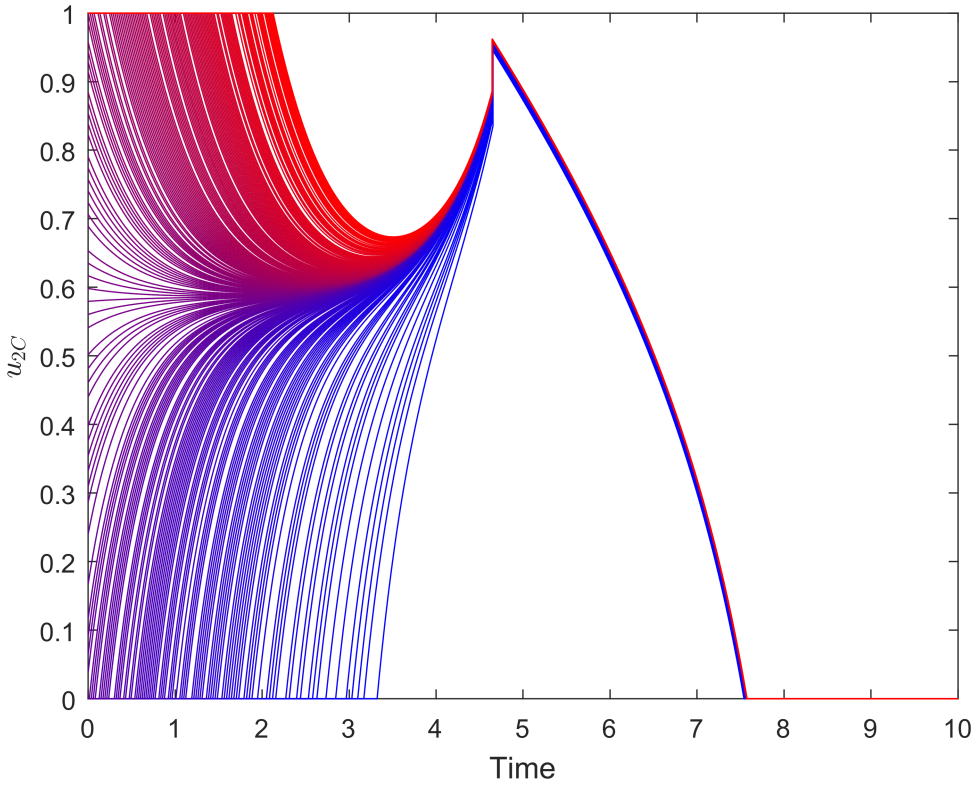

Figure 3.10: Progression of solution curves of $u_{2C}(t)$ (fraction of carbon allocated to roots) as the initial conditions (S_0, R_0) move along the $F(T) = 600$ contour. Going from blue to red corresponds to moving from left to right along the contour.

3.7 Discussion

Recall that in Chapter 2 we introduced a model for resource allocation in annual plants which served as an extension of Iwasa and Roughgarden's work in [5] to a two-resource model, in which the plant requires both carbon and nitrogen to grow. We made the simplifying assumption, however, that fruits only require carbon. In this chapter, we have extended our model from Chapter 2 to the more biologically realistic case in which fruits require both carbon and nitrogen. We used insight gained from the first model to form Ansatz 3.1 in regards to the structure of the optimal growth trajectory for a plant with C:N ratios ordered by $\nu_F < \nu_S < \nu_R$, and looked for solutions which matched our Ansatz. Perhaps unsurprisingly, we have recovered most of the results from our first model, providing weight to the general biological conclusions we discussed in Section 2.7. In this section, we will review these analogous results from this second model, as well as discuss the biological relevance, and outline some questions for further inquiry.

In Section 3.4 we established continuity of z_{0C}, z_{1C}, and z_{2C} between phases during which they are defined, with the exception of z_{2C} between the balanced growth phase and penultimate interval. Numerical evidence suggests that z_{2C} is continuous here, but we have yet to verify this analytically. As we discussed in Section 2.7, this means that, with the exception we mentioned above, whenever an organ is growing the amount of carbon allocated to that organ per unit of nitrogen allocated to that organ varies continuously in time.

In Section 3.6, particularly in Figures 3.2, 3.3, 3.4, and 3.5, we again see a spectrum of strategies for optimal growth, as we saw in the first model, with the exception that penultimate interval differs between the two models. We again observe allocation strategies that suggest a balance between avoiding resource limitation, and investing

in either roots or fruits. The plant begins with addressing deficiencies in either shoots or roots before entering a balanced growth phase. During balanced growth, we see the same types of allocation patterns as in the first model, those being what we termed Type S or Type R, depending on whether the plant initially emphasized shoots or roots, respectively. Unlike the first model, however, which saw an increase in allocation to shoots by the end of balanced growth, in this model we see an increase in allocation to roots by the end of balanced growth. This reversal is due to the fact that we have reordered the C:N ratios so that instead of $\nu_F = \infty$ as in the first model, here we have ν_F less than the other two C:N ratios. During the penultimate interval, then, we see a gradual shift from primarily investing in roots to exclusively investing in fruits, reflecting the fact that here fruit growth is more important that shoot growth for overall fruit yield, but not so much as to preclude investment in roots.

As with the first model, we see an equalizing effect that seems to eliminate initial transients by the end of balanced growth. Looking at Figures 3.6 and 3.7, we again see the wide range of initial conditions which result in $F(T) = 600$. Figure 3.8 shows a similar tight clustering of the terminal conditions that we saw in the first model. Looking at the particular strategies employed for plants with initial conditions along the $F(T) = 600$ contour, in Figures 3.9 and 3.10, we again see a high degree of variation in the strategies used in first half of the growing season, and only minimal variation in the second half of the growing season. It is worth noting here that, while the later stages appear primarily blue in both Figures 3.9 and 3.10, this is most likely an artifact of which piece of the $F(T) = 600$ contour we examined. Figure 3.7 shows that this region of the contour gets closer to $S_0 = 0$ than it does $R_0 = 0$, and so we see more variation toward that extreme, resulting in more visible blue in the latter half of the growing season in the plots S, R, and u_{2C}. As we concluded for the first model,

these results suggest that optimal growth may drive a population toward common sizes and optimal yields by means of variation in allocation strategy.

3.7.1 Future Directions

So far we have only looked for solutions which follow Ansatz 3.1, which while reasonable based on our work with the first model, begs the question as to whether it always holds true for the given ordering of C:N ratios, or whether there may be some region of the (ν_S, ν_R, ν_F) parameter space for which it fails. Furthermore, it seems likely that for some regions of this parameter space we may recover the structure of the first model (presumably where $\nu_S < \nu_R < \nu_F$), or the three-phase structure seen by Iwasa and Roughgarden in [5].

A natural extension of this work would involve making the model more biologically reasonable. In particular, we've assumed complete utilization of resources, when in reality we should include the possibility of resource rejection at the synthesizing units (see [7]). Furthermore, we have assumed that the carbon and nitrogen fluxes depend only on shoots and roots, respectively, whereas the carbon flux in particular should depend on both shoots and roots, because water is required for carbon capture.

At this point we have considered annual plants growing in isolation. It would be interesting to expand our work to different classes of plants, for which fruit yield would no longer be a complete measure of fitness. Here our objective function would need to include a measure of survival likelihood, as well as reproductive fitness. Lastly, the question remains as to whether the allocation trends we observe still exist when the effects of competition and other community dynamics are considered.

APPENDIX A

Necessary Conditions

In this appendix we will derive the necessary conditions presented in (2.19) in Section 2.2.4 and in (3.6) in Section 3.2.3. Note that because the carbon controls do not directly depend on the nitrogen controls we can obtain the necessary conditions for the first model by considering an n-state two-control problem and an n-state three-control problem separately, and for the second model we have two separate n-state three-control problems. In our particular situation $n = 3$, but we derive the conditions in more generality because there is minimal complexity added to the derivation. In all the following we consider $f, \vec{g}$ to be continuously differentiable in all arguments and $\vec{u}$ to be piecewise continuous. We begin with an n-state two-control problem in Appendix A.1 followed by an n-state three-control problem in Appendix A.2

A.1 n States, 2 Controls, Interval $[0, T]$

We consider the optimal control problem give by (A.1), with the goal of deriving the associated necessary conditions.

$$\max_{\vec{u}=\langle u_1,u_2\rangle} \int_0^T f(t,\vec{x}(t),\vec{u}(t))\,dt$$

$$\begin{aligned} \text{subject to:}\quad & 0 \le u_1(t) \le 1, \quad 0 \le u_2(t) \le 1 \text{ for } t \in [0,T], \\ & u_1(t) + u_2(t) = 1 \text{ for } t \in [0,T] \\ & \vec{x}\,'(t) = \vec{g}(t,\vec{x}(t),\vec{u}(t)), \quad \vec{x}(0) = \vec{x}_0 \end{aligned} \tag{A.1}$$

To this end, let U be the space of admissible controls, and define the functional $J\colon U \to \mathbb{R}$ by

$$\begin{aligned} J(\vec{u}) &= \int_0^T f(t,\vec{x}(t),\vec{u}(t))\,dt \\ &= \int_0^T \Big\{ f(t,\vec{x}(t),\vec{u}(t)) + \vec{\lambda}(t)\cdot(\vec{g}(t,\vec{x}(t),\vec{u}(t)) - \vec{x}\,'(t)) \Big\}\,dt \\ &= \int_0^T \Big\{ f(t,\vec{x}(t),\vec{u}(t)) + \vec{\lambda}(t)\cdot\vec{g}(t,\vec{x}(t),\vec{u}(t)) + \vec{\lambda}\,'(t)\cdot\vec{x}(t) \Big\}\,dt \\ &\qquad + \vec{\lambda}(0)\cdot\vec{x}_0 - \vec{\lambda}(T)\cdot\vec{x}(t), \end{aligned} \tag{A.2}$$

where $\vec{\lambda}(t)$ is a piecewise differentiable function to be specified later, and the final equality was obtained by integrating by parts. Note that by $\vec{\lambda}\,'$ we mean the function whose n^{th} component is the time derivative of the n^{th} component of $\vec{\lambda}$. We use the same notation for $\vec{x}\,'(t)$. Let $\vec{u}^*, \vec{x}^*$ be optimal, and for a piecewise continuous variations h_1, h_2 and $\varepsilon \in \mathbb{R}$ define $\langle u_1^\varepsilon, u_2^\varepsilon\rangle \in U$ by $u_1^\varepsilon(t) = u_1^*(t) + \varepsilon h_1(t)$ and $u_2^\varepsilon(t) = u_2^*(t) + \varepsilon h_2(t)$, and let $\vec{x}^\varepsilon$ be the corresponding state. Note that admissibility requires $u_1^\varepsilon + u_2^\varepsilon = 1$, so we have $h_1 = -h_2$. Because $\vec{u}^*, \vec{x}^*$ are optimal, we have

$$0 \ge \frac{dJ(\vec{u}^\varepsilon)}{d\varepsilon}\Big|_{\varepsilon=0} = \lim_{\varepsilon\to 0} \frac{J(\vec{u}^\varepsilon) - J(\vec{u}^*)}{\varepsilon}. \tag{A.3}$$

We have inequality rather than equality above because the boundedness of the controls makes it possible that optimality is only global rather than local, and the numerator in the difference quotient is nonpositive because $J(\vec{u}^*)$ is maximal over admissible controls. We differentiate (A.2) with respect to ε and note that conditions on the functions involved in the integrand allow us to interchange the derivative and integral via the Dominated Convergence Theorem. Suppressing the arguments from here on,

$$\begin{aligned} 0 \geq & \int_0^T \frac{\partial}{\partial \varepsilon} \left(f + \vec{\lambda} \cdot \vec{g} + \vec{\lambda}' \cdot \vec{x}^{\varepsilon} \right) \Big|_{\varepsilon=0} dt - \vec{\lambda}(T) \cdot \frac{\partial \vec{x}^{\varepsilon}(t)}{\partial \varepsilon} \Big|_{\varepsilon=0} \\ = & \int_0^T \left\{ \left(\nabla f + \sum_{i=1}^{n} \lambda_i \nabla g_i + \vec{\lambda}' \right) \cdot \frac{\partial \vec{x}^{\varepsilon}(t)}{\partial \varepsilon} \Big|_{\varepsilon=0} \right. \\ & \left. + \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} \right) h_1 + \left(f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} \right) h_2 \right\} dt \\ & - \vec{\lambda}(T) \cdot \frac{\partial \vec{x}^{\varepsilon}(t)}{\partial \varepsilon} \Big|_{\varepsilon=0}. \end{aligned}$$

Note that here we are using ∇f to mean the vector of derivatives of f with respect each component of $\vec{x}$, and likewise for ∇g_i. Choosing $\vec{\lambda}$ to satisfy

$$\vec{\lambda}'(t) = -\left(\nabla f + \sum_{i=1}^{n} \lambda_i \nabla g_i \right), \quad \vec{\lambda}(T) = \vec{0},$$

the inequality reduces to

$$0 \geq \int_0^T \left\{ \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} \right) h_1 + \left(f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} \right) h_2 \right\} dt.$$

Using the fact that $h_1 = -h_2$, we obtain

$$0 \geq \int_0^T \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \right) h_1 \, dt.$$

Next, we use this inequality to obtain necessary conditions for optimality. Let s be a point of continuity for u_1^* and u_2^* such that $0 \leq u_1^*(s) < 1$, and suppose for the sake of contradiction that $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} > 0$ at s. As $\vec{\lambda}$ is continuous, and f and $\vec{g}$ are continuously differentiable, by the continuity of u_1^* and u_2^* at s we have that $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2}$ is continuous at s as well. Therefore we can find a closed interval I about s such that $u_1^* < 1$ and $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} > 0$ on I. Let

$$M = \max\{u_1^*(t) \,|\, t \in I\} < 1,$$

and choose h_1 and h_2 to be

$$h_1(t) = (1 - M)\chi_I(t), \quad h_2(t) = (M - 1)\chi_I(t),$$

where χ_I is the characteristic function on I. Note that this gives us

$$u_1^\varepsilon = u_1^* + \varepsilon(1 - M)\chi_I, \quad u_2^\varepsilon = u_2^* + \varepsilon(M - 1)\chi_I.$$

As $u_1^* \leq M$ on I, then it must be the case that $u_2^* \geq 1 - M$ on I since $u_2^* = 1 - u_1^*$. This means that for all $\varepsilon \in [0, 1]$ we have

$$0 \leq u_1^\varepsilon \leq 1, \quad 0 \leq u_2^\varepsilon \leq 1,$$

and as the variations were chosen such that $h_1 + h_2 = 0$, we have that these variations lead to admissible controls. We have then

$$\begin{aligned} 0 &\geq \int_0^T \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \right) h_1 \, dt \\ &= \int_I \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \right) (1 - M) \, dt \\ &> 0, \end{aligned}$$

a contradiction. So, it must be the case that $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \leq 0$. Note that because the controls sum to one we have that $0 < u_2^* \leq 1$ implies $0 \leq u_1^* < 1$, and so we actually have the following condition

$$0 \leq u_1^* < 1 \quad \text{or} \quad 0 < u_2^* \leq 1 \implies f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \leq 0.$$

Interchanging the roles of u_1 and u_2 in the argument above also gives us

$$0 < u_1^* \leq 1 \quad \text{or} \quad 0 \leq u_2^* < 1 \implies f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \geq 0.$$

So, together we have the that

$$\begin{cases} 0 \leq u_1^* < 1 \quad \text{or} \quad 0 < u_2^* \leq 1 \implies f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \leq 0 \\ 0 < u_1^* \leq 1 \quad \text{or} \quad 0 \leq u_2^* < 1 \implies f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \geq 0 \end{cases} \tag{A.4}$$

Now, suppose that we are in the case where $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} < 0$. By (A.4) we can rule out the possibility that either $u_1^* > 0$ or $u_2^* < 1$ because either choice would lead to a contradiction. Therefore, in this case we can conclude that $u_1^* = 0$

and $u_2^* = 1$. Similar arguments lead to the following conditions:

$$\begin{cases} f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} < 0 \implies u_1^* = 0, u_2^* = 1 \\ f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} > 0 \implies u_1^* = 1, u_2^* = 0 \\ f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} = 0 \implies 0 < u_1^*, u_2^* < 1. \end{cases} \tag{A.5}$$

We can convert this into conditions involving a Hamiltonian as follows. Define H by

$$H(t, x, \vec{u}, \lambda) := f(t, x, \vec{u}) + \vec{\lambda} \cdot \vec{g}(t, x, \vec{u}).$$

For distinct indicies $i, j \in \{1, 2\}$, rewriting (A.5) in terms of H yields the necessary conditions

$$\begin{cases} \frac{\partial H}{\partial u_i} < \frac{\partial H}{\partial u_j} \implies u_i^* = 0, u_j^* = 1 \\ \frac{\partial H}{\partial u_i} = \frac{\partial H}{\partial u_j} \implies 0 \leq u_i^*, u_j^* \leq 1. \end{cases} \tag{A.6}$$

We can also express the differential equations for $\vec{x}$ and $\vec{\lambda}$ in terms of the Hamiltonian as

$$\begin{cases} x_i'(t) = \frac{\partial H}{\partial \lambda_i}, \quad x_i(0) = x_{i0} \text{ for } i = 1, \ldots, n \\ \lambda_i'(t) = -\frac{\partial H}{\partial x_i}, \quad \lambda_i(T) = 0 \text{ for } i = 1, \ldots, n. \end{cases} \tag{A.7}$$

Note that if we were to reduce the problem to one control by writing $u_2 = 1 - u_1$ these conditions would become exactly the standard necessary conditions for a problem with one bounded control and n states. Going through this derivation with two controls, however, gives us a starting point to approach the analogous problem with three controls.

A.2 n States, 3 Controls, Interval $[0, T]$

We now consider the following optimal control problem with n states and three controls. We will again derive the necessary conditions, following a similar procedure to that used in the simpler case with only two controls. The problem is stated as follows.

$$\max_{\vec{u}=\langle u_1,u_2,u_3\rangle} \int_0^T f(t, \vec{x}(t), \vec{u}(t))\, dt$$

$$\begin{aligned} \text{subject to:} \quad & 0 \le u_1(t) \le 1, \quad 0 \le u_2(t) \le 1, \quad 0 \le u_3(t) \le 1 \text{ for } t \in [0, T], \\ & u_1(t) + u_2(t) + u_3(t) = 1 \text{ for } t \in [0, T] \\ & \vec{x}\,'(t) = \vec{g}(t, \vec{x}(t), \vec{u}(t)), \quad \vec{x}(0) = \vec{x}_0 \end{aligned} \tag{A.8}$$

The derivation of the necessary conditions is similar to the two-control case. We let U be the space of admissible controls for (A.8) and define $J\colon U \to \mathbb{R}$ by (A.2), restated below.

$$\begin{aligned} J(\vec{u}) = \int_0^T \Big\{ f(t, \vec{x}(t), \vec{u}(t)) + \vec{\lambda}(t) \cdot \vec{g}(t, \vec{x}(t), \vec{u}(t)) + \vec{\lambda}\,'(t) \cdot \vec{x}(t) \Big\}\, dt \\ + \vec{\lambda}(0) \cdot \vec{x}_0 - \vec{\lambda}(T) \cdot \vec{x}(t). \end{aligned} \tag{(A.2) revisited}$$

Suppose that $\vec{u}^*, \vec{x}^*$ are optimal, and let h_1, h_2, and h_3 be piecewise continuous variations. Then for $\varepsilon \in \mathbb{R}$ we define $\vec{u}^{\,\varepsilon} \in U$ by $u_i^\varepsilon(t) = u_i^*(t) + \varepsilon h_i(t)$ for $i = 1, 2, 3$, and let $\vec{x}^{\,\varepsilon}$ be the corresponding state. Admissibility here requires that $\sum_{i=1}^3 u_i^\varepsilon = 1$ so we have that $h_3 = -(h_1 + h_2)$. By the optimality of $\vec{u}^*, \vec{x}^*$ we again get (A.3), restated below:

$$0 \ge \frac{dJ(\vec{u}^{\,\varepsilon})}{d\varepsilon}\Big|_{\varepsilon=0} = \lim_{\varepsilon \to 0} \frac{J(\vec{u}^{\,\varepsilon}) - J(\vec{u}^*)}{\varepsilon}. \tag{(A.3) revisited}$$

We again differentiate (A.2), using the DCT to interchange the order of differentiation and integration, this time arriving at

$$\begin{aligned}0 &\geq \int_0^T \frac{\partial}{\partial \varepsilon}\left(f+\vec{\lambda}\cdot\vec{g}+\vec{\lambda}'\cdot\vec{x}^{\varepsilon}\right)\Big|_{\varepsilon=0} dt - \vec{\lambda}(T)\cdot\frac{\partial \vec{x}^{\varepsilon}(t)}{\partial\varepsilon}\Big|_{\varepsilon=0} \\ &= \int_0^T \left\{\left(\nabla f + \sum_{i=1}^{n}\lambda_i\nabla g_i + \vec{\lambda}'\right)\cdot\frac{\partial \vec{x}^{\varepsilon}(t)}{\partial\varepsilon}\Big|_{\varepsilon=0} + \left(f_{u_1}+\vec{\lambda}\cdot\vec{g}_{u_1}\right)h_1 \right. \\ &\qquad \left. + \left(f_{u_2}+\vec{\lambda}\cdot\vec{g}_{u_2}\right)h_2 + \left(f_{u_3}+\vec{\lambda}\cdot\vec{g}_{u_3}\right)h_3\right\} dt - \vec{\lambda}(T)\cdot\frac{\partial \vec{x}^{\varepsilon}(t)}{\partial\varepsilon}\Big|_{\varepsilon=0}.\end{aligned}$$

As before, we choose $\vec{\lambda}$ so that

$$\vec{\lambda}'(t) = -\left(\nabla f + \sum_{i=1}^{n}\lambda_i\nabla g_i\right), \quad \vec{\lambda}(T) = \vec{0},$$

and so we arrive at the inequality

$$0 \geq \int_0^T \left\{\left(f_{u_1}+\vec{\lambda}\cdot\vec{g}_{u_1}\right)h_1 + \left(f_{u_2}+\vec{\lambda}\cdot\vec{g}_{u_2}\right)h_2 + \left(f_{u_3}+\vec{\lambda}\cdot\vec{g}_{u_3}\right)h_3\right\} dt. \tag{A.9}$$

We will use this inequality to derive the necessary conditions for optimality. Due to the similarity between the various cases we will only show one in detail. We begin by using the substitution $h_3 = -(h_1+h_2)$ to rewrite (A.9) as

$$0 \geq \int_0^T \left\{\left(f_{u_1}+\vec{\lambda}\cdot\vec{g}_{u_1} - f_{u_3} - \vec{\lambda}\cdot\vec{g}_{u_3}\right)h_1 + \left(f_{u_2}+\vec{\lambda}\cdot\vec{g}_{u_2} - f_{u_3} - \vec{\lambda}\cdot\vec{g}_{u_3}\right)h_2\right\} dt. \tag{A.10}$$

Next, let s be a point of continuity for all controls so that $0 \leq u_1^*(s) < 1$ and $0 < u_3^*(s) \leq 1$. Note that because the controls sum to one this also means that $0 \leq u_2^*(s) < 1$. Additionally, assume for the sake of contradiction that $f_{u_1}+\vec{\lambda}\cdot\vec{g}_{u_1} - f_{u_3} - \vec{\lambda}\cdot\vec{g}_{u_3} > 0$ at s. As $\vec{\lambda}$ is continuous, and f and $\vec{g}$ are continuously differentiable, by the continuity of the controls we have that $f_{u_1}+\vec{\lambda}\cdot\vec{g}_{u_1} - f_{u_3} - \vec{\lambda}\cdot\vec{g}_{u_3}$ is continuous at

s as well. Therefore, we can find a closed interval I about s such that $u_1^* < 1, u_3^* > 0$, and $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} > 0$ on I. Next, let

$$m = \min \left\{ 1 - \max\{u_1^*(t) \mid t \in I\}, \min\{u_3^*(t) \mid t \in I\} \right\}.$$

and note that $m > 0$. Next, we choose variations to be

$$h_1(t) = m\chi_I(t), \quad h_2(t) \equiv 0, \quad h_3(t) = -m\chi_I(t).$$

This gives us the controls

$$u_1^\varepsilon(t) = u_1^*(t) + \varepsilon m\chi_I(t), \quad u_2^\varepsilon(t) = u_2^*(t), \quad u_3^\varepsilon(t) = u_3^*(t) - \varepsilon m\chi_I(t).$$

Note that because $\sum_{i=1}^3 u_i^*(t) = 1$ and $\sum_{i=1}^3 h_i(t) = 0$ we have that $\sum_i^3 u_i^\varepsilon(t) = 1$ as well. Furthermore, restricting our attention to $t \in I$ and $\varepsilon \in [0, 1]$, we have

$$\begin{aligned} 0 &\leq u_1^\varepsilon(t) \\ &= u_1^*(t) + \varepsilon m \\ &\leq u_1^*(t) + \varepsilon(1 - \max\{u_1^*(t)\}) \\ &\leq u_1^*(t) + 1 - \max\{u_1^*(t)\} \\ &\leq 1. \end{aligned}$$

As this bound also holds outside of I by assumption we have that $0 \leq u_1^\varepsilon \leq 1$. Now, for u_3^ε, again restricting our attention to $t \in I$ and $\varepsilon \in [0,1]$ we have

$$\begin{aligned} 1 &\geq u_3^\varepsilon(t) \\ &= u_3^*(t) - \varepsilon m \chi_I(t) \\ &\geq u_3^*(t) - \varepsilon \min\{u_3^*(t)\} \\ &\geq u_3^*(t) - \min\{u_3^*(t)\} \\ &\geq 0. \end{aligned}$$

Again, as this bound holds outside of I by assumption we have that $0 \leq u_3^\varepsilon \leq 1$. Therefore, the controls $u_1^\varepsilon, u_2^\varepsilon$, and u_3^ε are admissible for all $\varepsilon \in [0,1]$. We have then

$$\begin{aligned} 0 &\geq \int_0^T \left\{ \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \right) h_1 + \left(f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \right) h_2 \right\} dt \\ &= \int_I \left(f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \right) m \, dt \\ &> 0, \end{aligned}$$

a contradiction. Therefore it must be the case that $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \leq 0$.

Now, by swapping the indicies 1 and 2 in the argument above we can reach the additional conclusion that $f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \leq 0$ for the same set of assumptions. Note that this is because if we assume that $0 < u_3^* \leq 1$ then we can conclude both $0 \leq u_1^* < 1$ and $0 \leq u_2^* < 1$ regardless of which assumption is used for a particular argument. Therefore, we have that if we are in the case that $0 \leq u_1^* < 1$, $0 \leq u_2^* < 1$, and $0 < u_3^* \leq 1$ then we have that both $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \leq 0$ and $f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \leq 0$. We can permute the roles of u_1, u_2, and u_3, or, equivalently, repeat the argument above with the substitutions of $h_1 = -(h_3 + h_2)$ or

$h_2 = -(h_3 + h_1)$ into (A.9) to obtain the following.

$$\begin{cases} 0 \leq u_1^*, u_2^* < 1, \quad 0 < u_3^* \leq 1 \implies \begin{cases} f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \leq 0 \\ f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} \leq 0 \end{cases} \\ 0 \leq u_1^*, u_3^* < 1, \quad 0 < u_2^* \leq 1 \implies \begin{cases} f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \leq 0 \\ f_{u_3} + \vec{\lambda} \cdot \vec{g}_{u_3} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \leq 0 \end{cases} \\ 0 \leq u_2^*, u_3^* < 1, \quad 0 < u_1^* \leq 1 \implies \begin{cases} f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_1} - \vec{\lambda} \cdot \vec{g}_{u_1} \leq 0 \\ f_{u_3} + \vec{\lambda} \cdot \vec{g}_{u_3} - f_{u_1} - \vec{\lambda} \cdot \vec{g}_{u_1} \leq 0 \end{cases} \end{cases} \tag{A.11}$$

We will now work to develop implications going in the other direction. We will illustrate this for a few cases and the remaining cases follow from permuting the controls. First, consider the case that at a point $t \in [0, T]$ we have $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} < 0$ and $f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} < 0$, and suppose for the sake of contradiction that $u_3^*(t) < 1$. Then it must be the case that at t either $0 < u_1^* \leq 1$ and $0 \leq u_2^* < 1$ or $0 \leq u_1^* < 1$ and $0 < u_2^* \leq 1$. Using (A.11) we see that the first case implies that $f_{u_3} + \vec{\lambda} \cdot \vec{g}_{u_3} - f_{u_1} - \vec{\lambda} \cdot \vec{g}_{u_1} \leq 0$ at t and the second case implies that $f_{u_3} + \vec{\lambda} \cdot \vec{g}_{u_3} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} \leq 0$ at t, both of which contradict our assumptions. So, it must be the case that $u_3^*(t) = 1$.

Next, consider the case that at a point $t \in [0, T]$ we have $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} < 0$ and $f_{u_1} + \vec{\lambda} \cdot \vec{g}_{u_1} - f_{u_2} - \vec{\lambda} \cdot \vec{g}_{u_2} < 0$, and suppose for the sake of contradiction that $u_1^*(t) > 0$. Then it must be the case that at t we have $0 \leq u_2^* < 1$ and $0 \leq u_3^* < 1$. Using (A.11) we see that this implies that $f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_1} - \vec{\lambda} \cdot \vec{g}_{u_1} \leq 0$ and $f_{u_3} + \vec{\lambda} \cdot \vec{g}_{u_3} - f_{u_1} - \vec{\lambda} \cdot \vec{g}_{u_1} \leq 0$ at t, both of which contradict our assumptions here. Therefore, it must be the case that $u_1^*(t) = 0$. A similar argument can be used to

show that if $f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_1} - \vec{\lambda} \cdot \vec{g}_{u_1} < 0$ and $f_{u_2} + \vec{\lambda} \cdot \vec{g}_{u_2} - f_{u_3} - \vec{\lambda} \cdot \vec{g}_{u_3} < 0$ at a point $t \in [0, T]$ then it must be the case that $u_2^*(t) = 0$.

By permuting the roles of the controls in the arguments above we can reach the following conclusions. For distinct indicies $i, j, k \in \{1, 2, 3\}$ we have that

$$\begin{cases} \begin{cases} f_{u_j} + \vec{\lambda} \cdot \vec{g}_{u_j} - f_{u_i} - \vec{\lambda} \cdot \vec{g}_{u_i} < 0 \\ f_{u_k} + \vec{\lambda} \cdot \vec{g}_{u_k} - f_{u_i} - \vec{\lambda} \cdot \vec{g}_{u_i} < 0 \end{cases} & \implies u_i^* = 1 \\ \begin{cases} f_{u_i} + \vec{\lambda} \cdot \vec{g}_{u_i} - f_{u_j} - \vec{\lambda} \cdot \vec{g}_{u_j} < 0 \\ f_{u_i} + \vec{\lambda} \cdot \vec{g}_{u_i} - f_{u_k} - \vec{\lambda} \cdot \vec{g}_{u_k} < 0 \end{cases} & \implies u_i^* = 0 \end{cases} \tag{A.12}$$

Note that this also tells us that the only way for a control to take on a value other than zero or one is for at least one of the inequalities in (A.12) to be an equality instead. We can convert this into conditions involving a Hamiltonian as follows. Define H by

$$H(t, x, \vec{u}, \lambda) := f(t, x, \vec{u}) + \vec{\lambda} \cdot \vec{g}(t, x, \vec{u}).$$

Rewriting (A.12) in terms of H yields the necessary conditions

$$\begin{cases} \frac{\partial H}{\partial u_j} < \frac{\partial H}{\partial u_i} \quad \text{and} \quad \frac{\partial H}{\partial u_k} < \frac{\partial H}{\partial u_i} & \implies u_i^* = 1 \\ \frac{\partial H}{\partial u_i} < \frac{\partial H}{\partial u_j} \quad \text{and} \quad \frac{\partial H}{\partial u_i} < \frac{\partial H}{\partial u_k} & \implies u_i^* = 0 \\ \frac{\partial H}{\partial u_i} = \frac{\partial H}{\partial u_j} \quad \text{or} \quad \frac{\partial H}{\partial u_i} = \frac{\partial H}{\partial u_k} \quad \text{or} \quad \frac{\partial H}{\partial u_j} = \frac{\partial H}{\partial u_k} & \implies 0 \leq u_i^* \leq 1. \end{cases} \tag{A.13}$$

We can also express the differential equations for $\vec{x}$ and $\vec{\lambda}$ in terms of the Hamiltonian:

$$\begin{cases} x_i'(t) = \frac{\partial H}{\partial \lambda_i}, & x_i(0) = x_{i0} \text{ for } i = 1, \ldots, n \\ \lambda_i'(t) = -\frac{\partial H}{\partial x_i}, & \lambda_i(T) = 0 \text{ for } i = 1, \ldots, n. \end{cases} \tag{A.14}$$

APPENDIX B

SUPPLEMENTAL ARGUMENTS AND DERIVATIONS

This appendix includes supplemental arguments and derivations omitted from the main body of the text. In particular, Appendix B.1 includes an argument, referenced in Section 2.3.3, for why the penultimate interval cannot be preceded by an interval of mixed root/shoot/fruit growth. Appendix B.2 includes a derivation, referenced in Section 2.5.5, for the differential equation for z_{1C} and z_{2C} during balanced growth for the first model. Lastly, Appendix B.3 includes a derivation, referenced in Section 3.5.2 for the differential equations for z_{0C} and z_{2C} during the penultimate interval for the second model.

B.1 First Model Growth Stage Argument

In this appendix we will show that an interval of mixed root/shoot/fruit growth cannot precede the penultimate interval of shoot/fruit growth in the first model. To this end, first note that if there were a phase of mixed growth between all three organs, the necessary conditions (2.19) would dictate that we have

$$\lambda_1 G'(z_{1C}) = \lambda_2 G'(z_{2C}) = 1 \tag{B.1}$$

$$\lambda_1 \nu_S G_2(z_{1C}) = \lambda_2 \nu_R G_2(z_{2C}). \tag{B.2}$$

We can eliminate λ_1 and λ_2 from (B.1) and (B.2) to obtain

$$\frac{G'(z_{2C})}{G_2(z_{2C})} = \frac{\nu_R}{\nu_S}\frac{G'(z_{1C})}{G_2(z_{1C})}. \tag{B.3}$$

Additionally, recalling that by (2.53) we have that $H = C^*$, and so we can rewrite the Hamiltonian (2.25) during this phase as

$$C^* = u_{0C}C + \lambda_1\nu_S u_{1N}NG(z_{1C}) + \lambda_2\nu_R u_{2N}NG(z_{2C}). \tag{B.4}$$

We can rewrite this using (2.14), and then use (B.1) and (B.2) as follows:

$$\begin{aligned} C^* &= u_{0C}C + \lambda_1\nu_S u_{1N}N\left[G_2(z_{1C}) + \frac{u_{1C}C}{\nu_S u_{1N}N}G'(z_{1C})\right] \\ &\quad + \lambda_2\nu_R u_{2N}N\left[G_2(z_{2C}) + \frac{u_{2C}C}{\nu_R u_{2N}N}G'(z_{2C})\right] \\ &= \lambda_2\nu_R NG_2(z_{2C})(u_{1N} + u_{2N}) + \lambda_2 CG'(z_{2C})(u_{0C} + u_{1C} + u_{2C}) \\ &= \lambda_2\left[\nu_R NG_2(z_{2C}) + CG'(z_{2C})\right]. \end{aligned}$$

This then gives us

$$\lambda_2 = \frac{C^*}{\nu_R NG_2(z_{2C}) + CG'(z_{2C})}. \tag{B.5}$$

Note that this is exactly equation (2.86) that we have during mixed root/shoot growth as well. This means that the same argument we used in Section 2.4.7, to show that z_{1C} is continuous between balanced growth and the penultimate interval, applies here as well to show that z_{1C} must be continuous between an interval of mixed root/shoot/fruit growth and an interval of shoot/fruit growth.

Now, by (2.39) and (2.40) we have that $\lambda_1' = -C_S$ and $\lambda_2' = -N_R\lambda_1\nu_S G_2(z_{1C})$ during this root/shoot/fruit phase, as well as the penultimate interval. As C_S is assumed to be continuous, this means that λ_1 is continuously differentiable during

these two intervals, which in turn means that z_{1C} is continuously differentiable as well because in both the root/shoot/fruit interval and the penultimate interval we have that $\lambda_1 G'(z_{1C}) = 1$. Furthermore, this, along with the fact that N_R is continuous, means that λ_2 is continuously differentiable here as well. In particular, we can say that z_{1C}, λ_1, and λ_2 are continuously differentiable at the transition point between these two intervals, $\hat{t}$.

Now, although z_{2C} is not defined in the penultimate interval, we can use (B.3) to define the left-hand limit of z_{2C} at any point in the penultimate interval, as if that point were, in fact, $\hat{t}$. We will refer to this as z_{2C}^-, and note that because z_{1C} is continuously differentiable at $\hat{t}$ we have by (B.3) that z_{2C}^- is as well, provided that z_{2C}^- is neither zero nor infinite. Note, however, that if z_{2C}^- were either zero or infinite, then it can be shown that (B.1) would no longer hold, so we can rule out that possibility.

Therefore, as both λ_2 and z_{2C}^- are continuously differentiable at $\hat{t}$, we are justified in computing $\frac{d}{dz_{2C}^-}\left(\lambda_2 G'(z_{2C}^-)\right)$ at $\hat{t}$. We have then

$$\begin{aligned}
\frac{d}{dz_{2C}^-}\left(\lambda_2 G'(z_{2C}^-)\right) &= \frac{d}{dz_{2C}^-}\frac{C^* G'(z_{2C}^-)}{\nu_R N^* G_2(z_{2C}^-) + C G'(z_{2C}^-)} \\
&= \frac{\nu_R N^* C^* \left[G_2(z_{2C}^-)G''(z_{2C}^-) - G'(z_{2C}^-)G_2'(z_{2C}^-)\right] + \frac{dC}{dz_{2C}^-}G'(z_{2C}^-)}{(\nu_R N^* G_2(z_{2C}^-) + C G'(z_{2C}^-))^2} \\
&= \frac{\nu_R N^* C^* G''(z_{2C}^-)\left[G_2(z_{2C}^-) + z_{2C}^- G_2'(z_{2C}^-)\right] + \frac{dC}{dz_{2C}^-}G'(z_{2C}^-)}{(\nu_R N^* G_2(z_{2C}^-) + C G'(z_{2C}^-))^2}. \quad \text{(B.6)}
\end{aligned}$$

Now, for $x \in (0, \infty)$, we have that $G_2(x) > 0, G'(x) > 0$, and $G_2(x) < 0$, and so the first term in the numerator of (B.6) is negative at $\hat{t}$. Furthermore, we know that by Lemma 2.7 that z_{1C} is monotonically decreasing throughout the penultimate interval, and so by (B.3) one can show that z_{2C}^- decreases as z_{1C} decreases. This, along with the fact that at $\hat{t}$ we know that $\frac{dC}{dt} \geq 0$ means that $\frac{dC}{dz_{2C}^-} \leq 0$. Therefore, the numerator of (B.6) must be negative at $\hat{t}$, and so $\lambda_2 G'(z_{2C}^-)$ is decreasing at $\hat{t}$. This however,

contradicts the fact that, by (B.1), the condition $\lambda_2 G'(z_{2C}) = 1$ must hold on an interval with $\hat{t}$ as an endpoint. Therefore, we can conclude that the penultimate interval cannot be preceded by an interval of mixed root/shoot/fruit growth.

B.2 First Model Differential Equations Derivation

In this appendix we will derive equations (2.130) and (2.131), revisited below.

$$\frac{dz_{1C}}{dt} = \frac{N_R \nu_S \nu_R G(z_{2C}) G_2(z_{1C}) - C_S G'(z_{1C}) \left[\nu_S G_2(z_{1C}) + z_{2C} \nu_R G'(z_{1C})\right]}{G''(z_{1C}) \left[\nu_S z_{1C} - \nu_R z_{2C}\right]}$$

((2.130) revisited)

$$\frac{dz_{2C}}{dt} = \frac{N_R \nu_S G_2(z_{1C}) \left[G'(z_{2C})\right]^2 - C_S \left[G'(z_{1C})\right]^2 G'(z_{2C}) + G''(z_{1C}) G'(z_{2C}) \frac{dz_{1C}}{dt}}{G'(z_{1C}) G''(z_{2C})}$$

((2.131) revisited)

In deriving these differential equations we will use the notation $z_1 = z_{1C}$ and $z_2 = z_{2C}$ for simplicity. We begin by differentiating (2.76) and (2.77) with respect to t:

$$\lambda_1' G'(z_1) + \lambda_1 G''(z_1) \frac{dz_1}{dt} = \lambda_2' G'(z_2) + \lambda_2 G''(z_2) \frac{dz_2}{dt} \tag{B.7}$$

$$\lambda_1' \nu_S G_2(z_1) + \lambda_1 \nu_S G_2'(z_1) \frac{dz_1}{dt} = \lambda_2' \nu_R G_2(z_2) + \lambda_2 \nu_R G_2'(z_2) \frac{dz_2}{dt}. \tag{B.8}$$

Solving (B.7) for $\lambda_2 G''(z_2) \frac{dz_2}{dt}$ give us

$$\lambda_2 G''(z_2) \frac{dz_2}{dt} = \lambda_1' G'(z_1) + \lambda_1 G''(z_1) \frac{dz_1}{dt} - \lambda_2' G'(z_2). \tag{B.9}$$

Likewise, making the substitution $G_2'(z) = -zG''(z)$ by means of (2.15), we can solve for $z_2 \nu_R \lambda_2 G''(z_2) \frac{dz_2}{dt}$ in (B.8) to get

$$z_2 \nu_R \lambda_2 G''(z_2) \frac{dz_2}{dt} = \lambda_2' \nu_R G_2(z_2) - \lambda_1' \nu_S G_2(z_1) + z_1 \lambda_1 \nu_S G''(z_1) \frac{dz_1}{dt}. \tag{B.10}$$

Substituting (B.9) into (B.10) gives us

$$z_2\nu_R\left[\lambda_1' G'(z_1) + \lambda_1 G''(z_1)\frac{dz_1}{dt} - \lambda_2' G'(z_2)\right] = \lambda_2'\nu_R G_2(z_2) - \lambda_1'\nu_S G_2(z_1) + z_1\lambda_1\nu_S G''(z_1)\frac{dz_1}{dt}. \quad \text{(B.11)}$$

Solving for $\frac{dz_1}{dt}$, making use of (2.14), gives us

$$\frac{dz_1}{dt} = \frac{\lambda_1'\left[\nu_S G_2(z_1) + z_2\nu_R G'(z_1)\right] - \lambda_2'\nu_R G(z_2)}{\lambda_1 G''(z_1)\left[\nu_S z_1 - \nu_R z_2\right]}. \quad \text{(B.12)}$$

Lastly, then, we use the differential equations for λ_1 and λ_2 during balanced growth, (2.80) and (2.81), to write both λ_1' and λ_2' in terms of λ_1. Canceling the resulting common factor of λ_1 leaves us with

$$\frac{dz_1}{dt} = \frac{N_R\nu_S\nu_R G(z_2)G_2(z_1) - C_S G'(z_1)\left[\nu_S G_2(z_1) + z_2\nu_R G'(z_1)\right]}{G''(z_1)\left[\nu_S z_1 - \nu_R z_2\right]} \quad \text{(B.13)}$$

Similarly, we can use (2.80) and (2.81) along with (2.76) to simplify (B.9), which upon canceling the common factors of λ_1 gives us

$$\frac{dz_2}{dt} = \frac{N_R\nu_S G_2(z_1)\left[G'(z_2)\right]^2 - C_S\left[G'(z_1)\right]^2 G'(z_2) + G''(z_1)G'(z_2)\frac{dz_1}{dt}}{G'(z_1)G''(z_2)}. \quad \text{(B.14)}$$

B.3 Second Model Differential Equations Derivation

In this appendix we will derive equations (3.98) and (3.99), revisited below.

$$\frac{dz_2}{dt} = \frac{N_R\nu_R\left[G_2(z_2)\right]^2 G'(z_2)G(z_0)}{G''(z_2)\left[z_0 G'(z_0)G_2(z_2) - z_2 G'(z_2)G_2(z_0)\right]}. \quad \text{((3.98) revisited)}$$

$$\frac{dz_0}{dt} = \frac{G_2(z_0)}{z_0 G''(z_0)G_2(z_2)}\left[z_2 G''(z_2)\frac{dz_2}{dt} + N_R\nu_R\left[G_2(z_2)\right]^2\right] \quad \text{((3.99) revisited)}$$

For the sake of simplicity, we make the substitutions $z_0 = z_{0C}$ and $z_2 = z_{2C}$. We begin by differentiating (3.67) with respect to time, using (3.67) and (3.72) to rewrite λ_2 and λ_2', respectively.

$$\begin{aligned} G''(z_0)\frac{dz_0}{dt} &= \lambda_2' G'(z_2) + \lambda_2 G''(z_2)\frac{dz_2}{dt} \\ &= -N_R\nu_R\lambda_2 G_2(z_2)G'(z_2) + \lambda_2 G''(z_2)\frac{dz_2}{dt} \\ &= \frac{G'(z_0)}{G'(z_2)}\left[G''(z_2)\frac{dz_2}{dt} - N_R\nu_R G_2(z_2)G'(z_2)\right] \end{aligned} \tag{B.15}$$

Next, differentiating (3.68) with respect to time and using (3.68) and (3.72) to rewrite λ_2 and λ_2', respectively.

$$\begin{aligned} \frac{\nu_F}{\nu_R}G_2'(z_0)\frac{dz_0}{dt} &= \lambda_2' G_2(z_2) + \lambda_2 G_2'(z_2)\frac{dz_2}{dt} \\ &= \lambda_2\left[G_2'(z_2)\frac{dz_2}{dt} - N_R\nu_R\left[G_2(z_2)\right]^2\right] \\ &= \frac{\nu_F}{\nu_R}\frac{G_2(z_0)}{G_2(z_2)}\left[G_2'(z_2)\frac{dz_2}{dt} - N_R\nu_R\left[G_2(z_2)\right]^2\right] \end{aligned} \tag{B.16}$$

Now, by using (2.15) we can rewrite (B.16) as

$$\begin{aligned} -z_0 G''(z_0)\frac{dz_0}{dt} &= \frac{G_2(z_0)}{G_2(z_2)}\left[-z_2 G''(z_2)\frac{dz_2}{dt} - N_R\nu_R\left[G_2(z_2)\right]^2\right] \\ \implies z_0 G''(z_0)\frac{dz_0}{dt} &= \frac{G_2(z_0)}{G_2(z_2)}\left[z_2 G''(z_2)\frac{dz_2}{dt} + N_R\nu_R\left[G_2(z_2)\right]^2\right]. \end{aligned} \tag{B.17}$$

Equations (B.15) and (B.17) then give us a system of two equations that is linear in $\frac{dz_0}{dt}$ and $\frac{dz_2}{dt}$, so we can solve them to get the requisite differential equations. Noting that both equations have a factor of $G''(z_0)\frac{dz_0}{dt}$ on the left-hand side we can obtain

the following:

$$z_0 \frac{G'(z_0)}{G'(z_2)} \left[G''(z_2) \frac{dz_2}{dt} - N_R \nu_R G_2(z_2) G'(z_2) \right] = \frac{G_2(z_0)}{G_2(z_2)} \left[z_2 G''(z_2) \frac{dz_2}{dt} + N_R \nu_R \left[G_2(z_2) \right]^2 \right].$$

After some algebraic simplification, making use of (2.14), we find

$$\frac{dz_2}{dt} = \frac{N_R \nu_R \left[G_2(z_2) \right]^2 G'(z_2) G(z_0)}{G''(z_2) \left[z_0 G'(z_0) G_2(z_2) - z_2 G'(z_2) G_2(z_0) \right]}. \tag{B.18}$$

We can then substitute (B.18) into the following, simplified from (B.17):

$$\frac{dz_0}{dt} = \frac{G_2(z_0)}{z_0 G''(z_0) G_2(z_2)} \left[z_2 G''(z_2) \frac{dz_2}{dt} + N_R \nu_R \left[G_2(z_2) \right]^2 \right]. \tag{B.19}$$

APPENDIX C

MATLAB Scripts

This appendix includes the MATLAB scripts we used to implement the numerical schemes for each model. Appendix C.1 includes the MATLAB script for the numerical scheme for the first model, and Appendix C.2 includes the MATLAB script for the second model.

C.1 First Model Numerical Scheme MATLAB Script

This appendix includes the MATLAB script for the numerical scheme we developed to solve the optimal control problem (2.17) associated with the first model, as discussed in Section 2.5. The function `growthpath` is the primary function for constructing the optimal trajectory, and takes the terminal values of shoots and roots as arguments. The primary parameters for `growthpath` are given in Table C.1. All functions called by `growthpath` are included at the end of the script.

Parameter	Value	Meaning
nu_R	1	ν_R
beta	3	ν_R/ν_S
n	2^{10}	RK4 step size is $1/n$
T	10	T
BGPI_tol	10^{-10}	Tolerance for the balanced growth phase to penultimate interval transition condition (2.85)
IBG_tol	10^{-10}	Tolerance for the initial stage to balanced growth phase transition condition (2.132)

Table C.1: Primary parameters for `growthpath`.

```
%Carbon-Only Fruits Optimal Control Problem

function [t, S, R, F, U, L] = growthpath(S_Final, R_Final)
  %Terminal Conditions
  p.S_Final = S_Final;  %Final value of shoots
  p.R_Final = R_Final;  %Final value of roots

  % Primary parameters
  p.nu_R = 1; %C:N ratio in roots
  beta = 3;  %Ratio of C:N ratio in roots to C:N ratio in
     shoots
  p.n = 2^10;  %Number of grid points per unit time
  p.T = 10;     %Length of growing season
  p.BGPI_tol = 10^-10;  %Tolerance for BG-PI Hamiltonian
     condition
  p.IBG_tol = 10^-10; %Tolerance for initial phase to BG
     controls condition
```

```
% Secondary parameters
p.C_Final = C(p.S_Final); %Final rate of carbon fixation
p.N_Final = N(p.R_Final); %Final rate of nitrogen uptake
p.dCdS_Final = dCdS(p.S_Final); %Final rate of change of
    carbon fixation rate wrt shoots
p.dNdR_Final = dNdR(p.R_Final); %Final rate of change of
    nitrogen uptake rate wrt roots
p.nu_S = p.nu_R/beta; %C:N ratio in roots
p.t_star = p.T - 1/p.dCdS_Final;      %Compute t_star

% Create structure w/ terminal conditions for PI solver
  % Note that since we solve forward in z, terminal/
     initial are in reference to z, not t
p.PI.args.step = 1;  %Used to compute RK4 step size via
   h = p.PI.args.step/p.n;
p.PI.args.z_start = 0; %Starting z1C value for the PI
   solver
p.PI.args.z_end = 100;  %End z1C for PI solver, 100 is
   an arbitrary initial guess
p.PI.args.L2_init = 0;  %PI initial condition for L2
p.PI.args.S_init = p.S_Final; %PI initial condition for
   S
p.PI.args.t_init = p.t_star;  %PI initial condition for
   t
```

```
% Penultimate interval solver - first run
[tP, zP, SP, RP, UP, LP] = PI(p);

p.PI.BGPI_zwindow_begin_index = find(LP(2,:)<0.99,1,'
   last');  %Find last index where L2 < 1

% BG-PI transition point finder - first run
[tP, zP, SP, RP, UP, LP, p] = BGPI_Find(tP, zP, SP, UP,
   LP, p);


% Reorder by time - note that we don't need to reorder R
    b/c it is constant here
[~,tPsort] = sort(tP);
tP = tP(tPsort);
SP = SP(tPsort);
LP = LP(:,tPsort);
UP = UP(:,tPsort);
zP = zP(tPsort);

% Refine integration mesh and iterate until BG-PI
   transition located to within p.BGPI_tol
while p.BGPI.H > p.BGPI_tol
  [tP2, zP2, SP2, RP2, UP2, LP2] = PI(p);
  p.PI.BGPI_zwindow_begin_index = 1;
```

```
  [tP2, zP2, SP2, RP2, UP2, LP2, p] = BGPI_Find(tP2, zP2
     , SP2, UP2, LP2, p);
end

% Append value of variables at BG-PI transition point
tP = [p.PI.args.t_init, tP];
SP = [p.PI.args.S_init, SP];
RP = [p.R_Final RP];
LP = [[1/g_prime(p.PI.args.z_start);p.PI.args.L2_init],
   LP];
zP = [p.PI.args.z_start, zP];
UP = [[1-UP2(2,end); UP2(2,end); 0; 1; 0], UP];

clear SP2 RP2 LP2 UP2 tP2 zP2 %Clear unnecessary
   variables

% Create struture w/ terminal conditions for balanced
   growth solver
p.BG.args.step = 1;
p.BG.args.t_start = 0;
p.BG.args.t_end = tP(1);
p.BG.args.S_end = SP(1);
p.BG.args.R_end = p.R_Final;
p.BG.args.L1_end = LP(1,1);
p.BG.args.L2_end = LP(2,1);
```

```
p.BG.args.z1_end = zP(1);
[p.BG.args.z2_end,~,~] = fsolve(@(Z)(ZLeft(Z,zP(1),p))
   ,1,optimoptions('fsolve','MaxFunEvals',10000,'Display
   ','off','OptimalityTolerance',1e-20)); %Solve for z2C
    using necessary conditions


% Solve for fruits during PI
FP = FruitsPI(tP, SP, UP(1,:));

% Final interval
[tF, SF, RF, FF, UF, LF] = final(FP(end), p);

% Balanced growth
[tB, zB, SB, RB, UB, LB] = BG(p);

% Find beginning of balanced growth
if(tB(1) < 2/p.n)       %If U < 1 and t = 0 then there
   is no intial phase
  FB = zeros(1,length(tB));
  t = [tB, tP, tF];
  S = [SB, SP, SF];
  R = [RB, RP, RF];
  F = [FB, FP, FF];
  L = [LB, LP, LF];
```

```
  U = [UB, UP, UF];
else
  [tB, zB, SB, RB, UB, LB, p] = CBG_Find(tB, zB, SB, RB,
      UB, LB, p);  %Search for beginning of BG - first
     run
  CBG_ind = 0;
  while p.CBG.Ucondition > p.IBG_tol  %Refine
     integration mesh and iterate until beginning of BG
     found to within p.IBG_tol
    [tB2, zB2, SB2, RB2, UB2, LB2] = BG(p);
    CBG_ind = 1;
    [tB2, zB2, SB2, RB2, UB2, LB2, p] = CBG_Find(tB2,
       zB2, SB2, RB2, UB2, LB2, p);
  end
  if CBG_ind == 0 %Move on if no refinement is possible
    SB2 = nan;
    RB2 = nan;
    LB2 = [nan;nan];
    UB2 = [nan;nan;nan;nan;nan];
    tB2 = nan;
    zB2 = [nan;nan];
  end

  % Append value of variables at beginning of balanced
     growth
```

```
SB = [SB2, SB];
RB = [RB2, RB];
LB = [LB2, LB];
UB = [UB2, UB];
tB = [tB2, tB];
zB = [zB2, zB];
FB = zeros(1,length(tB));
clear SB2 RB2 LB2 UB2 tB2 zB2 %Clear unnecessary
   variables

% Compute conditions required for S-only or R-only
   initial phase
[S_only_condition, R_only_condition] =
   convergence_conditions(p);

% Initial phase
if S_only_condition < R_only_condition
  [tC, SC, RC, FC, UC, LC] = shootonly(p);
elseif R_only_condition < S_only_condition
  [tC, SC, RC, FC, UC, LC] = rootonly(p);
else
  error('could not determind C-BG transition, both
     shoot-only and root-only growth are possible')
end
```

```
    % Combine Vectors From 4 Stages
    t = [tC, tB, tP, tF];
    S = [SC, SB, SP, SF];
    R = [RC, RB, RP, RF];
    F = [FC, FB, FP, FF];
    L = [LC, LB, LP, LF];
    U = [UC, UB, UP, UF];
  end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Functions called in growthpath
% Growth stage functions

% Penultimate interval function
function [t, z, S, R, U, L] = PI(p)
  %RK4 Parameters
  h = p.PI.args.step/p.n;       %Step size
  h2 = h/2;     %Half step size for RK4
  h6 = h/6;     %h/6 for RK4 update

  %Initialize vectors for penultimate interval
  z = p.PI.args.z_start:h:p.PI.args.z_end;  %z1C
  length_z = length(z);
```

```
S = p.PI.args.S_init*ones(1,length_z);
lambda_1 = 1./g_prime(z);
lambda_2 = p.PI.args.L2_init*ones(1,length_z);  %L1
   given by necessary conditions
t = p.PI.args.t_init*ones(1,length_z);

%Solve forward in z1C via RK4
for j = 1:(length_z-1)

  [k11, k12, k13] = PIRK4(S(j)          , z(j)     , p);
  [k21, k22, k23] = PIRK4(S(j) + h2*k11, z(j) + h2, p);
  [k31, k32, k33] = PIRK4(S(j) + h2*k21, z(j) + h2, p);
  [k41, k42, k43] = PIRK4(S(j) + h*k31 , z(j) + h , p);

  S(j+1) = S(j) + h6*(k11 + 2*(k21 + k31) + k41);
  lambda_2(j+1) = lambda_2(j) + h6*(k12 + 2*(k22 + k32)
     + k42);
  t(j+1) = t(j) + h6*(k13 + 2*(k23 + k33) + k43);

  u1C = p.nu_S*p.N_Final*z(j+1)/C(S(j+1));    %Update
     u1c

  % Stop and truncate if controls are unbounded
  if u1C > 1  || u1C < 0
    if j == 1
```

```
        warning('No PI - controls unbounded immediately')
        t = nan;
        S = nan;
        R = nan;
        z = nan;
        L = nan;
        U = nan;
        break
      else
        ind = j + 1;
        t = t(1:ind);
        S = S(1:ind);
        R = p.R_Final*ones(1,ind);
        z = z(1:ind);
        L(1,:) = lambda_1(1:ind);
        L(2,:) = lambda_2(1:ind);
        u1C = p.nu_S*p.N_Final*z./C(S);
        U(1,:) = 1 - u1C;  %u0C
        U(2,:) = u1C;  %u1C
        U(3,:) = zeros(1,ind); %u2C
        U(4,:) = ones(1,ind);  %u1N
        U(5,:) = zeros(1,ind); %u2N
        break
      end
    elseif j == length_z - 1
```

```
        ind = j + 1;
        t = t(1:ind);
        S = S(1:ind);
        R = p.R_Final*ones(1,ind);
        z = z(1:ind);
        L(1,:) = lambda_1(1:ind);
        L(2,:) = lambda_2(1:ind);
        u1C = p.nu_S*p.N_Final*z./C(S);
        U(1,:) = 1 - u1C;  %u0C
        U(2,:) = u1C;  %u1C
        U(3,:) = zeros(1,ind); %u2C
        U(4,:) = ones(1,ind);  %u1N
        U(5,:) = zeros(1,ind); %u2N
        break
      end
    end
end

% BG-PI transition point finder
function [tP, zP, SP, RP, UP, LP, p] = BGPI_Find(t, z, S,
   U, L, p)
  left_index = p.PI.BGPI_zwindow_begin_index;
  right_index = length(z);
  [ZL,~,~] = fsolve(@(Z)(ZLeft(Z,z(left_index),p)),1,
     optimoptions('fsolve','MaxFunEvals',10000,'Display','
```

```
   off','OptimalityTolerance',1e-20));
[ZR,~,~] = fsolve(@(Z)(ZLeft(Z,z(right_index),p)),1,
   optimoptions('fsolve','MaxFunEvals',10000,'Display','
   off','OptimalityTolerance',1e-20));
HL = L(2,left_index) *(p.nu_R*p.N_Final*g_prime_recip(ZL
   ) + C(S(left_index))*g_prime(ZL));
HR = L(2,right_index) *(p.nu_R*p.N_Final*g_prime_recip(
   ZR) + C(S(right_index))*g_prime(ZR));
HDiffsignL = sign(HL - p.C_Final);
HDiffsignR = sign(HR - p.C_Final);

if HDiffsignL*HDiffsignR > 0
  warning('sgn(H - H*) same on both sides of potential
     BG-PI transition point ')
  stop = 1;
  SP = nan;
  RP = nan;
  LP = nan;
  UP = nan;
  tP = nan;
  zP = nan;
  return
end

while abs(right_index - left_index) > 4
```

```
    mid_index = floor((right_index + left_index)/2);
    [ZM,~,~] = fsolve(@(Z)(ZLeft(Z,z(mid_index),p)),1,
       optimoptions('fsolve','MaxFunEvals',10000,'Display'
       ,'off','OptimalityTolerance',1e-20));
    HM = L(2,mid_index) *(p.nu_R*p.N_Final*g_prime_recip(
       ZM) + C(S(mid_index))*g_prime(ZM));
    if sign(HM - p.C_Final) == HDiffsignR
      right_index = mid_index;
    else
      left_index = mid_index;
    end
  end
  U1NM = ((C(S(mid_index))./p.N_Final) - p.nu_R*ZM)./(p.
     nu_S*z(mid_index) - p.nu_R*ZM);
  U1CM = z(mid_index)*p.nu_S*U1NM*p.N_Final/C(S(mid_index)
     );
  if U1NM > 1 || U1NM < 0 || U1CM > 1 || U1CM < 0
    warning('Conrols Not Bounded At Potential BG-PI
       Boundary')
  end

  % Update variables and arguments for PI solver
  tP = t(1:left_index);
  SP = S(1:left_index);
  RP = p.R_Final*ones(1,left_index);
```

```
  LP = L(:,1:left_index);
  UP = U(:,1:left_index);
  zP = z(1:left_index);

  p.PI.args.step = p.PI.args.step/100;  %Reduce step size
     for next iteration
  p.PI.args.z_start = z(left_index);
  p.PI.args.z_end = z(right_index);
  p.PI.args.L2_init = L(2,left_index);
  p.PI.args.S_init = S(left_index);
  p.PI.args.t_init = t(left_index);
  p.BGPI.H = abs(HM - p.C_Final);       %Hamiltonian
     condition
end

% Fruits during the penultimate interval
function F = FruitsPI(t,S,u0C)
  F = zeros(1,length(t));
  hF = diff(t); %Account for non-uniform time stepping
  hF6 = hF/6; %For RK4

  % Solve forwared in time via RK4, using midpoints for
     half time-steps when necessary
  for i = 1:length(t)-1
    k1 = u0C(i)*C(S(i));
```

```
    k2 = 0.5*(u0C(i) + u0C(i+1))*(C(0.5*(S(i) + S(i+1))));
    k4 = u0C(i+1)*C(S(i+1));
    F(i+1) = F(i) + hF6(i)*(k1 + 4*k2 + k4);
  end
end

% Final interval function
function [t, S, R, F, U, L] = final(F_star,p)
  h = 1/p.n;
  t = p.t_star:h:p.T;
  length_t = length(t);
  F = C(p.S_Final)*(t - p.t_star) + F_star;
  S = p.S_Final*ones(1,length_t);
  R = p.R_Final*ones(1,length_t);
  L(1,:) = dCdS(p.S_Final)*(p.T-t);
  L(2,:) = zeros(1,length_t);
  U = zeros(5,length_t);
  U(1,:) = ones(1,length_t);
  U(4,:) = ones(1,length_t);
end

% Balanced growth function
function [t, Z, S, R, U, L] = BG(p)
  %Parameters
  h = p.BG.args.step/p.n;       %Step size
```

```
h2 = h/2;      %Half step size for RK4
h6 = h/6;      %h/6 for RK4 update

%  Verify that there is a balanced growth phase
u1N_hat = (C(p.BG.args.S_end) - (p.nu_R*N(p.BG.args.
   R_end)*p.BG.args.z2_end))/(N(p.BG.args.R_end)*(p.nu_S
   *p.BG.args.z1_end - p.nu_R*p.BG.args.z2_end));
   %Compute u1N at the end of BG
u1C_hat = (p.nu_S*N(p.BG.args.R_end)*p.BG.args.z1_end*
   u1N_hat)/C(p.BG.args.S_end); %Compute u1C at the end
   of BG

if u1N_hat > 1 || u1N_hat < 0 || u1C_hat > 1 || u1C_hat
   < 0  %Check that controls are bounded
  S = p.BG.args.S_end;
  R = p.BG.args.R_end;
  L = [p.BG.args.L1_end; p.BG.args.L2_end];
  U = [0; 1; 0; 1; 0];
  t = p.BG.args.t_end;
  Z = [p.BG.args.z1_end; p.BG.args.z2_end];
  warning('No BG - controls unbounded immediately')
  return
end

%Initialize vectors for balanced growth
```

```
t = p.BG.args.t_start:h:p.BG.args.t_end;
length_t = length(t);
S = p.BG.args.S_end*ones(1,length_t);
R = p.BG.args.R_end*ones(1,length_t);
lambda_1 = p.BG.args.L1_end*ones(1,length_t);
lambda_2 = p.BG.args.L2_end*ones(1,length_t);
z1 = p.BG.args.z1_end*ones(1,length_t);
z2 = p.BG.args.z2_end*ones(1,length_t);

% Solve backwards using RK4
for i = 1:length_t-1
  j = length_t + 1 - i;

  [k11, k12, k13, k14, k15, k16] = BGRK4(S(j)          ,
     R(j)          , lambda_1(j)          , z1(j)
     , z2(j)          , p);
  [k21, k22, k23, k24, k25, k26] = BGRK4(S(j) - h2*k11,
     R(j) - h2*k12, lambda_1(j) - h2*k13, z1(j) - h2*k15
     , z2(j) - h2*k16, p);
  [k31, k32, k33, k34, k35, k36] = BGRK4(S(j) - h2*k21,
     R(j) - h2*k22, lambda_1(j) - h2*k23, z1(j) - h2*k25
     , z2(j) - h2*k26, p);
  [k41, k42, k43, k44, k45, k46] = BGRK4(S(j) - h*k31 ,
     R(j) - h*k32 , lambda_1(j) - h*k33 , z1(j) - h*k35
     , z2(j) - h*k36 , p);
```

```
S(j-1) = S(j) - h6*(k11 + 2*(k21 + k31) + k41);
R(j-1) = R(j) - h6*(k12 + 2*(k22 + k32) + k42);
lambda_1(j-1) = lambda_1(j) - h6*(k13 + 2*(k23 + k33)
   + k43);
lambda_2(j-1) = lambda_2(j) - h6*(k14 + 2*(k24 + k34)
   + k44);
z1(j-1) = z1(j) - h6*(k15 + 2*(k25 + k35) + k45);
z2(j-1) = z2(j) - h6*(k16 + 2*(k26 + k36) + k46);

    % Compute controls at current time step
u1N = (C(S(j-1)) - (p.nu_R*N(R(j-1)))*z2(j-1))/(N(R(j
   -1))*(p.nu_S*z1(j-1) - p.nu_R*z2(j-1)));
u1C = (p.nu_S*N(R(j-1))*z1(j-1)*u1N)/C(S(j-1));

% Stop and update variables if controls become
   unbounded or if t = 0 reached
if u1N > 1  || u1N < 0 || u1C > 1  || u1C < 0 || j==2
  balgrowthindex = j-1;
  u1N = (C(S(balgrowthindex:end)) - (p.nu_R*N(R(
     balgrowthindex:end)).*z2(balgrowthindex:end))./(
     N(R(balgrowthindex:end)).*(p.nu_S*z1(
     balgrowthindex:end) - p.nu_R*z2(balgrowthindex:
     end)));
```

```
        u1C = (p.nu_S*N(R(balgrowthindex:end)).*z1(
           balgrowthindex:end).*u1N)./C(S(balgrowthindex:end
           ));
        S = S(balgrowthindex:end);
        R = R(balgrowthindex:end);
        L(1,:) = lambda_1(balgrowthindex:end);
        L(2,:) = lambda_2(balgrowthindex:end);
        U(1,:) = zeros(1,length(S)); %u0C
        U(2,:) = u1C;  %u1C
        U(3,:) = 1 - U(2,:);  %u2C
        U(4,:) = u1N;  %u1N
        U(5,:) = 1 - U(4,:);  %u2N
        Z(1,:) = z1(balgrowthindex:end);
        Z(2,:) = z2(balgrowthindex:end);
        t = t(balgrowthindex:end);
        break
      end
    end
end

% Balanced Growth Constraints on Z - used for computing
   left limit of z2C from BG
function X = ZLeft(Z2,Z1,p)
  X = (g_prime(Z2)/g_prime_recip(Z2)) - (p.nu_R/p.nu_S)*(
     g_prime(Z1)/g_prime_recip(Z1));
```

```
end

% Convergence stage to balanced growth phase transition
   point finder
function [t, z, S, R, U, L, p] = CBG_Find(t, z, S, R, U, L
   , p)
  % Check to see if transition is between first two time
     steps
  if sign(U(2,1) - U(4,1))*sign(U(2,2) - U(4,2)) < 0
    left_index = 1;
    right_index = 2;
  else  %If not, do full binary search over entire BG
     interval
    left_index = 1;
    right_index = length(t);
    ULsign = sign(U(2,left_index) - U(4,left_index));
    URsign = sign(U(2,right_index) - U(4,right_index));

    if ULsign*URsign > 0
      error('sgn(u1c - u1n) same on both sides of
         potential C-BG transition point')
    end

    while abs(right_index - left_index) > 4
      mid_index = floor((right_index + left_index)/2);
```

```
    UMsign = sign(U(2,mid_index) - U(4,mid_index));
    if UMsign == URsign
      right_index = mid_index;
    else
      left_index = mid_index;
    end
  end
end

% Update variables and terminal conditions for next
   refinement
p.BG.args.t_start = t(left_index);
S = S(right_index:end);
R = R(right_index:end);
L = L(:,right_index:end);
U = U(:,right_index:end);
t = t(right_index:end);
z = z(:,right_index:end);
p.BG.args.step = p.BG.args.step/1000;
p.BG.args.t_end = t(1);
p.BG.args.S_end = S(1);
p.BG.args.R_end = R(1);
p.BG.args.L1_end = L(1,1);
p.BG.args.L2_end = L(2,1);
p.BG.args.z1_end = z(1,1);
```

```
  p.BG.args.z2_end = z(2,1);
  p.CBG.Ucondition = abs(U(2,1) - U(4,1));
end

% Function to compute conditions for each type of initial
   phase
function [S_only_condition, R_only_condition] =
   convergence_conditions(p)
  S_only_condition = abs(p.nu_S*p.BG.args.L1_end*N(p.BG.
     args.R_end)*g(C(p.BG.args.S_end)/(p.nu_S*N(p.BG.args.
     R_end))) - p.C_Final);
  R_only_condition = abs(p.nu_R*p.BG.args.L2_end*N(p.BG.
     args.R_end)*g(C(p.BG.args.S_end)/(p.nu_R*N(p.BG.args.
     R_end))) - p.C_Final);
end

% Initial (convergence) stage
% Shoot-only growth
function [t, S, R, F, U, L] = shootonly(p)
  h = 1/p.n;
  h2 = h/2;
  h6 = h/6;

  % Define constants
  p.S.N_end = N(p.BG.args.R_end); %Initial nitrogen
```

```
p.S.dNdR_end = dNdR(p.BG.args.R_end); %Initial nitrogen
   derivative

%Initialize vectors for shoot-only growth
t = 0:h:p.BG.args.t_end;
length_t = length(t);
S = p.BG.args.S_end*ones(1,length_t);
R = p.BG.args.R_end*ones(1,length_t);
lambda_1 = p.BG.args.L1_end*ones(1,length_t);
lambda_2 = p.BG.args.L2_end*ones(1,length_t);
F = zeros(1,length_t);

% Solve backwards in time using RK4
for i = 1:length_t-1
  j = length_t + 1 - i;

  [k11, k12, k13] = ShootonlyRK4(S(j)          ,
     lambda_1(j)          , p);
  [k21, k22, k23] = ShootonlyRK4(S(j) - h2*k11,
     lambda_1(j) - h2*k12, p);
  [k31, k32, k33] = ShootonlyRK4(S(j) - h2*k21,
     lambda_1(j) - h2*k22, p);
  [k41, k42, k43] = ShootonlyRK4(S(j) - h*k31 ,
     lambda_1(j) - h*k32 , p);
```

```
    S(j-1) = S(j) - h6*(k11 + 2*(k21 + k31) + k41);
    lambda_1(j-1) = lambda_1(j) - h6*(k12 + 2*(k22 + k32)
       + k42);
    lambda_2(j-1) = lambda_2(j) - h6*(k13 + 2*(k23 + k33)
       + k43);
  end
  U = zeros(5,length_t);
  U(2,:) = ones(1,length_t);
  U(4,:) = ones(1,length_t);
  L = [lambda_1; lambda_2];
end

% Root-only growth
function [t, S, R, F, U, L] = rootonly(p)
  h = 1/p.n;
  h2 = h/2;
  h6 = h/6;

  % Define constants
  p.R.C_end = C(p.BG.args.S_end); %Initial carbon
  p.R.dCdS_end = dCdS(p.BG.args.S_end); %Initial carbon
     derivative

  %Initialize vectors for root-only growth
  t = 0:h:p.BG.args.t_end;
```

```
length_t = length(t);
S = p.BG.args.S_end*ones(1,length_t);
R = p.BG.args.R_end*ones(1,length_t);
lambda_1 = p.BG.args.L1_end*ones(1,length_t);
lambda_2 = p.BG.args.L2_end*ones(1,length_t);
F = zeros(1,length_t);

% Solve backwards in time using RK4
for i = 1:length_t-1
  j = length_t + 1 - i;

  [k11, k12, k13] = RootonlyRK4(R(j)          ,  lambda_2
     (j)          , p);
  [k21, k22, k23] = RootonlyRK4(R(j) - h2*k11,  lambda_2
     (j) - h2*k13, p);
  [k31, k32, k33] = RootonlyRK4(R(j) - h2*k21,  lambda_2
     (j) - h2*k23, p);
  [k41, k42, k43] = RootonlyRK4(R(j) - h*k31 ,  lambda_2
     (j) - h*k33 , p);

  R(j-1) = R(j) - h6*(k11 + 2*(k21 + k31) + k41);
  lambda_1(j-1) = lambda_1(j) - h6*(k12 + 2*(k22 + k32)
     + k42);
  lambda_2(j-1) = lambda_2(j) - h6*(k13 + 2*(k23 + k33)
     + k43);
```

```
  end
  U = zeros(5,length_t);
  U(3,:) = ones(1,length_t);
  U(5,:) = ones(1,length_t);
  L = [lambda_1; lambda_2];
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% RK4 Functions

% Penultimate interval RK4 function
function [k1, k2, k3] = PIRK4(S, z, p)
  k1 = (p.nu_S*p.N_Final*g(z)*g_2prime(z))/(dCdS(S)*(
     g_prime(z))^2); %S
  k2 = -(p.dNdR_Final*p.nu_S*g_prime_recip(z)*g_2prime(z))
     /(dCdS(S)*(g_prime(z))^3); %L2
  k3 = (g_2prime(z))/(dCdS(S)*(g_prime(z))^2);  %t
end

% Balanced growth RK4 function
function [k1, k2, k3, k4, k5, k6] = BGRK4(S, R, L1, z1, z2
   , p)
  u = ((C(S)/N(R)) - p.nu_R*z2)/(p.nu_S*z1 - p.nu_R*z2);
     %u_1N
```

```
  k1 = p.nu_S*u*N(R)*g(z1);  %S
  k2 = p.nu_R*(1 - u)*N(R)*g(z2);  %R
  k3 = -dCdS(S)*L1*g_prime(z1);  %l1
  k4 = -dNdR(R)*L1*p.nu_S*g_prime_recip(z1);  %l2
  k5 = (dNdR(R)*p.nu_S*p.nu_R*g(z2)*g_prime_recip(z1) -
     dCdS(S)*g_prime(z1)*(p.nu_S*g_prime_recip(z1) + z2*p.
     nu_R*g_prime(z1)))/(g_2prime(z1)*(p.nu_S*z1 - p.nu_R*
     z2)); %z1C
  k6 = (dNdR(R)*p.nu_S*g_prime_recip(z1)*(g_prime(z2))^2 -
      dCdS(S)*(g_prime(z1))^2*g_prime(z2) + g_2prime(z1)*
     g_prime(z2)*k5)/(g_prime(z1)*g_2prime(z2)); %z2C
end

% Shoots-only growth RK4 function
function [k1, k2, k3] = ShootonlyRK4(S, L1, p)
  z = C(S)/(p.nu_S*p.S.N_end);  %z1C
  k1 = p.nu_S*p.S.N_end*g(z); %S
  k2 = -dCdS(S)*L1*g_prime(z);  %L1
  k3 = -p.S.dNdR_end*p.nu_S*L1*g_prime_recip(z);  %L2
end

% Root-only Growth RK4 Function
function [k1, k2, k3] = RootonlyRK4(R, L2, p)
  z1C p.R.C_end/(p.nu_R*N(R));  %z2C
  k1 = p.nu_R*N(R)*g(z);  %R
```

```
  k2 = -p.R.dCdS_end*L2*g_prime(z); %L1
  k3 = -dNdR(R)*p.nu_R*L2*g_prime_recip(z); %L2
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Model Functions - G/C/N and Derivatives

% Carbon functions
% Note: Modify both C and dCdS together
function Carbon = C(S)  %Rate of carbon fixation for a
   given amount of shoot biomass
        a = 1;  %Proportionality constant for carbon
           production
        Carbon = a*S;   %Rate of carbon fixation is
           proportional to shoot biomass in carbon
end

function Cs = dCdS(S)   %Derivative of carbon wrt shoot
        Cs = 1;
end

% Nitrogen functions
% Note: Modify both N and dNdR together
function Nitrogen = N(R) %Rate of nitrogen assimilation
   for a given amount of root biomass
```

```
        b = 1;  %Proportionality constant
        Nitrogen = b*R; %Rate of nitrogen assimilation is
           proportional to root biomass in carbon
end

function Nr = dNdR(R)   %Derivative of nitrogen wrt root
        Nr = 1;
end

% G functions and derivatives

% G(z)
function G = g(z)
        G = z.*(1+z)./(1+z+z.^2);
end

% G'(z)
function gprime = g_prime(z)
        gprime = (1+2*z)./(1+z+z.^2).^2;
end

% G''(z)
function g2prime = g_2prime(z)
  g2prime = -(6*z*(z+1))./(z.^2 + z + 1).^3;
end
```

```
% G'(1/z)
function x = g_prime_recip(z)
  x = ((z+2).*(z.^3))./(1+z+z.^2).^2;
end

% d/dz(G'(1/z))
function x = g_prime_recip_prime(z)
  x = (6*(z.^2).*(z+1))./(1+z+z.^2).^3;
end
```

C.2 Second Model Numerical Scheme MATLAB Script

This appendix includes the MATLAB script for the numerical scheme we developed to solve the optimal control problem 3.4 associated with the first model, as discussed in Section 3.5. The function `growthpath` is the primary function for constructing the optimal trajectory, and takes the terminal values of shoots and roots as arguments. The primary parameters for `growthpath` are given in Table C.2. All functions called by `growthpath` are included at the end of the script.

Parameter	Value	Meaning
`nu_F`	$1/9$	ν_F
`nu_S`	$1/3$	ν_S
`beta`	3	ν_R/ν_S
`n`	2^{10}	RK4 step size is $1/n$
`T`	10	T
`BGPI_tol`	10^{-10}	Tolerance for the balanced growth phase to penultimate interval transition condition (3.100)
`IBG_tol`	10^{-10}	Tolerance for the initial stage to balanced growth phase transition condition (2.132)

Table C.2: Primary parameters for `growthpath`.

```
%Carbon/Nitrogen Fruits Optimal Control Problem

function [t, S, R, F, U, L] = growthpath(S_Final, R_Final)
  % Terminal Conditions
    p.S_Final = S_Final;  %Final value of shoots
    p.R_Final = R_Final;  %Final value of roots

  %Primary Parameters
    p.BGPI_tol = 10^-10;  %Tolerance for BG-PI Hamiltonian
        condition
```

```
    p.IBG_tol = 10^-10; %Tolerance for initial phase to BG
        controls condition

    % C:N Ratios
    p.nu_F = 1/9; %C:N ratio in fruits
    p.nu_S = 1/3; %C:N ratio in shoots
    p.beta = 3;  %Ratio of C:N ratio in roots to C:N ratio
        in shoots
    p.n = 2^10;  %Number of grid points per unit time
    p.T = 10;   %Length of growing season

% Secondary Parameters
    p.nu_R = p.nu_S*p.beta; %C:N ratio in roots
    p.C_Final = C(p.S_Final); %Final rate of carbon
       fixation
    p.N_Final = N(p.R_Final); %Final rate of nitrogen
       uptake
    p.dCdS_Final = dCdS(p.S_Final); %Final rate of change
       of carbon fixation rate wrt shoots
    p.dNdR_Final = dNdR(p.R_Final); %Final rate of change
       of nitrogen uptake rate wrt roots
    p.z0C_FI = p.C_Final/(p.nu_F*p.N_Final);  %z0C in the
       final interval
    p.HamCon = p.nu_F*p.N_Final*g(p.z0C_FI);  %Hamiltonian
        constant
```

```
% Find t_star, the PI-FI transition point
[x.t_star,x.c,x.z2C_PI_End,x.exitflag,x.output,stop] =
   t_star_finder(p); %Find t_star, save relevant
   information in temporary structure
  p.t_star_finder = x;  %Add structure to p
  clear x;  %Clear redundant structure
  if stop == 1
    error('Problem w/ t_star_finder')
  end

% Compute/store L1(t*) and L2(t*)
p.L1_star = p.dCdS_Final*gp(p.z0C_FI)*(p.T - p.
   t_star_finder.t_star);
p.L2_star = p.dNdR_Final*p.nu_F*g2(p.z0C_FI)*(p.T - p.
   t_star_finder.t_star);
  assert(p.L1_star > 0 & p.L2_star > 0, 'L1* and/or L2*
     nonpositive');  %Check that L1* and L2* are
     positive

% Create structure w/ terminal conditions for PI solver
p.PI.args.step = 1;  %Used to compute RK4 step size via
   h = p.PI.args.step/p.n;
p.PI.args.t_start = 0; %Starting time for the PI solver
```

```
p.PI.args.t_end = p.t_star_finder.t_star;  %End time for
    PI solver
p.PI.args.L1_end = p.L1_star;  %PI terminal condition
   for L1
p.PI.args.L2_end = p.L2_star;  %PI terminal condition
   for L2
p.PI.args.z0C_end = p.z0C_FI;  %PI terminal condition
   for Z0C
p.PI.args.z2C_end = p.t_star_finder.z2C_PI_End;  %PI
   terminal condition for Z2C
p.PI.args.R_end = p.R_Final; %PI terminal condition for
   R

% Penultimate interval solver - first run
[tP, z0CP, z2CP, SP, RP, UP, LP, stop] = PI(p);
  if stop == 1
    error('Problem w/ PI Solver')
  end

% BG-PI transition point finder - first run
[tP, z0CP, z2CP, SP, RP, UP, LP, p,stop] = BGPI_Find(tP,
    z0CP, z2CP, SP, RP, UP, LP, p);
if stop == 1
  error('Problem w/ BGPI_Find')
end
```

```
% Refine integration mesh and iterate until BG-PI
   transition located to within p.BGPI_tol
while p.BGPI.H > p.BGPI_tol
  [tP2, z0CP2, z2CP2, SP2, RP2, UP2, LP2, stop] = PI(p);

  [tP2, z0CP2, z2CP2, SP2, RP2, UP2, LP2, p, stop] =
     BGPI_Find(tP2, z0CP2, z2CP2, SP2, RP2, UP2, LP2, p)
     ;
  if stop == 1
    error('Problem w/ BGPI_Find or PI When Trying to
       Resolve BG-PI Boundary')
  end
end

clear SP2 RP2 LP2 UP2 tP2 z0CP2 z2CP2 %Clear unnecessary
    variables

% Append value of variables at BG-PI transition point
SP = [p.S_Final, SP];
RP = [p.PI.args.R_end, RP];
LP = [[p.PI.args.L1_end; p.PI.args.L2_end], LP];
tP = [p.PI.args.t_end, tP];
z0CP = [p.PI.args.z0C_end, z0CP];
z2CP = [p.PI.args.z2C_end, z2CP];
```

```
u0N_PI_start = ((p.C_Final/N(p.PI.args.R_end)) - p.nu_R*
   p.PI.args.z2C_end)/(p.nu_F*p.PI.args.z0C_end - p.nu_R
   *p.PI.args.z2C_end);  %u_0N
u0C_PI_start = (p.nu_F*N(p.PI.args.R_end)*p.PI.args.
   z0C_end*u0N_PI_start)/p.C_Final;
UP = [[u0C_PI_start; 0; 1-u0C_PI_start; u0N_PI_start; 0;
    1-u0N_PI_start], UP];
clear u0N_PI_start u0C_PI_start %Clear unnecessary
   variables

[z_BGPI,~,~] = fsolve(@(Z)(ZBGPI(Z,LP(1),LP(2),p))
   ,[1,1],optimoptions('fsolve','MaxFunEvals',10000,'
   Display','off','OptimalityTolerance',1e-20)); %
   Compute terminal values of z1C and z2C for BG using
   necessary conditions

% Create struture w/ terminal conditions for balanced
   growth solver
p.BG.args.step = 1;
p.BG.args.t_start = 0;
p.BG.args.t_end = tP(1);
p.BG.args.S_end = p.S_Final;
p.BG.args.R_end = RP(1);
p.BG.args.L1_end = LP(1,1);
p.BG.args.L2_end = LP(2,1);
```

```
p.BG.args.z1_end = z_BGPI(1);
p.BG.args.z2_end = z_BGPI(2);

% Solve for fruits during PI
FP = FruitsPI(tP, SP, RP, UP(1,:), UP(4,:), p);



% Final interval
[tF, SF, RF, FF, UF, LF] = final(FP(end), p);

% Balanced growth
[tB, zB, SB, RB, UB, LB] = BG(p);

% Find beginning of balanced growth
if(tB(1) < 2/p.n)       %If U < 1 and t = 0 then there
   is no intial phase
  FB = zeros(1,length(tB));
  t = [tB, tP, tF];
  S = [SB, SP, SF];
  R = [RB, RP, RF];
  F = [FB, FP, FF];
  L = [LB, LP, LF];
  U = [UB, UP, UF];
else
```

```
[tB, zB, SB, RB, UB, LB, p] = CBG_Find(tB, zB, SB, RB,
    UB, LB, p);  %Search for beginning of BG - first
   run
CBG_ind = 0;
while p.CBG.Ucondition > p.IBG_tol  %Refine
   integration mesh and iterate until beginning of BG
   found to within p.IBG_tol
  [tB2, zB2, SB2, RB2, UB2, LB2] = BG(p);
  CBG_ind = 1;
  [tB2, zB2, SB2, RB2, UB2, LB2, p] = CBG_Find(tB2,
     zB2, SB2, RB2, UB2, LB2, p);
end
if CBG_ind == 0 %Move on if no refinement is possible
  SB2 = nan;
  RB2 = nan;
  LB2 = [nan;nan];
  UB2 = [nan;nan;nan;nan;nan;nan];
  tB2 = nan;
  zB2 = [nan;nan];
end

% Append value of variables at beginning of balanced
   growth
SB = [SB2, SB];
RB = [RB2, RB];
```

```
LB = [LB2, LB];
UB = [UB2, UB];
tB = [tB2, tB];
zB = [zB2, zB];
FB = zeros(1,length(tB));
clear SB2 RB2 LB2 UB2 tB2 zB2 %Clear unnecessary
   variables

% Compute conditions required for S-only or R-only
   initial phase
[S_only_condition, R_only_condition] =
   convergence_conditions(p);

% Initial phase
if S_only_condition < R_only_condition
  [tC, SC, RC, FC, UC, LC] = shootonly(p);
elseif R_only_condition < S_only_condition
  [tC, SC, RC, FC, UC, LC] = rootonly(p);
else
  error('could not determind C-BG transition, both
     shoot-only and root-only growth are possible')
end

% Combine Vectors From 4 Stages
t = [tC, tB, tP, tF];
```

```
    S = [SC, SB, SP, SF];
    R = [RC, RB, RP, RF];
    F = [FC, FB, FP, FF];
    L = [LC, LB, LP, LF];
    U = [UC, UB, UP, UF];
  end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Functions called in growthpath
% Growth stage functions

% Function to locate t*
function [t_star,c,z2C,exitflag,output,stop] =
   t_star_finder(p)
  c = (p.nu_R*gp(p.z0C_FI))/(p.nu_F*g2(p.z0C_FI));
  [z2C,~,exitflag,output] = fsolve(@(z)(gp(z)./g2(z)) - c
     ,1,optimoptions('fsolve','MaxFunEvals',10000,'Display
     ','off','OptimalityTolerance',1e-20));
  t_star = p.T - 1/(p.nu_R*p.dNdR_Final*g2(z2C));

  % Check for errors
  if exitflag <=0
    warning('equation not solved')
    stop = 1;
```

```
    elseif z2C < 0
      warning('z2C_star < 0')
      stop = 1;
    elseif t_star > p.T
      warning('t_star invalid: t_star > T')
      stop = 1;
    elseif t_star < 0
      warning('t_star invalid: t_star < 0')
      stop = 1;
    else
      stop = 0;
  end
end

% Function to solve during PI
function [t, z0, z2, S, R, U, L, stop] = PI(p)

  h = p.PI.args.step/p.n;        %Step size
  h2 = h/2;      %Half step size for RK4
  h6 = h/6;      %h/6 for RK4 update
  stop = 0;

  %Initialize vectors for penultimate interval
  t = p.PI.args.t_start:h:p.PI.args.t_end;
  length_t = length(t);
```

```
S = p.S_Final*ones(1,length_t);
R = p.PI.args.R_end*ones(1,length_t);
L1 = p.PI.args.L1_end*ones(1,length_t);
L2 = p.PI.args.L2_end*ones(1,length_t);
z0 = p.PI.args.z0C_end*ones(1,length_t);
z2 = p.PI.args.z2C_end*ones(1,length_t);
u0N = ones(1,length_t);
u0C = ones(1,length_t);

% RK4 Backwards in Time
for i = 1:length_t-1
  j = length_t + 1 - i;

  [k11, k12, k13, k14, k15, u] = PIRK4(R(j)          , z0
     (j)          , z2(j)          , p);
  [k21, k22, k23, k24, k25, ~] = PIRK4(R(j) - h2*k11, z0
     (j) - h2*k15, z2(j) - h2*k14, p);
  [k31, k32, k33, k34, k35, ~] = PIRK4(R(j) - h2*k21, z0
     (j) - h2*k25, z2(j) - h2*k24, p);
  [k41, k42, k43, k44, k45, ~] = PIRK4(R(j) - h*k31 , z0
     (j) - h*k35 , z2(j) - h*k34 , p);

  % Update controls
  u0N(j) = u;
  u0C(j) = (p.nu_F*N(R(j))*z0(j)*u0N(j))/p.C_Final;
```

```
% Stop and truncate if controls are unbounded
if u0N(j) < 0 || u0N(j) > 1 || u0C(j) < 0 || u0C(j) >
   1 || R(j) < 0 || j == 2
  if i == 1
    stop = 1;
    warning('No PI - controls unbounded immediately')
    U = nan;
    L = nan;
    R = nan;
    S = nan;
    z2 = nan;
    z0 = nan;
    t = nan;
    break
  else
    ind = j+1;
    U(1,:) = u0C(ind:end);
    U(3,:) = 1 - u0C(ind:end);
    U(4,:) = u0N(ind:end);
    U(6,:) = 1 - u0N(ind:end);
    L(1,:) = L1(ind:end);
    L(2,:) = L2(ind:end);
    R = R(ind:end);
    S = S(ind:end);
```

```
        z2 = z2(ind:end);
        z0 = z0(ind:end);
        t = t(ind:end);
        break
      end
    end

    R(j-1) = R(j) - h6*(k11 + 2*(k21 + k31) + k41);
    L1(j-1) = L1(j) - h6*(k12 + 2*(k22 + k32) + k42);
    L2(j-1) = L2(j) - h6*(k13 + 2*(k23 + k33) + k43);
    z2(j-1) = z2(j) - h6*(k14 + 2*(k24 + k34) + k44);
    z0(j-1) = z0(j) - h6*(k15 + 2*(k25 + k35) + k45);

    % Error check
    if R(j-1) < 0
      warning('PI: R < 0')
      stop = 1;
      elseif min(L1(j-1),L2(j-1)) < 0
        warning('PI: L1 or L2 negative')
        stop = 1;
      elseif min(z2(j-1),z0(j-1)) < 0
        warning('PI: z0 or z2 negative')
        stop = 1;
    end
  end
```

```
end

% BG-PI transition point finder
function [tP, z0CP, z2CP, SP, RP, UP, LP, p,stop] =
    BGPI_Find(t, z0C, z2C, S, R, U, L, p)
  stop = 0; %Initialize stop
  left_index = 1;
  right_index = length(t);
  [ZL,~,~] = fsolve(@(Z)(ZBGPI(Z,L(1,left_index),L(2,
     left_index),p)),[1,1],optimoptions('fsolve','
     MaxFunEvals',10000,'Display','off','
     OptimalityTolerance',1e-20));
  [ZR,~,~] = fsolve(@(Z)(ZBGPI(Z,L(1,right_index),L(2,
     right_index),p)),[1,1],optimoptions('fsolve','
     MaxFunEvals',10000,'Display','off','
     OptimalityTolerance',1e-20));
  HL = L(1,left_index)*(p.nu_S*N(R(left_index))*g2(ZL(1))
     + C(S(left_index))*gp(ZL(1)));
  HR = L(1,right_index)*(p.nu_S*N(R(right_index))*g2(ZR(1)
     ) + C(S(right_index))*gp(ZR(1)));
  HDiffsignL = sign(HL - p.HamCon);
  HDiffsignR = sign(HR - p.HamCon);

  if HDiffsignL*HDiffsignR > 0
```

```
    warning('sgn(H - H*) same on both sides of potential
       BG-PI transition point ')
    stop = 1;
    SP = nan;
    RP = nan;
    LP = nan;
    UP = nan;
    tP = nan;
    z0CP = nan;
    z2CP = nan;
    return
  end

  while right_index - left_index > 4
    mid_index = floor((right_index + left_index)/2);
    [ZM,~,~] = fsolve(@(Z)(ZBGPI(Z,L(1,mid_index),L(2,
       mid_index),p)),[1,1],optimoptions('fsolve','
       MaxFunEvals',10000,'Display','off','
       OptimalityTolerance',1e-20));
    U1NM = (p.C_Final./(N(R(mid_index))) - p.nu_R*ZM(2))
       ./(p.nu_S*ZM(1) - p.nu_R*ZM(2));
    HM = L(1,mid_index)*(p.nu_S*N(R(mid_index))*g2(ZM(1))
       + C(S(mid_index))*gp(ZM(1)));
    if sign(HM - p.HamCon) == HDiffsignR
      right_index = mid_index;
```

```
  else
    left_index = mid_index;
  end
end
U1CM = ZM(1)*p.nu_S*U1NM*N(R(mid_index))/p.C_Final;
if U1NM > 1 || U1NM < 0 || U1CM > 1 || U1CM < 0
  warning('Conrols Not Bounded At Potential BG-PI
     Boundary')
  stop = 1;
end

% Update variables and arguments for PI solver
tP = t(right_index:end);
SP = S(right_index:end);
RP = R(right_index:end);
LP = L(:,right_index:end);
UP = U(:,right_index:end);
z0CP = z0C(right_index:end);
z2CP = z2C(right_index:end);

p.PI.args.step = p.PI.args.step/1000; %Reduce step size
   for next iteration
p.PI.args.t_start = t(left_index);
p.PI.args.t_end = t(right_index);
p.PI.args.L1_end = L(1,right_index);
```

```
  p.PI.args.L2_end = L(2,right_index);
  p.PI.args.z0C_end = (U(1,right_index)*p.C_Final)/(p.nu_F
     *U(4,right_index)*N(R(right_index)));
  p.PI.args.z2C_end = (U(3,right_index)*p.C_Final)/(p.nu_R
     *U(6,right_index)*N(R(right_index)));
  p.PI.args.R_end = R(right_index);
  p.BGPI.H = abs(HM - p.HamCon);  %Hamiltonian condition
end

% Fruits during the penultimate interval
function F = FruitsPI(t, S, R, u0C, u0N, p)  %t,S,u0C
   during the PI
  F = zeros(1,length(t));
  hFs = diff(t(1:2)); %First smaller time step
  hF = 1/p.n;
  hFs6 = hFs/6;
  hF6 = hF/6;

  % Solve forwared in time via RK4, using midpoints for
     half time-steps when necessary
  for i = 1:length(t)-1
    k1 = p.nu_F*u0N(i)*N(R(i))*g((u0C(i)*C(S(i)))/(p.nu_F*
       u0N(i)*N(R(i))));
    k2 = p.nu_F*0.5*(u0N(i) + u0N(i+1))*N(0.5*(R(i)+R(i+1)
       ))*g((0.5*(u0C(i) + u0C(i+1))*C(0.5*(S(i)+S(i+1))))
```

```
        /(p.nu_F*0.5*(u0N(i) + u0N(i+1))*N(0.5*(R(i) + R(i
        +1)))));
      k4 = p.nu_F*u0N(i+1)*N(R(i+1))*g((u0C(i+1)*C(S(i+1)))
        /(p.nu_F*u0N(i+1)*N(R(i+1))));

      if i == 1
        F(i+1) = F(i) + hFs6*(k1 + 4*k2 + k4);
      else
        F(i+1) = F(i) + hF6*(k1 + 4*k2 + k4);
      end
    end
end

% Final interval function
function [t, S, R, F, U, L] = final(F_star,p)
    h = 1/p.n;
    t = p.t_star_finder.t_star:h:p.T;
    length_t = length(t);
    F = F_star + p.nu_F*p.N_Final*g(p.z0C_FI)*(t-p.
       t_star_finder.t_star);
    S = p.S_Final*ones(1,length_t);
    R = p.R_Final*ones(1,length_t);
    L(1,:) = p.dCdS_Final*gp(p.z0C_FI)*(p.T - t);
    L(2,:) = p.dNdR_Final*p.nu_F*g2(p.z0C_FI)*(p.T-t);
    U = zeros(6,length_t);
```

```
  U(1,:) = ones(1,length_t);
  U(4,:) = ones(1,length_t);
end

% Balanced growth function
function [t, Z, S, R, U, L] = BG(p)
  %Parameters
  h = p.BG.args.step/p.n;        %Step size
  h2 = h/2;      %Half step size for RK4
  h6 = h/6;      %h/6 for RK4 update

  %  Verify that there is a balanced growth phase
  u1N_hat = (C(p.BG.args.S_end) - (p.nu_R*N(p.BG.args.
     R_end)*p.BG.args.z2_end))/(N(p.BG.args.R_end)*(p.nu_S
     *p.BG.args.z1_end - p.nu_R*p.BG.args.z2_end));
     %Compute u1N at the end of BG
  u1C_hat = (p.nu_S*N(p.BG.args.R_end)*p.BG.args.z1_end*
     u1N_hat)/C(p.BG.args.S_end); %Compute u1C at the end
     of BG

  if u1N_hat > 1 || u1N_hat < 0 || u1C_hat > 1 || u1C_hat
     < 0  %Check that controls are bounded
    S = p.BG.args.S_end;
    R = p.BG.args.R_end;
    L = [p.BG.args.L1_end; p.BG.args.L2_end];
```

```
  U = [0; 1; 0; 0; 1; 0];
  t = p.BG.args.t_end;
  Z = [p.BG.args.z1_end; p.BG.args.z2_end];
  warning('No BG - controls unbounded immediately')
  return
end

%Initialize vectors for balanced growth
t = p.BG.args.t_start:h:p.BG.args.t_end;
length_t = length(t); %length of t vector
S = p.BG.args.S_end*ones(1,length_t);
R = p.BG.args.R_end*ones(1,length_t);
lambda_1 = p.BG.args.L1_end*ones(1,length_t);
lambda_2 = p.BG.args.L2_end*ones(1,length_t);
z1 = p.BG.args.z1_end*ones(1,length_t);
z2 = p.BG.args.z2_end*ones(1,length_t);

% Solve backwards using RK4
for i = 1:length_t-1
  j = length_t + 1 - i;

  [k11, k12, k13, k14, k15, k16] = BGRK4(S(j)          ,
    R(j)          , lambda_1(j)          , z1(j)
    , z2(j)          , p);
```

```
[k21, k22, k23, k24, k25, k26] = BGRK4(S(j) - h2*k11,
   R(j) - h2*k12, lambda_1(j) - h2*k13, z1(j) - h2*k15
   , z2(j) - h2*k16, p);
[k31, k32, k33, k34, k35, k36] = BGRK4(S(j) - h2*k21,
   R(j) - h2*k22, lambda_1(j) - h2*k23, z1(j) - h2*k25
   , z2(j) - h2*k26, p);
[k41, k42, k43, k44, k45, k46] = BGRK4(S(j) - h*k31 ,
   R(j) - h*k32 , lambda_1(j) - h*k33 , z1(j) - h*k35
   , z2(j) - h*k36 , p);

S(j-1) = S(j) - h6*(k11 + 2*(k21 + k31) + k41);
R(j-1) = R(j) - h6*(k12 + 2*(k22 + k32) + k42);
lambda_1(j-1) = lambda_1(j) - h6*(k13 + 2*(k23 + k33)
   + k43);
lambda_2(j-1) = lambda_2(j) - h6*(k14 + 2*(k24 + k34)
   + k44);
z1(j-1) = z1(j) - h6*(k15 + 2*(k25 + k35) + k45);
z2(j-1) = z2(j) - h6*(k16 + 2*(k26 + k36) + k46);

% Compute controls at current time step
u1N = (C(S(j-1)) - (p.nu_R*N(R(j-1)))*z2(j-1))/(N(R(j
   -1))*(p.nu_S*z1(j-1) - p.nu_R*z2(j-1)));       %
   Update u1N
u1C = (p.nu_S*N(R(j-1))*z1(j-1)*u1N)/C(S(j-1));
```

```
% Stop and update variables if controls become
   unbounded or if t = 0 reached
if u1N > 1  || u1N < 0 || u1C > 1  || u1C < 0 || j==2
  balgrowthindex = j-1;
  u1N = (C(S(balgrowthindex:end)) - (p.nu_R*N(R(
     balgrowthindex:end)).*z2(balgrowthindex:end))./(
     N(R(balgrowthindex:end)).*(p.nu_S*z1(
     balgrowthindex:end) - p.nu_R*z2(balgrowthindex:
     end)));
  u1C = (p.nu_S*N(R(balgrowthindex:end)).*z1(
     balgrowthindex:end).*u1N)./C(S(balgrowthindex:end
     ));
  S = S(balgrowthindex:end);
  R = R(balgrowthindex:end);
  L(1,:) = lambda_1(balgrowthindex:end);
  L(2,:) = lambda_2(balgrowthindex:end);
  U(1,:) = zeros(1,length(S)); %u0C
  U(2,:) = u1C;  %u1C
  U(3,:) = 1 - U(2,:);  %u2C
  U(4,:) = U(1,:);  %u0N
  U(5,:) = u1N;  %u1N
  U(6,:) = 1 - U(5,:);  %u2N
  Z(1,:) = z1(balgrowthindex:end);
  Z(2,:) = z2(balgrowthindex:end);
  t = t(balgrowthindex:end);
```

```
      break
    end
  end
end

% Balanced Growth Constraints on Z - used for computing
   left limits of z1C and z2C from BG
function X = ZBGPI(Z,L1,L2,p)
  X(1) = L1*gp(Z(1)) - L2*gp(Z(2));
  X(2) = L1*p.nu_S*g2(Z(1)) - L2*p.nu_R*g2(Z(2));
end

% Convergence stage to balanced growth phase transition
   point finder
function [t, z, S, R, U, L, p] = CBG_Find(t, z, S, R, U, L
   , p)
  if sign(U(2,1) - U(5,1))*sign(U(2,2) - U(5,2)) < 0
    left_index = 1;
    right_index = 2;
  else  %If not, do full binary search over entire BG
     interval
    left_index = 1;
    right_index = length(t);
    ULsign = sign(U(2,left_index) - U(5,left_index));
    URsign = sign(U(2,right_index) - U(5,right_index));
```

```
  if ULsign*URsign > 0
    error('sgn(u1c - u1n) same on both sides of
       potential C-BG transition point')
  end

  while abs(right_index - left_index) > 4
    mid_index = floor((right_index + left_index)/2);
    UMsign = sign(U(2,mid_index) - U(5,mid_index));
    if UMsign == URsign
      right_index = mid_index;
    else
      left_index = mid_index;
    end
  end
end

% Update variables and terminal conditions for next
   refinement
p.BG.args.t_start = t(left_index);
S = S(right_index:end);
R = R(right_index:end);
L = L(:,right_index:end);
U = U(:,right_index:end);
t = t(right_index:end);
```

```
  z = z(:,right_index:end);
  p.BG.args.step = p.BG.args.step/1000;
  p.BG.args.t_end = t(1);
  p.BG.args.S_end = S(1);
  p.BG.args.R_end = R(1);
  p.BG.args.L1_end = L(1,1);
  p.BG.args.L2_end = L(2,1);
  p.BG.args.z1_end = z(1,1);
  p.BG.args.z2_end = z(2,1);
  p.CBG.Ucondition = abs(U(2,1) - U(5,1));
end

% Function to compute conditions for each type of initial
   phase
function [S_only_condition, R_only_condition] =
   convergence_conditions(p)
  S_only_condition = abs(p.nu_S*p.BG.args.L1_end*N(p.BG.
     args.R_end)*g(C(p.BG.args.S_end)/(p.nu_S*N(p.BG.args.
     R_end))) - p.nu_F*p.N_Final*g(p.z0C_FI));
  R_only_condition = abs(p.nu_R*p.BG.args.L2_end*N(p.BG.
     args.R_end)*g(C(p.BG.args.S_end)/(p.nu_R*N(p.BG.args.
     R_end))) - p.nu_F*p.N_Final*g(p.z0C_FI));
end

% Initial (convergence) stage
```

```
% Shoot-only growth
function [t, S, R, F, U, L] = shootonly(p)
  h = 1/p.n;
  h2 = h/2;
  h6 = h/6;

  % Define constants
  p.S.N_end = N(p.BG.args.R_end); %Initial nitrogen
  p.S.dNdR_end = dNdR(p.BG.args.R_end); %Initial nitrogen
     derivative

  %Initialize vectors for shoot-only growth
  t = 0:h:p.BG.args.t_end;
  length_t = length(t); %length of t vector
  S = p.BG.args.S_end*ones(1,length_t);
  R = p.BG.args.R_end*ones(1,length_t);
  lambda_1 = p.BG.args.L1_end*ones(1,length_t);
  lambda_2 = p.BG.args.L2_end*ones(1,length_t);
  F = zeros(1,length_t);

  % Solve backwards in time using RK4
  for i = 1:length_t-1
    j = length_t + 1 - i;
```

```
    [k11, k12, k13] = ShootonlyRK4(S(j)          ,
       lambda_1(j)          , p);
    [k21, k22, k23] = ShootonlyRK4(S(j) - h2*k11,
       lambda_1(j) - h2*k12, p);
    [k31, k32, k33] = ShootonlyRK4(S(j) - h2*k21,
       lambda_1(j) - h2*k22, p);
    [k41, k42, k43] = ShootonlyRK4(S(j) - h*k31 ,
       lambda_1(j) - h*k32 , p);

    S(j-1) = S(j) - h6*(k11 + 2*(k21 + k31) + k41);
    lambda_1(j-1) = lambda_1(j) - h6*(k12 + 2*(k22 + k32)
       + k42);
    lambda_2(j-1) = lambda_2(j) - h6*(k13 + 2*(k23 + k33)
       + k43);
  end
  U = zeros(6,length_t);
  U(2,:) = ones(1,length_t);
  U(5,:) = ones(1,length_t);
  L = [lambda_1; lambda_2];
end

% Root-only growth
function [t, S, R, F, U, L] = rootonly(p)
  h = 1/p.n;
  h2 = h/2;
```

```
h6 = h/6;

% Define constants
p.R.C_end = C(p.BG.args.S_end); %Initial carbon
p.R.dCdS_end = dCdS(p.BG.args.S_end); %Initial carbon
   derivative

%Initialize vectors for root-only growth
t = 0:h:p.BG.args.t_end;
length_t = length(t);
S = p.BG.args.S_end*ones(1,length_t);
R = p.BG.args.R_end*ones(1,length_t);
lambda_1 = p.BG.args.L1_end*ones(1,length_t);
lambda_2 = p.BG.args.L2_end*ones(1,length_t);
F = zeros(1,length_t);

% Solve backwards in time using RK4
for i = 1:length_t-1
  j = length_t + 1 - i;

  [k11, k12, k13] = RootonlyRK4(R(j)          ,  lambda_2
     (j)          , p);
  [k21, k22, k23] = RootonlyRK4(R(j) - h2*k11,  lambda_2
     (j) - h2*k13, p);
```

```
    [k31, k32, k33] = RootonlyRK4(R(j) - h2*k21,  lambda_2
       (j) - h2*k23, p);
    [k41, k42, k43] = RootonlyRK4(R(j) - h*k31 ,  lambda_2
       (j) - h*k33 , p);

    R(j-1) = R(j) - h6*(k11 + 2*(k21 + k31) + k41);
    lambda_1(j-1) = lambda_1(j) - h6*(k12 + 2*(k22 + k32)
       + k42);
    lambda_2(j-1) = lambda_2(j) - h6*(k13 + 2*(k23 + k33)
       + k43);
  end
  U = zeros(6,length_t);
  U(3,:) = ones(1,length_t);
  U(6,:) = ones(1,length_t);
  L = [lambda_1; lambda_2];
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% RK4 Functions

% Penultimate Interval RK4 Function
function [k1, k2, k3, k4, k5, u] = PIRK4(R, z0, z2, p)
  u = ((p.C_Final/N(R)) - p.nu_R*z2)/(p.nu_F*z0 - p.nu_R*
     z2);  %u_ON
```

```
  k1 = p.nu_R*(1 - u)*N(R)*g(z2);  %R
  k2 = -p.dCdS_Final*gp(z0);  %l1
  k3 = -dNdR(R)*p.nu_F*g2(z0);  %L2
  k4 = ((dNdR(R)*p.nu_R*gp(z2)*(g2(z2))^2)*g(z0))/(gpp(z2)
     *(z0*gp(z0)*g2(z2) - z2*gp(z2)*g2(z0))); %z2C
  k5 = (g2(z0)/(z0*gpp(z0)*g2(z2)))*(z2*gpp(z2)*k4 + dNdR(
     R)*p.nu_R*(g2(z2))^2); %z0C
end

% Balanced Growth RK4 Function
function [k1, k2, k3, k4, k5, k6] = BGRK4(S, R, L1, z1, z2
   , p)
  u = ((C(S)/N(R)) - p.nu_R*z2)/(p.nu_S*z1 - p.nu_R*z2);
     %u_1N

  k1 = p.nu_S*u*N(R)*g(z1);  %S
  k2 = p.nu_R*(1 - u)*N(R)*g(z2);  %R
  k3 = -dCdS(S)*L1*gp(z1);  %l1
  k4 = -dNdR(R)*L1*p.nu_S*g2(z1);  %l2
  k5 = (dNdR(R)*p.nu_S*p.nu_R*g(z2)*g2(z1) - dCdS(S)*gp(z1
     )*(p.nu_S*g2(z1) + z2*p.nu_R*gp(z1)))/(gpp(z1)*(p.
     nu_S*z1 - p.nu_R*z2)); %z1C
  k6 = (dNdR(R)*p.nu_S*g2(z1)*(gp(z2))^2 - dCdS(S)*(gp(z1)
     )^2*gp(z2) + gpp(z1)*gp(z2)*k5)/(gp(z1)*gpp(z2)); %
     z2C
```

```
end

% Shoots-only growth RK4 function
function [k1, k2, k3] = ShootonlyRK4(S, L1, p)
  z = C(S)/(p.nu_S*p.S.N_end);  %z1C
  k1 = p.nu_S*p.S.N_end*g(z); %S
  k2 = -dCdS(S)*L1*gp(z); %L1
  k3 = -p.S.dNdR_end*p.nu_S*L1*g2(z); %L2
end

% Root-only Growth RK4 Function
function [k1, k2, k3] = RootonlyRK4(R, L2, p)
  z = p.R.C_end/(p.nu_R*N(R));  %z2C
  k1 = p.nu_R*N(R)*g(z);  %R
  k2 = -p.R.dCdS_end*L2*gp(z);  %L1
  k3 = -dNdR(R)*p.nu_R*L2*g2(z);  %L2
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Model Functions - G/C/N and Derivatives
% Carbon functions
% Note: Modify both C and dCdS together
function Carbon = C(S)  %Rate of carbon fixation for a
   given amount of shoot biomass
```

```
        a = 1;  %Proportionality constant for carbon
           production
        Carbon = a*S;   %Rate of carbon fixation is
           proportional to shoot mass in carbon
end

function Cs = dCdS(S)   %Derivative of carbon wrt shoot
        Cs = 1;
end

% Nitrogen functions
% Note: Modify both N and dNdR together
function Nitrogen = N(R) %Rate of nitrogen assimilation
   for a given amount of root biomass
        b = 1;  %Proportionality constant
        Nitrogen = b*R; %Rate of nitrogen assimilation is
           proportional to root mass in carbon
end

function Nr = dNdR(R)   %Derivative of nitrogen wrt root
        Nr = 1;
end

% G Functions and Derivatives
```

```
% G(z)
function x = g(z) % G(z)
        x = z.*(1+z)./(1+z+z.^2);
end

% G'(z)
function x = gp(z)  % G'(z)
        x = (1+2*z)./(1+z+z.^2).^2;
end

% G''(z)
function x = gpp(z) % G''(z)
  x = -(6*z.*(1+z))./(1+z+z.^2).^3;
end

% G_2(z) = G'(1/z)
function x = g2(z)  % G_2(z) = G'(1/z)
  x = ((z.^3).*(2+z))./(1+z+z.^2).^2;
end

% G_2'(z)
function x = g2p(z) %G_2'(z) = -zG''(z)
  x = -z.*gpp(z);
end
```

Bibliography

[1] Arnold J Bloom, F Stuart Chapin III, and Harold A Mooney, *Resource limitation in plants-an economic analogy*, Annual review of Ecology and Systematics **16** (1985), no. 1, 363–392.

[2] Ray Dybzinski, Caroline Farrior, Adam Wolf, Peter B Reich, and Stephen W Pacala, *Evolutionarily stable strategy carbon allocation to foliage, wood, and fine roots in trees competing for light and nitrogen: an analytically tractable, individual-based model and quantitative comparisons to data*, The American Naturalist **177** (2011), no. 2, 153–166.

[3] Brian J Enquist and Karl J Niklas, *Global allocation rules for patterns of biomass partitioning in seed plants*, Science **295** (2002), no. 5559, 1517–1520.

[4] Phyllis A Hicks, *Distribution of carbon/nitrogen ratio in the various organs of the wheat plant at different periods of its life history*, New Phytologist **27** (1928), no. 2, 108–116.

[5] Yoh Iwasa and Jonathan Roughgarden, *Shoot/root balance of plants: optimal growth of a system with many vegetative organs*, Theoretical population biology **25** (1984), no. 1, 78–105.

[6] S.A.L.M. Kooijman, *Dynamic energy budget theory for metabolic organisation*, Cambridge university press, 2010.

[7] Glenn Ledder, Sabrina E. Russo, Erik B. Muller, Angela Peace, and Roger M. Nisbet, *Local control of resource allocation is sufficient to model optimal dynamics in syntrophic systems*, Theoretical Ecology (2020).

[8] Suzanne Lenhart and John T Workman, *Optimal control applied to biological models*, Chapman and Hall/CRC, 2007.

[9] MC McCarthy and BJ Enquist, *Consistency between an allometric approach and optimal partitioning theory in global patterns of plant biomass allocation*, Functional Ecology **21** (2007), no. 4, 713–720.

[10] V Minden and M Kleyer, *Internal and external regulation of plant organ stoichiometry*, Plant Biology **16** (2014), no. 5, 897–907.

[11] Lev Semenovich Pontryagin, EF Mishchenko, VG Boltyanskii, and RV Gamkrelidze, *The mathematical theory of optimal processes*, (1962).

[12] Hendrik Poorter and Oscar Nagel, *The role of biomass allocation in the growth response of plants to different levels of light, co2, nutrients and water: a quantitative review*, Functional Plant Biology **27** (2000), no. 12, 1191–1191.

[13] Hendrik Poorter, Karl J Niklas, Peter B Reich, Jacek Oleksyn, Pieter Poot, and Liesje Mommer, *Biomass allocation to leaves, stems and roots: meta-analyses of interspecific variation and environmental control*, New Phytologist **193** (2012), no. 1, 30–50.

[14] Kai Velten and Otto Richter, *Optimal root/shoot-partitioning of carbohydrates in plants*, Bulletin of Mathematical Biology **57** (1995), no. 1, 99–107.

[15] Jacob Weiner, *Allocation, plasticity and allometry in plants*, Perspectives in Plant Ecology, Evolution and Systematics **6** (2004), no. 4, 207–215.