Commitment Planning for Drawdown Vehicles in a Stochastic Setting

J. Adriazola[1]†, A. Chen [2], P.R. Kramer [3], K. Sikorski [4], S. Sturm [5], and A. Wey [6]

[1] *New Jersey Institute of Technology, Newark, USA*
[2] *University of Michigan, Ann Arbor, USA*
[3] *Rensselaer Polytechnic Institute, Troy, USA*
[4] *Goldman Sachs Asset Management, New York, USA*
[5] *Worcester Polytechnic Institute, Worcester, USA*
[6] *University of Oxford, Oxford, UK*
[7] *San Francisco State, San Francisco, USA*
[8] *East Stroudsburg University, Stroudsberg, USA*
[9] *University of California, Riverside, USA*

(*Communicated to* MIIR *on 6 September 2022*)

Study Group: 38^{th} Mathematical Problems in Industry Workshop (MPI), Worcester, USA, June 13-17 2022
Communicated by: Burt S. Tilley [5]
Industrial Partner: Goldman Sachs
Presenter: Kai Sikorski [4]
Team Members: Jimmie Adriazola [1]; Darcy Brunk [7]; Anthony Chen [2]; Peter R. Kramer [3]; Johnathan Makar [8]; Kai Sikorski [4]; Stephan Sturm [5]; Khoi Vo [9]; Arkady Wey [6]; Maxim Zyskin [6]
Industrial Sector: Finance

Key Words: Asset Management, Yale Model, Dynamic Programming, Stochastic Control

MSC2020 Codes: 90C15, 90C39, 90C40

Summary

We address the problem of how to best manage a limited partner's capital commitments to various vintages of private equity funds in order to achieve a desired target level of exposure. We adopt and extend the Yale modeling framework by accounting for dynamical noise on the fund returns with both systematic and idiosyncratic components assumed independent across vintages. Adequate models for the fund returns, including volatility, and for a general partner's procedure for capital calls and distributions are assumed to be known. The management problem is quantitatively expressed as choosing a capital commitment strategy that minimizes the mean-squared difference between the actual and target exposure, averaged both over the noise and a specified horizon of time. We computationally address this problem in three ways: choosing the entire capital commitment strategy non-adaptively at the start, optimizing a heuristic control strategy, and employing stochastic dynamical programming.

1 Introduction

Over the past several decades investments in illiquid asset classes such as venture capital, private equity, and private credit have played an increasingly large role in the portfolios of many institutional investors. For example based on S&P Global SNL Financial data, GSAM estimates that private equity accounted

† Corresponding Author (Now at the University of California Santa Barbara): `jadriazola@UCSB.edu`

for \$122bn of the assets on US Life Insurers' balance sheets as of year end 2021 compared to \$8.6bn as of year end 2006. Unlike investing in liquid asset classes such as public equity, investing in alternatives is typically implemented through commingled limited partnership (LP) funds. These structures introduce implementation challenges not present in the context of liquid public investments. As explained in Takahashi and Alexander [5]:

"The funds are raised every few years on a blind pool basis by general partners (GPs) who actively invest, manage and harvest portfolio investments. At the onset of the partnership, investors commit capital that gets drawn down over several years by the GP. The uncertain schedule of drawdowns, unknowable changes in the valuation of the partnership's investments, and unpredictable distributions of cash or securities to the limited partners combine to make it difficult to predict accurately the future value of partnership interests."

Therefore in addition to the standard problem of deciding their strategic asset allocation, when investing in alternatives the investor is also faced with the non-trivial problem of designing a multi-year commitment plan (CP) that will achieve a desired alternatives allocation. The LPs (investor clients whose financial interests are to be served) make capital commitments to the GPs of the fund, the GPs then call this capital over a period of years, invest the capital and eventually unwind the investments and distribute cash flows back to the LPs. Takahashi and Alexander [5] introduced a commitment planning model often referred to as the "Yale model". While the Yale model offers a very useful and intuitive framework for projecting cash flows of alternative assets, one drawback from which it suffers is the assumption of a constant deterministic rate of return for each alternative asset class. Under this assumption a CP can be easily designed that reaches a given target allocation for each alternatives asset class (in steady state). However in the stochastic context, where returns for each asset class are stochastic the CP may need to be dynamically adapted over time. A balance must also be sought between the complexity of the commitment plan and the expected ability of the CP to quickly ramp up to and track the desired asset allocation.

In this work, we address the question of how to best manage sequential capital commitments of an LP to various vintages of private equity funds in order to achieve a desired target level of exposure, in light of stochastic fluctuations in the returns of the funds. The key challenge is that the choice of a capital commitment to a vintage implies cash flows into and out of the private equity fund over its lifetime that are under the control of the GP managing the private equity fund but not the LP. We adopt the Yale modeling framework, but include dynamical noise on the fund returns, independent across different time periods, with both a systematic component and an idiosyncratic component which is independent across vintages. Our goal is quantitatively expressed as choosing a capital commitment strategy which minimizes the mean-square difference between the actual and target exposure, averaged both over the noise and a specified horizon of time. We develop three classes of approaches toward numerically optimizing the exposure: 1) simple planning with no feedback control, 2) optimizing a heuristic control strategy, and 3) employing stochastic dynamical programming to compute an optimal control strategy. Stochastic dynamical programming involves a sequence of averaging and optimization steps at each decision point, marching backward from the end of the time horizon. This requires the dynamics to be expressed in Markovian form and the separate computation, at each decision time, of the optimal capital commitment as a function of every state the system could be in. To make the stochastic extension of the Yale model Markovian, the state space must include the net asset values (NAV) of all live vintages as well as the previous capital commitments that still have future capital calls.

This high-dimensional setting impedes numerical computations, so we instead examine an approximation to the Yale model where only the total NAV across all vintages is tracked, and the distribution of the NAV across vintages is assumed to be in a computable steady state. With this reduction and the specification that all capital calls occur in the first three years of a fund, the state space is three-dimensional and we are able to compute optimal control strategies over a long time horizon.

Our work is organized as follows. In Section 2, we review the Yale model and discuss extensions which account for various vintages and dynamical noise. In Section 3, we study possible steady state behaviors of the Yale model which have important implications in terms of our analytical modelling and numerical computations carried out in later sections. In Section 4, we introduce a simplifying approximation for the

Symbol	Name	Description
$\text{PIC}_{(t)}$	Paid in capital	Sum of capital contributions up to end of year t (\$)
$C_{(t)}$	Capital contributions	Capital called by the GP (\$) in year t
$D_{(t)}$	Capital distributions	Capital distributed by the GP (\$) in year t
$\text{RD}_{(t)}$	Rate of distribution	Rate of capital distributions in year t (%)
$\text{NAV}_{(t)}$	Net Asset Value	Market value of the fund assets (\$) at end of year t
$\text{RC}_{(t)}$	Rate of contribution	Rate of capital calls by the GP in year t (%)
$\text{CC}_{(t)}$	Capital commitments (\$)	LP's total commitment to GP (\$)
L	Life of fund	Total number of years fund is active (years)
B	Bow	Factor describing changes in $\text{RD}_{(\text{t})}$
G	Annual Growth Rate (%)	Assumed rate of return
Y	Yield (%)	Only applicable for yield focused asset classes

Table 1. The various symbols appearing in (2.1) which define the Yale model.

NAV's dynamics, which follow from steady state analysis and are employed in our optimization strategies. This uses the assumption that the distribution of the NAV across vintages follows a steady-state distribution which neglects fluctuations in return rates and capital commitments. In Section 5, we provide a precise formulation of the optimization problem considered in this work, and in Section 6, discuss our numerical methods for optimizing capital commitments in order to achieve a desired target exposure.

2 Model Systems

2.1 The Yale Model

In this section, we review the Yale model (see for example [5] for a more thorough discussion) and provide some basic insight into its mechanisms. The Yale model is a discrete-time dynamical system, with discrete time t measured in years, which governs the five following variables: the paid in capital (PIC), capital contribution (C), capital distribution (D), rate of distribution (RD), and NAV. The dynamics, according to the Yale model, are given by

$$\text{PIC}_{(t)} = \sum_{i=0}^{t} C_{(i)} \tag{2.1a}$$

$$C_{(t)} = \text{RC}_{(t)} \left(\text{CC}_{(t)} - \text{PIC}_{(t-1)}\right) \tag{2.1b}$$

$$D_{(t)} = \text{RD}_{(t)} \left(\text{NAV}_{(t-1)}(1+G)\right) \tag{2.1c}$$

$$\text{RD}_{(t)} = \max\left(Y, (t/L)^B\right) \tag{2.1d}$$

$$\text{NAV}_{(t)} = \left(\text{NAV}_{(t-1)}(1+G)\right) + C_{(t)} - D_{(t)}. \tag{2.1e}$$

The subscripted t is used to emphasize which variables are time-dependent and which are not. Observe that the Yale model system (2.1) contains several variables and parameters. For this reason, we organize the various symbols defining this system in Table 1 by providing each symbol's name along with a brief description of its role in the financial model.

In order to gain traction with understanding the dynamics of the Yale model, we make the following assumptions on the model's parameters. We assume $t = 0$ is the first year of the investment, use a bow of $B = 4$, assume an annual growth rate of $G = .13$ (a rather conservative estimate for private equity), set the fund's lifetime to $L = 10$, and, for simplicity, assume there is no yield, i.e., $Y = 0$. We use, as an initial condition on the net asset value, $\text{NAV}_{(-1)} = 0$, i.e., there is no initial investment; the end of year -1 is the start of our investment horizon. We also assume that the rate of the capital contribution $CC_{(t)}$ is constant (we assume that $CC_{(t)} = 1$ for each year t) and assume that we start with an initial investment period of 3

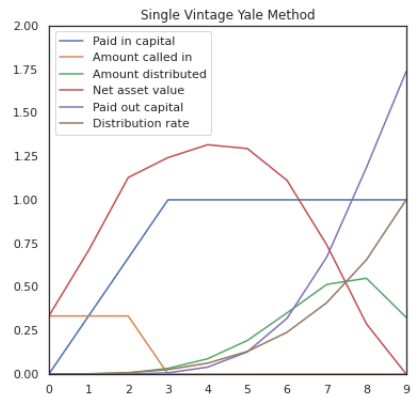

Figure 1. An example of the Yale Model's dynamics (2.1) using the assumptions described in the text near Table 1.

years, where $\frac{1}{3}$ of the total capital commitment is called in each year. Observe that this entails

$$\begin{aligned}\mathrm{RC}_{(0)} &= \frac{1}{3}\\ \mathrm{RC}_{(1)} &= \frac{\frac{1}{3}}{\frac{2}{3}} = \frac{1}{2}\\ \mathrm{RC}_{(2)} &= 1\\ \mathrm{RC}_{(n)} &= 0 \text{ for } n \geq 3\end{aligned}$$

With these assumptions in place, the first few iterations of (2.1a)—(2.1e) are

$$\begin{aligned}&\mathrm{PIC}_{(0)} = \frac{1}{3}, \quad C_{(0)} = \frac{1}{3}, \quad D_{(0)} = 0, \quad &&\mathrm{RD}_{(0)} = 0, \quad \mathrm{NAV}_{(0)} = \frac{1}{3},\\ &\mathrm{PIC}_{(1)} = \frac{2}{3}, \quad C_{(1)} = \frac{1}{3}, \quad D_{(1)} \approx 3.76 \times 10^{-5}, \quad &&\mathrm{RD}_{(1)} = 10^{-4}, \quad \mathrm{NAV}_{(1)} \approx 0.71,\\ &\mathrm{PIC}_{(2)} = 1, \quad C_{(2)} = \frac{1}{3}, \quad D(2) \approx 0.0013, \quad &&\mathrm{RD}_{(2)} = 5^{-4}, \quad \mathrm{NAV}_{(2)} \approx 1.13.\end{aligned}$$

Iterating further, we find $\mathrm{NAV}_{(L)} = 0$, that is, the investment has liquidated. We show, in Figure 1, the evolution of the Yale model, under the given assumptions.

2.2 The Extended Yale Model

We now consider two extensions of the Yale model. The first removes the assumption of a single vintage. The second introduces risk, converting the dynamics from deterministic to stochastic. Recall that our main objective is maintain a constant exposure, namely that we desire a $\mathrm{NAV}_{(t)}$ as close to a target value of T over a specified range of years. Both extensions of the Yale model we consider here introduces several challenges when attempting to formulate and solve this optimization problem.

Consider the Yale model with multiple vintages where each vintage v gives one capital commitment $\mathrm{CC}_{(t)}^{(v)}$, i.e., the amount of money committed to the vintage v at time t. Note that the variable $\mathrm{NAV}_{(t)}$ now accounts for the total net asset value over all of the vintages. We first restrict the strategy for capital commitments to a fixed, one degree of freedom, i.e., $\mathrm{CC}^{(v)} = \mathrm{CC}$ for each vintage. Although this plan for capital commitments has been idealized, the ensuing dynamics are instructive. In fact, this strategy leads to a steady state exposure since every vintage now has identical behavior. Extending the time horizon to 60 years, and introducing 50 vintages, we observe this steady state behavior shown in Figure 2. We find a steady state NAV of approximately 8.166. We also provide how each vintage contributes to the NAV over the first 12 years of the time horizon in Figure 3.

We now consider the second extension of the Yale model which introduces stochasticity in the form of risk.

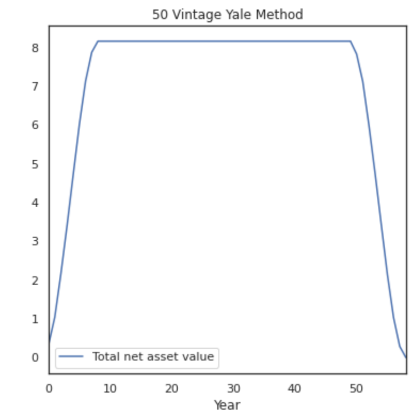

Figure 2. Behavior of the total net asset value for 50 vintages with conventions consistent with that of Figure 1 and discussion in Section 2.1.

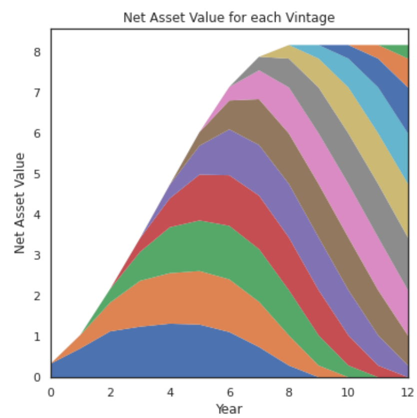

Figure 3. Behavior of individual vintages with all conventions and parameters consistent with Figure 1. In the steady state, 4.1% of the target is taken up by the most recent vintage, 8.7% by the second most recent vintage, and then 13.8%, 15.2%, 16.1%, 15.9%, 13.6%, 9.1%, and 3.5% for the oldest active vintage.

Assume that the growth rate for each vintage at a given time can be expressed in the following way:

$$G_t^{(v)} = S_t + I_t^{(v)}, \tag{2.2}$$

where S_t is systematic risk that is inherent to the markets, and $I_t^{(v)}$ is idiosyncratic risk that varies by the vintage, representing features such as the behavior of the fund's managers. We assume that both S_t and $I_t^{(v)}$ have a mean of 0.065 so $G_t^{(v)}$ has a mean of 0.13. We start by assuming that both S_t and $I_t^{(v)}$ are normally distributed and independent across their indices. We would like the standard deviation of $G_t^{(v)}$ to be 0.2, and we distribute the randomness evenly between systematic and idiosyncratic risk, so each has standard deviation $0.2/\sqrt{2} \approx 0.14$. Thus we take $S_t \sim N(0.065, 0.02)$, $I_t^{(v)} \sim N(0.0065, 0.02)$ and independent, where $N(\mu, \sigma^2)$ denotes the distribution of a Gaussian (normal) random variable with mean μ and variance σ^2.

The dynamical equations now extended to account for vintages and stochastic growth become:

$$\mathrm{PIC}_{(t)}^{(v)} = \sum_{i=v}^{t} C_{(i)}^{(v)} \tag{2.3a}$$

$$C_{(t)}^{(v)} = \mathrm{RC}_{(t-v)} \left(\mathrm{CC}^{(v)} - \mathrm{PIC}_{(t-1)}^{(v)} \right) \tag{2.3b}$$

$$D_{(t)}^{(v)} = \mathrm{RD}_{(t-v)} \left(\mathrm{NAV}_{(t-1)}^{(v)} \left(1 + G_t^{(v)} \right) \right) \tag{2.3c}$$

$$\mathrm{RD}_{(\tau)} = \max \left(Y, (\tau/L)^B \right) \tag{2.3d}$$

$$\mathrm{NAV}_{(t)}^{(v)} = \left(\mathrm{NAV}_{(t-1)}^{(v)} \left(1 + G_t^{(v)} \right) \right) + C_{(t)}^{(v)} - D_{(t)}^{(v)} \tag{2.3e}$$

The total exposure at time t to private equities is:

$$\mathrm{NAV}_{(t)} = \sum_{v=0}^{t} \mathrm{NAV}_{(t)}^{(v)}, \tag{2.4}$$

though we note $\mathrm{NAV}_{(t)}^{(v)} = 0$ when $t \geq v + L$ because $\mathrm{RD}_{(L)} = 1$. Since the system given by (2.3) is high-dimensional, we seek to make informed decisions on reducing the dynamics in following sections. Before that, in the next section, we seek to further understand the steady state behavior of the extended Yale model, such as what is shown for the deterministic case in Figure 2, and how this may help with constructing commitment strategies.

3 Forward Solutions of the Extended Yale Model

Before proceeding with the optimal allocation question, we develop a few analytical results regarding the extended Yale model that can be useful. First of all, in Subsection 3.1, we express the extended Yale model in a more concise way in terms of just the net asset value of each given fund vintage. As we show in Subsection 3.2, this allows us in particular to obtain an analytical expression for the steady-state distribution of NAVs across vintages when a constant capital commitment is allocated to each vintage. This solution is generalized to variable capital commitments and stochastic returns in Subsection 3.3. Finally, in Subsection 3.4, we point out that when trying to approximately model stochastic fund returns with a deterministic model, a geometric mean growth rate would appear to be more appropriate than the arithmetic mean.

3.1 An Obvious Simplification

We observe that Equations (2.3a)-(2.3b) can be rewritten to a closed update equation for $\mathrm{PIC}_{(t)}^{(v)}$, which we write backwards in time:

$$\mathrm{PIC}_{(t-1)}^{(v)} = \frac{\mathrm{PIC}_{(t)}^{(v)} - \mathrm{RC}_{(t-v)}\mathrm{CC}^{(v)}}{1 - \mathrm{RC}_{(t-v)}}.$$

Assuming $\mathrm{RC}_{(L-1)} = 0$ and $\mathrm{PIC}_{(v+L-1)}^{(v)} = \mathrm{CC}^{(v)}$, we can solve backwards from $t = v + L - 1$. This gives an explicit, yet cumbersome expression. An important implication, though, is that one may alternatively model the called capital as:

$$C_{(t)}^{(v)} = \gamma_{(t-v)}\, \mathrm{CC}^{(v)} \tag{3.1}$$

where $\gamma_{(\tau)}$ denotes the fraction of the capital commitment called at the end of year τ of a given vintage's life. Naturally $\sum_{\tau=0}^{L-1} \gamma_{(\tau)} = 1$. Moreover, $\gamma_{(\tau)}$ is some function of $\{\mathrm{RC}_{(\tau')}\}_{\tau'=0}^{\tau}$ whose explicit expression can, in principle, be computed. The expression (3.1) is in any case an equivalent and more transparent model for capital calls over Eqs. (2.3a)- (2.3b). We will typically take the model $\gamma_{(0)} = \gamma_{(1)} = \gamma_{(2)} = 1/3$, $\gamma_{(\tau)} = 0$ for $\tau \geq 3$.

This now allows (2.3e) to be rewritten in the form

$$\mathrm{NAV}_{(t)}^{(v)} = \left(1 - \mathrm{RD}_{(t-v)}\right)\mathrm{NAV}_{(t-1)}^{(v)}(1 + G_t^{(v)}) + \mathrm{CC}^{(v)}\, \gamma_{(t-v)}. \tag{3.2}$$

This can be solved recursively forward from $\mathrm{NAV}^{(v)}_{(v-1)} = 0$. Although we do not provide an explicit formula for this end result, we interpret the structure of the final expression computed through these means as that of a random Green's function applied to the forcing from the capital contributions.

3.2 Deterministic Steady State Distribution Across Vintages

When the capital commitments to each vintage are constant $\mathrm{CC}^{(v)} \equiv \mathrm{CC}$, and the fund growth factors $G^{(v)}_t \equiv G$ are taken as deterministic, we can solve (3.2) for the steady state NAV of fund vintages based on their age τ:

$$\overline{\mathrm{NAV}}_{(\tau)} = (1 - \mathrm{RD}_{(\tau)})\,\overline{\mathrm{NAV}}_{(\tau-1)}(1+G) + \mathrm{CC}\,\gamma_{(\tau)} \tag{3.3}$$

where $\overline{\mathrm{NAV}}_{(\tau)} = \mathrm{NAV}^{(t-\tau)}_{(t)}$ for $\tau = 0, \ldots, L-1$. Starting from $\overline{\mathrm{NAV}}_{(-1)} = 0$ and iterating forward,

$$\overline{\mathrm{NAV}}_{(\tau)} = \mathrm{CC} \sum_{\tau'=0}^{\tau} \gamma_{(\tau')}(1+G)^{\tau-\tau'} \prod_{\tau''=\tau'+1}^{\tau} (1 - \mathrm{RD}_{(\tau'')}) \text{ for } \tau = 0, 1, 2, \ldots, L-1. \tag{3.4}$$

Recall $\mathrm{RD}_{(\tau)}$ is explicitly given in (2.3c).

3.3 General Forward Solution in Stochastic Setting

In a stochastic environment with growth rates $G^{(v)}_{(t)} > -1$ independent in t, we find by iterating (3.2):

$$\mathrm{NAV}^{(v)}_{(t+s)} = \mathrm{NAV}^{(v)}_{(t)} \prod_{j=0}^{s-1}\big(1 - \mathrm{RD}_{(t+s-v-j)}\big) \prod_{j=1}^{s}\big(1 + G^{(v)}_{(t+j)}\big) \tag{3.5}$$

$$+ \sum_{k=1}^{s} \mathrm{CC}^{(v)}\,\gamma_{(t+k-v)} \prod_{j=k}^{s-1}\big(1 - \mathrm{RD}_{(t+s-v-j)}\big) \prod_{j=k+1}^{s}\big(1 + G^{(v)}_{(t+j)}\big) \tag{3.6}$$

$$\text{for } v \le t \le t+s \le v+L-1$$

By choosing $t = v-1$, the first term drops out as $\mathrm{NAV}^{(v)}_{(v-1)} = 0$.

3.4 Deterministic Growth Rate Modeling Choice

Computing expectation values in (3.5), we find expected growth factors $\mathbb{E}\big[\prod_{j=k+1}^{s}\big(1+G^{(v)}_{(t+j)}\big)\big] = \prod_{j=k+1}^{s}\big(1+\mathbb{E}\big[G^{(v)}_{(t+j)}\big]\big)$ due to independence of the growth rates, and this would match the expected value of the deterministic model (3.3) if we choose the deterministic growth rate as the standard mean $G = \mathbb{E}[G^{(v)}_{(t)}]$. On the other hand, the law of large numbers applied to the log of the product implies that $\prod_{j=k+1}^{s}\big(1+G^{(v)}_{(t+j)}\big) \to \big(1+\hat{G}\big)^{s-k}$ where $\hat{G}$ is a *geometric* average growth rate, i.e.,

$$\hat{G} = \exp\Big\{\Big(\mathbb{E}\Big[\log\Big(1 + G^{(v)}_{(t+j)}\Big)\Big]\Big)\Big\} - 1 \tag{3.7}$$

Actually this convergence of the product is valid when both sides are raised to the $1/(s-k)$ power, but the idea is that the product does not behave like its mean, and rather follows a slower growth rate $\hat{G}$ with stochastic fluctuations. The justification for this difference is due to the wide variability of trajectories, as is elaborated by elementary stochastic linear growth models. The inequality $\hat{G} \le \mathbb{E}[G^{(v)}_{(t)}]$ follows from the geometric-arithmetic mean inequality (a special case of Jensen's inequality), and is strict in the presence of true randomness. In our numerical experiments, we find using a geometric mean growth rate $\hat{G}$ in approximations involving stochastic returns gives better behavior of the control. That is, running a commitment strategy using (3.4) to target $T = \sum_{\tau=0}^{L-1} \overline{\mathrm{NAV}}_{(\tau)}$ with growth rate $G = \hat{G}$ from (3.7) yields a strategy that on average is at the right target even in the presence of stochastic fluctuations in return. If instead $\mathbb{E}\big[G^{(v)}_{(t)}\big]$ were used, the overestimation of the effective growth rate tends to lead to a behavior of the exposure that undershoots the target level T. If the NAV is quite off target (as the investment is still ramping up or there is a shock in the random growth rate which puts the investment off target), then naturally some time is required to return to the steady state as the vintages off-target have to be "phased out."

4 Simplifying Assumptions and Dynamical Reduction of the Extended Yale Model

For the nonequilibrium dynamics of our actual concern, even with the simplification in Subsection 3.1, we have to track the NAV of L active vintages $\{\mathrm{NAV}_{(t)}^{(v)}\}_{v=t-L+1}^{t}$ and the still active capital commitments $\{\mathrm{CC}^{(v)}\}_{v=t-N_{\mathrm{C}}+2}^{t}$, where $N_{\mathrm{C}} = 1 + \arg\max\{\tau : \gamma_{(\tau)} > 0\}$ is the duration of the inflow of capital into a vintage given the capital call protocol. Thus, the dynamics of the private equity investment requires a $L + N_{\mathrm{C}} - 1$ dimensional state space to be Markovian. This makes systematic control procedures computationally challenging, so we propose in Subsection 4.1 a modeling assumption in order to reduce the state space dimensionality to just N_{C} variables by compressing the information about how the investment is distributed across various vintages. Within this approximate framework, we show in Subsection 4.2 how to compute the first two moments of the NAV under fluctuations in the fund returns.

4.1 Dimension Reduction of Dynamics Through Quasi-Steady State Vintage Distribution

We could think to approximate the distribution of NAV across currently held vintages by the steady state distribution (3.4) under a constant capital commitment strategy into funds which always grow at the geometric mean $1 + \hat{G}$ growth rate (3.7), which could just as well be obtained from the numerical simulations of the deterministic system pursued in Section 2.2 . Thus, we approximate

$$\begin{pmatrix} \mathrm{NAV}_{(t)}^{(t-L+1)} \\ \mathrm{NAV}_{(t)}^{(t-L+2)} \\ \vdots \\ \mathrm{NAV}_{(t)}^{(t)} \end{pmatrix} \approx \mathrm{NAV}_{(t)} \begin{pmatrix} \phi_{(t)}^{(t-L+1)} \\ \phi_{(t)}^{(t-L+2)} \\ \vdots \\ \phi_{(t)}^{(t)} \end{pmatrix} \tag{4.1}$$

where $\phi_{(t)}^{(v)}$ denotes the fraction of $\mathrm{NAV}_{(t)}$ that is invested in vintage v under the deterministic computation. Note that $\sum_{v=t-L+1}^{t} \phi_{(t)}^{(v)} = 1$, and we expect for $t > L$ that $\phi_{(t)}^{(v)}$ only depends on $t - v$. In that case, we write $\overline{\phi}_{\tau} = \phi_{(t-\tau)}^{(t)}$.

If we first take only systematic risk ($G_t^{(v)} = S_t$), and the commitment plan $C_{(v)}^{(v)} = C_{(v+1)}^{(v)} = C_{(v+2)}^{(v)} = 1/3$ (so in particular $N_{\mathrm{C}} = 2$), then the actual dynamics can be expressed in the following non-closed form:

$$\mathrm{NAV}_{(t+1)} = \mathrm{NAV}_{(t)}(1 + S_{t+1}) + \frac{1}{3} \sum_{v=t-1}^{t+1} \mathrm{CC}^{(v)} - \sum_{v=t-L+1}^{t} \mathrm{RD}_{(t+1-v)} \, \mathrm{NAV}_{(t)}^{(v)} (1 + S_{t+1}) \tag{4.2}$$

Using the approximation (4.1), we can simplify this to:

$$\mathrm{NAV}_{(t+1)} \approx \mathrm{NAV}_{(t)}(1 + S_{t+1})(1 - \delta_{t+1}) + \frac{1}{3} \sum_{v=t-1}^{t+1} \mathrm{CC}^{(v)}$$

where

$$\delta_{t+1} = \sum_{v=t-L+1}^{t} \mathrm{RD}_{(t+1-v)} \, \phi_{(t)}^{(v)}$$

is the effective distribution rate in year $t + 1$ of the total NAV based on the empirical distribution across vintages. If we take a steady-state distribution across vintages, then we have more simply:

$$\delta_{t+1} = \delta \equiv \sum_{\tau=0}^{L-1} \mathrm{RD}_{(\tau+1)} \, \overline{\phi}_{\tau}.$$

We further allow for idiosyncratic risk by modifying (4.2) as follows:

$$\mathrm{NAV}_{(t+1)} = \mathrm{NAV}_{(t)}(1+S_{t+1}) + \frac{1}{3}\sum_{v=t-1}^{t+1} \mathrm{CC}^{(v)} + \sum_{v=t-L+1}^{t} I_{t+1}^{(v)}\,\mathrm{NAV}_{(t)}^{(v)} \tag{4.3}$$

$$- \sum_{v=t-L+1}^{t} \mathrm{RD}_{(t+1-v)}\,\mathrm{NAV}_{(t)}^{(v)}\left(1+S_{t+1}+I_{t+1}^{(v)}\right) \tag{4.4}$$

$$\Longrightarrow \mathrm{NAV}_{(t+1)} \approx \mathrm{NAV}_{(t)} \sum_{v=t-L+1}^{t} \phi_{(t)}^{(v)}(1+S_{t+1}+I_{t+1}^{(v)})(1-RD_{(t+1-v)}) + \frac{1}{3}\sum_{v=t-1}^{t+1}\mathrm{CC}^{(v)} \tag{4.5}$$

$$= \mathrm{NAV}_{(t)}(1+S_{t+1}+\iota_{t+1})(1-\delta_{(t+1)}) + \frac{1}{3}\sum_{v=t-1}^{t+1}\mathrm{CC}^{(v)} \tag{4.6}$$

where

$$\iota_{t+1} = \sum_{v=t-L+1}^{t} \phi_{(t)}^{(v)} I_{t+1}^{(v)} \frac{1-\mathrm{RD}_{(t+1-v)}}{1-\delta_{(t+1)}}.$$

Since $\{I_t^{(v)}\}_{v=t+L-1}^{t}$ are independent, identically distributed random variables, we can in principle compute the distribution of ι_t and treat it as a stream of independent effective random variables arising from idiosyncratic returns from various vintages. Thus, the overall NAV variable update becomes modified by the addition of the effects of the vintage-idiosyncratic noise:

$$\mathrm{NAV}_{(t+1)} = \mathrm{NAV}_{(t)}(1+S_{t+1}+\iota_{t+1})(1-\delta_{(t+1)}) + \frac{1}{3}\sum_{v=t-1}^{t+1}\mathrm{CC}^{(v)}$$

With the Gaussian models from Section 2.2, we have $S_t \sim N(0.065, 0.02)$ and $\iota_t \sim N(0.065, \sigma_\iota^2)$ with

$$\sigma_\iota^2 = \frac{0.02}{(1-\delta_{(t+1)})^2} \sum_{v=t-L+1}^{t} (\phi_{(t)}^{(v)})^2 (1-\mathrm{RD}_{(t+1-v)})^2.$$

Under the assumption of a steady-state vintage distribution, we have more simply:

$$\sigma_\iota^2 = \frac{0.02(\overline{\phi}_\tau)^2}{(1-\delta)^2} \sum_{\tau=0}^{L-1} (1-\mathrm{RD}_{(\tau+1)})^2.$$

These reduced dynamics and assumptions are what will be used in our backward programming strategy undertaken in Section 6.3

4.2 Approximate Statistical Variations in the Net Asset Value

As a side product of the approximate NAV dynamics (4.6), we attempt to make use of it in order to compute the long-time mean and variance of NAV assuming the capital commitments $\mathrm{CC}^{(v)}$ have achieved a statistically stationary process (with upward and downward fluctuations responding to NAV fluctuations). Taking expectations, assuming a statistical steady state for NAV and CC, and noting the independence of the noise at year $t+1$ from previous randomness, we have:

$$\mathbb{E}[\mathrm{NAV}] = \mathbb{E}[\mathrm{NAV}](1+\bar{G})(1-\delta) + \mathbb{E}[\mathrm{CC}] \tag{4.7}$$

where $\bar{G}$ is the arithmetic mean of the growth rate. The arithmetic mean is relevant for computing statistical averages, while the geometric mean, as explained in Subsection 3.4, is relevant for the statistical convergence.

Solving (4.7) for $\mathbb{E}[\mathrm{NAV}]$, we have

$$\mathbb{E}[\mathrm{NAV}] \approx \frac{\mathbb{E}[\mathrm{CC}]}{-\bar{G}+\delta+\bar{G}\delta}. \tag{4.8}$$

Note the denominator is not guaranteed to be positive since the effective distribution rate may be smaller than the mean growth rate. This would imply an inconsistency in the approximate reduction in terms of distribution across vintages. However, the actual distribution rate does eventually grow to a substantially large value, and if the recent vintages are shedding distributions at a small rate relative to the growth, the more seasoned vintages must be carrying a relatively large proportion of the NAV and thus contribute their larger distribution rates substantially to δ.

Computing the variability of NAV is complicated since it is necessary to account for correlations between the NAV and the various capital commitments CC when considering the second moment of (4.6). Thus, we need to specify the strategy for the capital commitments in order to estimate the variability of the NAV. If this strategy can be simply expressed (such as the naive control strategies explored in Subsection 6.1), then one could attempt to compute the second moment of the NAV by taking the second moment of (4.6), after substituting the capital commitment strategy.

5 Preliminary Considerations for General Strategies for Controlling Exposure

5.1 Precise and General Formulation of the Control Problem

As a first step, we consider fixing a commitment plan for the next H years to optimize exposure to our desired target. This amounts to computing:

$$\underset{\substack{\mathrm{CC}^{(0)}\geq 0,\ldots,\\ \mathrm{CC}^{(H-1)}\geq 0}}{\arg\min}\; U, \quad \text{with } U \equiv \mathbb{E}\left[\sum_{t=0}^{H-1}(\mathrm{NAV}_{(t)} - T)^2\right] \tag{5.1}$$

Here we are allowing the commitments to various vintages to vary, but one could simplify by applying the constraint $\mathrm{CC}^{(0)} = \mathrm{CC}^{(1)} = \ldots \mathrm{CC}^{(H-1)} \equiv \mathrm{CC}$. Using the solution formula from Section 4, we can write

$$U = \sum_{v,v'=0}^{H-1} B^{(v,v')}\mathrm{CC}^{(v)}\mathrm{CC}^{(v')} + \sum_{v=0}^{H} a^{(v)}\mathrm{CC}^{(v)} + K$$

where B is a non-negative definite matrix, $\vec{a}$ is a vector, and K is an irrelevant constant, each of which has an explicit expression (given by symbolic computation) in terms of the mean and standard deviations of the components of the growth rates.

However, this is not the problem we seek to resolve; rather at every time t, we take $G^{(v)}_{(t')}$ for t' as observed, and we want to optimize future target exposure by a good choices of $\{\mathrm{CC}^{(v)}\}_{v=t-L+2}^{t}$. Alternatively, one should solve the problem in a backward recursion as follows. We imagine first that choosing the best choice of $\mathrm{CC}^{(H-1)}$ given $\{G^{(v)}_t\}_{0\leq v<H-1, v\leq t\leq v+L-1, H-2}$, that is $\mathrm{CC}^{(H-1)}$ is the minimizing value and $U^{(H-1)}$ the minimized value of

$$\mathbb{E}\left[\sum_{t=0}^{H-1}\left(\mathrm{NAV}_{(t)} - T\right)^2 \middle| \mathcal{F}_{(H-2)}\right]$$

where $\mathcal{F}_{(t)}$ is the filtration generated by $\{G^{(v)}_{t'}\}_{0\leq v\leq t'\leq t}$, that is all growth noise generated up to and including time t. The optimal solutions $\mathrm{CC}^{(H-1)}$ and $U^{(H-1)}$ are measurable with respect to $\mathcal{F}_{(H-2)}$, meaning a function of those random variables generating that σ-algebra. Now when we optimize $\mathrm{CC}^{(H-2)}$ we do it by noting we will subsequently make the optimal choice of $\mathrm{CC}^{(H-1)}$ after we can observe $\{G^{(v)}_{(H-2)}\}_{v\leq H-2}$, so we don't need to make it yet. Thus, we now let $\mathrm{CC}^{(H-2)}$ be the minimizing value and $U^{(H-2)}$ the minimized value of

$$\mathbb{E}\left[U^{(H-1)} \middle| \mathcal{F}_{(H-3)}\right]$$

Alternatively, this is the average of $U^{(H-1)}$ (which is $\mathcal{F}_{(H-2)}$ measurable) with respect to the variables $\{G^{(v)}_{(H-2)}\}_{v\leq H-2}$. Then we iterate back to get $U^{(0)}$ and $\mathrm{CC}^{(0)}$. That is, even if one takes U as the function to optimize, the controls are not exchangable variables. We need to minimize U over $\{\mathrm{CC}^{(v)} \in \mathcal{F}_{(v-1)}, \mathrm{CC}^{(v)} \geq 0\}_{0 \leq v \leq H-1}$, which is what motivates the backward stochastic programming solution. This essentially expresses the expectation in the definition of U by the law of total expectation as the iterated conditional expectation from $\mathcal{F}_{(H-1)}$ to $\mathcal{F}_{(-1)}$, which is then computed recursively.

5.2 Obstruction to Analytical Simplification When Executing a Backward Program

Given the linearity of the dynamics and quadratic structure of the cost function, there is motivation to pursue analytical reductions for the above backward stochastic programming procedure. However, this is frustrated by the non-negativity constraint on the capital commitments, as we show by conducting two steps of the iteration.

At the beginning of the last year $t = H - 1$ in the horizon, we wish to choose $\mathrm{CC}^{(H-1)}$ to minimize $\mathbb{E}\left[(\mathrm{NAV}_{(H-1)} - T)^2 \middle| \mathcal{F}_{H-2}\right]$. From (3.2), this is equivalent to minimizing

$$\begin{aligned}
&\mathbb{E}\left[\left(\sum_{v=H-L}^{H-2} (1 - \mathrm{RD}_{(H-1-v)})\, \mathrm{NAV}^{(v)}_{(H-2)} (1 + G^{(v)}_{H-1}) + \sum_{v=H-L}^{H-1} \mathrm{CC}^{(v)}\, \gamma_{(H-1-v)} - T\right)^2 \middle| \mathcal{F}_{H-2}\right] \\
&\quad = (\gamma_{(0)}\, \mathrm{CC}^{(H-1)})^2 + \\
&\qquad + 2\gamma_{(0)}\, \mathrm{CC}^{(H-1)} \left[\sum_{v=H-1-L}^{H-2} (1 - \mathrm{RD}_{(H-1-v)})\, \mathrm{NAV}^{(v)}_{(H-2)} (1 + \bar{G}) + \mathrm{CC}^{(v)}\, \gamma_{(H-1-v)} - T\right] \\
&\qquad + \dots
\end{aligned} \tag{5.2}$$

with respect to $\mathrm{CC}^{(H-1)}$, where we have omitted terms independent of this quantity. This amounts to minimizing a quadratic, assuming $\gamma_{(0)} > 0$, giving the optimal choice

$$\mathrm{CC}_*{}^{(H-1)} = \max\left(-\frac{1}{\gamma_{(0)}}\left[\sum_{v=H-L}^{H-2} (1 - \mathrm{RD}_{(H-1-v)})\, \mathrm{NAV}^{(v)}_{(H-2)} (1 + \bar{G}) + \mathrm{CC}^{(v)}\, \gamma_{(H-1-v)} - T\right], 0\right).$$

We have here enforced the best choice is no new capital contribution when $\mathrm{NAV}_{(H-1)}$ is expected to exceed T based on the information available at the beginning of year $H - 1$. However, we cannot short the private equity!

Now, initializing an optimization strategy relies on the previous capital commitment $\mathrm{CC}^{(H-2)}$. This will involve an average with respect to the noise variables $G^{(v)}_{(H-2)}$ of a quadratic expression in $\mathrm{CC}_*{}^{(H-1)}$, among other terms which are easy to compute. Note that $\mathrm{CC}_*{}^{(H-1)}$ depends on $G^{(v)}_{(H-2)}$ via $\mathrm{NAV}^{(v)}_{(H-2)}$, and this dependence is nonlinear because of the rectification. This makes evaluating the average a complicated function of the noise variables from earlier time as well as the capital commitments $\mathrm{CC}^{(0)}, \dots, \mathrm{CC}^{(H-2)}$ and so, unfortunately, analytical progress in this direction is not feasible.

6 Numerical Optimization Strategies

In this section, we attempt three numerical strategies. The three strategies are: optimization of a naive linear heuristic control (Subsection 6.1), gradient descent optimization of a linear heuristic control taking into account NAV in previous years (Subsection 6.2), and dynamic, or backward, programming (Subsection 6.3).

6.1 Naive Heuristic Approach

Our first update strategy is as follows: at each time after L, we compute the difference $\mathrm{NAV}_{(t)} - T$ and then set

$$\mathrm{CC}_{(t+1)} = \big(1 + \alpha(\mathrm{NAV}_{(t)} - T) + \beta\big)_+$$

where α and β are parameters that we determine experimentally, and $(y)_+ \equiv \max(y, 0)$ keeps the capital commitment from going negative. In particular, we minimize α and β with respect to the loss function given by

$$L\big(\text{NAV}_{(20)}\big) = \mathbb{E}\Big[\big(\text{NAV}_{(20)} - T\big)^2\Big] \tag{6.1}$$

where $\text{NAV}_{(20)}$ is the NAV at year 20 and T is the target value.

In Figure 4, we for some reason plot the ℓ^2 error of 1000 realizations, which is just the loss function (6.1) multiplied by 1000. We find that the parameters that minimize the loss function are approximately $\alpha \approx 0.06$ and $\beta \approx 0.03$, as seen in Figure 4. When simulating 1000 realizations of the stochastic process and letting $\alpha = \beta = 0$, which corresponds to no attempts to control $\text{NAV}_{(t+1)}$, we have a ℓ^2 error of approximately 40, so this very naive control provides roughly a 5% improvement.

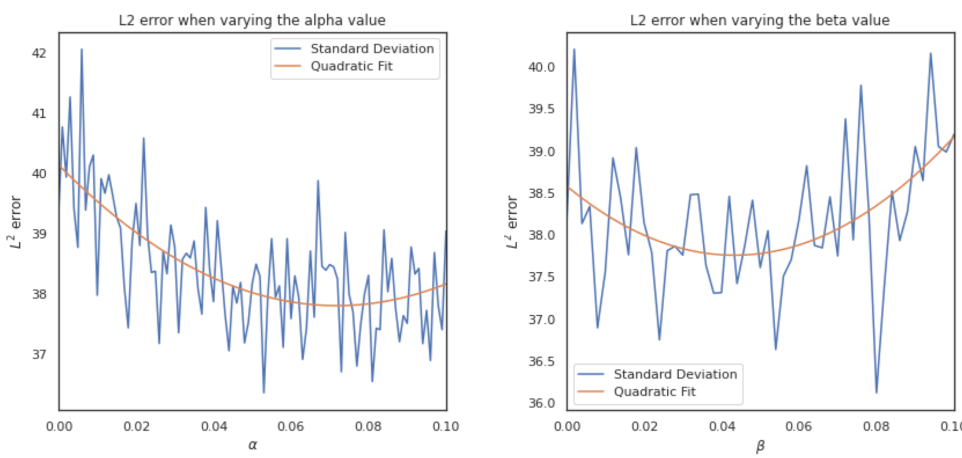

Figure 4. Analyzing the ℓ^2 error ((6.1) multiplied by 1000, the number of realizations) when varying (a) α and (b) β. The ℓ^2 errors are then fit with a quadratic polynomial.

6.2 Heuristic Gradient Descent Approach

As a generalization of the naive strategy, we attempt

$$\text{CC}_{(t+1)} = 1 + \sum_{j=0}^{L-1} S_j \, \text{NAV}_{(t-j)} + \beta \tag{6.2}$$

with parameters $\{\{S_j\}_{j=0}^{L-1}, \beta\}$. We attempt to find an optimal choice of these control parameters via gradient descent of the same loss function L as in Section 6.1. For a general function F, gradient descent performs the iterative updates

$$x_{i+1} = x_i - \gamma \nabla F(x_i) \tag{6.3}$$

where x_i is a sequence of guesses that are intended to approach x^*, the true minimizer and where γ is the step size parameter, chosen non-adaptively. Also, we regularize noisy behavior by using limiters on the gradient in order to prevent wild updates which may occur because of the stochasticity. The gradient descent algorithm, by virtue of its local nature, is only guaranteed to converge to a unique minima if the function F is convex. Results shown in Figure 4 for the simpler model from Subsection 6.1 show the noisiness clearly spoils convexity, though maybe if the limiters provide enough smoothing, one might hope for some kind of convexity.

6.3 Dynamic (Backward) Programming Strategy

In this section, we discuss a more general approach to stochastic optimization based on dynamic programming (see [2, 4] for helpful introductions). We begin by using a more standard notation so that the computational method remains clear. To this end, consider the stochastic control problem:

$$J\left(t, x_t, v_t\right)=\min_{u \in \mathcal{U}} \mathbb{E}\left\{\left.\sum_{s=t}^{T} f_0(s, X_s, u_s(X_s, V_s)) \right\rvert\, X_t=x_t, V_t=v_t\right\} \tag{6.4}$$

subject to the discrete-time state equation

$$X_{t+1}=f_1(X_t, u_t, V_t), \quad X(t=0)=X_0 \in \mathbb{R}^n \tag{6.5}$$

The admissible class of controls $\mathcal{U}$ is a given compact subset of the positive reals and $V_t \in \mathbb{R}^m,\ t=0,1,...$ are independently distributed random variables such that $\mathbb{E}\left\{V_t\right\}<\infty$. Note that this implies the dynamics are Markovian.

The key idea in a dynamic programming method is to break a decision problem into smaller subproblems. Quoting Richard Bellman, the father of dynamic programming, as he describes his celebrated eponymous principle of optimality: "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision." [1, 3]

Bellman's principle of optimality, applied to the objective functional in (6.4) leads to the following functional equation

$$J\left(t-1, x_{t-1}, v_{t-1}\right)=\min_{u_{t-1} \in \mathcal{U}}\left\{f_0\left(t-1, x_{t-1}, u_{t-1}\right)+\mathbb{E}\left\{J(t, X_t, V_t) | X_{t-1}=x_{t-1}, V_{t-1}=v_{t-1}\right\}\right\}. \tag{6.6}$$

Equation (6.6) is often called Bellman's equation. Using (6.5), Bellman's equation can equivalently be expressed as

$$J\left(t-1, x_{t-1}, v_{t-1}\right)=\min_{u_{t-1} \in \mathcal{U}}\left\{f_0\left(t-1, x_{t-1}, u_{t-1}\right)+\mathbb{E}\left\{J(t, f_1\left(x_{t-1}, u_{t-1}, v_{t-1}\right), V_t)\right\}\right\}. \tag{6.7}$$

Now (6.7) is solved pointwise in some compact set $\Omega \subset \mathbb{R}^{n+m}$. This process is then performed backwards and iteratively from $t-1$, until the initial time $t=0$ is reached. This computational procedure, which is surprisingly quite simple to implement, is briefly summarized in Algorithm 1.

Algorithm 1: Backward Programming

Result: A control u_t which solves Problem (6.4)
Input: A state/noise-space domain Ω, an objective function f_0, a dynamical update function f_1, and a probability model for V_t.
for each moment in time $t-1$ **do**
  for each point in the discretized space Ω **do**
    a minimization of $f_0\left(t-1, x_{t-1}, u_{t-1}\right)+\mathbb{E}\left\{J(t, f_1\left(x_{t-1}, u_{t-1}, v_t\right), V_t)\right\}$ over u_{t-1}
  end
end

The greatest challenge in attempting to execute this stochastic programming numerically for the extended Yale model is the $L+N_{\mathrm{C}}-1$-dimensional state space (currently active vintage NAVs plus the impact of the CCs that are still paying in). With our basic model parameters $L=10$ and $N_{\mathrm{C}}=3$ and limited computational resources, this is not feasible within the span of a week long workshop. We attempt to circumvent this "curse of dimensionality" by using the reduced dynamics discussed in Section 4.1, which gives an approximate Markovian dynamics in N_{C}-dimensional space.

Specifically, with $N_{\mathrm{C}}=3$, we take the state variable as

$$\mathbf{X}_t=\left(\mathrm{NAV}_{(t)}, \mathrm{CC}^{(t)}, \mathrm{CC}^{(t-1)}\right), \tag{6.8}$$

with dynamics:

$$\begin{aligned}\mathbf{X}_{t+1} &= \left(\mathrm{NAV}_{(t+1)}, \mathrm{CC}^{(t+1)}, \mathrm{CC}^{(t)}\right)\\ &= \left(\mathrm{NAV}_{(t)}(1+S_{t+1}+\iota_{t+1})(1-\delta_{(t+1)}) + \tfrac{1}{3}\textstyle\sum_{v=t-1}^{t+1}\mathrm{CC}^{(v)}, \mathrm{CC}^{(t+1)}, \mathrm{CC}^{(t)}\right)\\ &\equiv f_1(\mathbf{X}_t, \mathrm{CC}_{(t+1)}, \mathbf{v}_t)\end{aligned}$$

where now the noise variable is two-dimensional:

$$\mathbf{V}_t = \left(S_{t+1}, \iota_{t+1}\right)$$

with independent systematic, respectively idiosyncratic, components S_t and ι_t. Note the indexing shift from the usual control $u_t = \mathrm{CC}_{(t+1)}$ (the control applied to update from state at end of year t to end of year $t+1$ is associated to year $t+1$), and our noise variable has a similarly unimportant shift in index.

Now, referring to previous notation in the context of the objective functional (5.1), we have

$$f_0(\mathbf{x}) = (x_{T,1} - T)^2,$$

depending only on the first (NAV) component of the state variable and not explicitly on the control or the noise. This means we can drop the v_t dependence in (6.4). Bellman's equation then becomes

$$\begin{aligned}J(t-1, \mathbf{x}_{t-1}) = \min_{CC^{(t)}\geq 0} &\left\{(x_{t-1,1} - T)^2\right.\\ &\left. +\mathbb{E}J\left(t, x_{t-1,1}(1+v_{t-1,1}+v_{t-1,2})(1-\delta_t) + \frac{1}{3}\left(x_{t-1,2}+x_{t-1,3}+CC^{(t)}\right), \mathbf{V}_t\right)\right\}.\end{aligned} \tag{6.9}$$

The backward stochastic programming iteration starts at $t = H-1$ by noting

$$J(H-1, \mathbf{x}_{H-1}) = (x_{H-1,1} - T)^2,$$

with no relevant control to optimize over at this step. As shown in Subsection 5.2, the optimization over $\mathrm{CC}^{(H-1)}$ at the previous step $t = H-2$ can be calculated analytically as a one-dimensional constrained quadratic optimization. The resulting $J(H-2, \mathbf{x}_{H-2})$ is no longer quadratic, and subsequent averaging and optimization steps must be implemented numerically.

During the workshop, this general backward programming procedure was simplified to actually only look one time step into the future (rather than all the way out to the end step at $t = H-1$) so that the optimization functional is at each time t of the simple form (5.2), with the simplification of assuming steady-state distribution across vintages. Thus in particular, we do not need to store the numerically computed $J(t, x_t)$ as we would if we were indeed optimizing over the entire future horizon. This simplified backward program with our reduced three-dimensional discrete-time dynamical system can be run on an average laptop within the span of an hour.

In essence, one step of the backward program reduces to solving an optimization problem based on the steady-state approximation to (5.2) for each point in $\tilde{\Omega} \subset \mathbb{R}^{N_C}$ representing the state space of $\mathbf{x}_{t-1}$. Though this could have been done analytically, it was implemented numerically with the aim of allowing for generalization to the full future horizon planning. This step is expressed in Algorithm 1 and in the MATLAB code provided in Appendix A.3 where MATLAB's `fmincon` is used in the optimization. As the optimization problem is solved for each point in Ω and for each time, the optimal commitment plan is stored in a four-dimensional array; three for the states and one for the control. In realizing the dynamics of NAV_t, we interpolate the array storing the optimal plan in order to realize its value, $CC^{(t)}$ for a given $NAV_{(t-1)}$, $CC^{(t-2)}$, and $CC^{(t-1)}$. This interpolation is handled via MATLAB's `gridded Interpolant`. The backward program costs just a few minutes on an average Macbook Pro, while realizing the dynamics for a given plan costs less than one second.

A result of the backward programming is shown in Figure 5 with a target value of $T = 3$ and a time horizon of $H = 20$ years is used. We see that the backward program is successful in maintaining a constant exposure near the target net asset value. In Figure 6, we show several realizations of the dynamics with

$T = 3$ and $T = 8$. Again, we see that, on average, the capital commitment plans computed are successful in their objectives over the later times. Discrepancies at earlier times are likely attributable to the failure of the assumption of steady state distribution across vintages which was used to reduce the state space in Subsection 4.

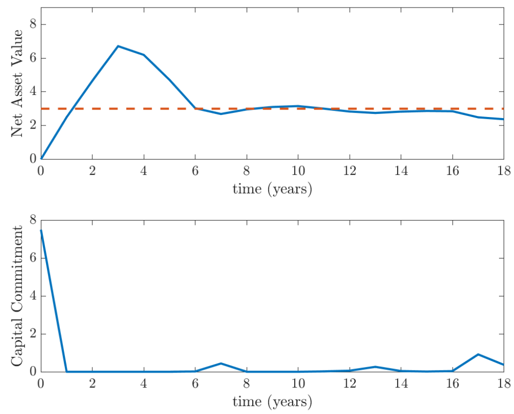

Figure 5. A result of Algorithm 1 applied to the optimal control problem given by (5.1). Here, a target of $T = 3$ is used, as indicated by the dotted line.

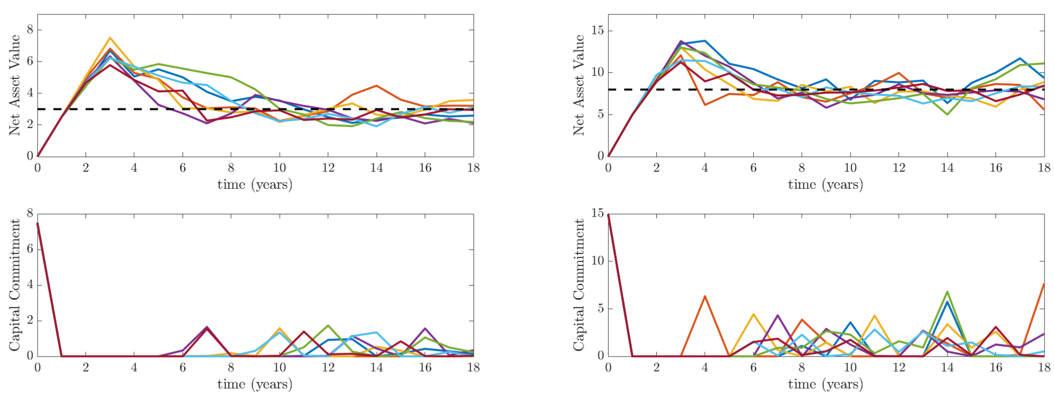

Figure 6. Several realizations with commitment plans computed via dynamic programming and with conventions consistent with Figure 5. The left has a target of $T = 3$ while the right has a target of $T = 8$.

7 Concluding Remarks and Future Directions

In this work, we addressed the problem of how to manage capital commitments to various vintages in order to achieve a desired target level of exposure. We adopted and extended the Yale modeling framework by accounting for dynamical noise across various vintages. We pursued a few numerical approaches with various levels of success.

For the heuristic control strategy, we examined linear relationships between the next capital commitment and recent net asset values, constrained to forbid negative (short) capital commitments. Optimal coefficient choices in these rectified linear strategies were sought which minimized the mean-square target exposure mismatch, but the numerical evaluations of this loss function via Monte Carlo simulations were quite noisy and thus frustrated the search for the minimizing parameters. None of the parameter choices explored seemed to substantially reduce the variability in actual exposure. But the numerical studies reinforced a basic mathematical point that emerged in the discussions – that in calibrating a stochastic asset return model to a noise-free deterministic asset return model, the deterministic growth factor appears more appropriately to be chosen in terms of the *geometric* mean not the *arithmetic* mean of the multiplicative noisy growth factor. Preliminary explorations of bang-bang controls, in which one chooses at each opportunity a fixed positive value of capital commitment or no capital commitment based on a threshold for the current net asset value, seemed to work better.

We also pursued a simplified one-step dynamical programming approach which shows promise despite some drawbacks. While managing reasonable control of exposure fluctuations in the second 10 years of the time horizon, the early years show a substantial overshoot of the target exposure. Better results might be achieved by modifying the loss function during the ramp-up period, modifying the steady-state vintage distribution assumption, and/or implementing the full horizon optimal control.

Finally, we outline some ideas that were generated at the workshop but were not yet explored. One idea is to formally write the continuum limit of the extended Yale model, with a short time between updates. Although this does not model the given financial application, this may give ideas for control based on experience with continuous-time differential equations. The other extended thoughts are presented in the remaining subsections: a modified one-step optimization procedure (Subsection 7.1), efforts toward a meaningful and interpretable representation of control strategies that could be optimized by dynamical programming (Subsection 7.2), and an extension to public as well as private equity commitments (Subsection 7.3).

7.1 Greedy One-Step Forward Optimization

As an alternative to the complexities of the technically precise backward programming strategy, which in practice appeared to necessitate collapsing the dynamics of NAV by vintage by assuming a steady state distribution, we propose a greedy forward iteration approach which proceeds at each time t by choosing an optimal choice of the capital commitment $\mathrm{CC}^{(t)}$ under the simplifying assumption that the same capital commitment will be made at all future years: $\mathrm{CC}^{(t)} = \mathrm{CC}^{(t+1)} = \ldots = \mathrm{CC}^{(H-1)}$. Then, from (3.5), we have a linear relationship between the future NAV values of extant and new vintages and the current NAV values of extant vintages and the single future capital commitment variable. Then, in the optimization step at time t,

$$\min_{\mathrm{CC}^{(t)} \geq 0} \mathbb{E}\left[\sum_{s=t}^{H-1} (\mathrm{NAV}_{(s)} - T)^2\right],$$

the expectations can be explicitly computed to give a constrained quadratic optimization in the single variable $\mathrm{CC}^{(t)}$ (with the current NAV values entering as parameters). This solution can be expressed analytically using a similar calculation to Subsection 5.2. After $\mathrm{CC}^{(t)}$ is chosen in this way, in the next year, the optimization problem is repeated (with no reference to how previous capital commitments were chosen) in terms of the shorter horizon and updated NAV values.

While this does not provide an optimal solution, it potentially improves over the quasi-steady state strategy, which tends to cause on overshoot in the early years. This forward iteration idea can be generalized to allow for some variability in the capital commitments over the next few years (and then specifying later commitments as a linear function of these somehow), but the optimization strategy would be harder to implement due to the non-negativity constraints in multiple dimensions. So a forward iterative one-step greedy improvement at every step (pretending the future capital commitments will always be the same) should strike a reasonable balance between effectiveness and simplicity. In particular, it does not require a steady state distribution assumption on the NAV values of vintages, and allows them to be represented in full detail.

7.2 Obtaining Interpretable Control Strategies from Backward Programming

An optimal control strategy derived from stochastic dynamical programming does in general suffer from a lack of transparent interpretability. The rules of the computed optimal control strategy could be plotted at later times to suggest a simpler structural rule whose parameters could be fit to the results of the optimal control strategy. The performance of variations of parameters of this simpler rule could then be examined via simulations, just as for the *a priori* heuristic rules considered in Subsection 6.1. The optimal control strategy in particular can be examined at different times to suggest different rules during the ramp-up period and the maintenance period. Alternatively, if the capital commitments are restricted to a finite set of choices, the stochastic dynamical program could have a shorter computation time and produce more interpretable rules in terms of time-varying thresholds of net asset value to determine the capital commitment choice.

To elaborate on this latter idea, suppose we restrict the capital commitments to a finite set of options, say $\mathrm{CC}^{(v)} \in \{0, \mathrm{cc}_1, \mathrm{cc}_2, \ldots, \mathrm{cc}_{N_{\mathrm{CC}}}\}$, with the positive state values indexed in increasing order. This naturally reduces the computational complexity through the restriction of the set of possible controls, and in particular makes an exhaustive search appear feasible for small values such as $N_{\mathrm{CC}} = 3$. We can attempt to further speed up the search for an optimal control $\mathrm{CC}^{(t+1)}$ at each backward iteration step by quite reasonably assuming $\mathrm{CC}^{(t+1)}$ should be a decreasing function of $\mathrm{NAV}_{(t)}$, for any specified still active capital commitments $\mathrm{CC}^{(t)}, \mathrm{CC}^{(t-1)}$. That is, the optimal choice should have the following form

$$CC^{(t+1)} = \sum_{j=1}^{N_{\mathrm{CC}}} \mathrm{cc}_j \mathbf{1}_{[\theta_{j+1}(t,\mathrm{CC}^{(t)},\mathrm{CC}^{(t-1)}),\theta_j(t,\mathrm{CC}^{(t)},\mathrm{CC}^{(t-1)})}(\mathrm{NAV}_{(t)})$$

where $\mathbf{1}_S(\cdot)$ is the indicator function of the set S, and $\theta_1 \geq \theta_2 \geq \ldots \theta_{N_{\mathrm{CC}}+1} \equiv 0$ denote the thresholds at which the optimal capital commitment choice changes. In other words, the optimal strategy amounts to solving for N_{CC} decision thresholds at each time; at long time these thresholds presumably stabilize at fixed values. This makes the long-term investment strategy easily interpretable, with the time-dependent transient phase allowing for the rampup period. These thresholds also depend on active capital commitments, but we might expect this to be a monotonic dependence. If the active capital commitments are higher, this should reduce the NAV thresholds for reducing future capital commitments. In other words, $\theta_j(t, \mathrm{CC}^{(t)}, \mathrm{CC}^{(t+1)})$ should be a decreasing function of the second two arguments.

This all suggests various routes for speeding up the solution for the optimal control. For a given $\mathrm{CC}^{(t)}$ and $\mathrm{CC}^{(t-1)}$, one could solve for the decision thresholds $\{\theta_j\}_{j=1}^{N_{\mathrm{CC}}}$ by a bisection process. Initial values of NAV to sample could be guided by solving them for the case of deterministic returns ($G_t^{(v)} = \hat{G}$) or from previous solutions. The expected monotonic dependence on the existing capital commitments could expedite the optimal solution over their other possible values.

7.3 Extension to Include Public Equity Investments

Additionally, there is yet another extension to the Yale model that may be considered. In practice, we do not have only one type of asset; instead we have many different types of assets. Consider a two asset model

$$\mathbf{T} = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

where T_1 is a target value for public assets and T_2 is a target value for private assets. Work involving this extension is also subject for future study.

8 Acknowledgments

We thank the Worcester Polytechnic Institute (WPI) for their kind hospitality and for hosting the Mathematical Problems in Industry workshop during the week of June 13, 2022. We especially thank Dr. Kai Sikorski, representing Goldman Sachs, for introducing the team to this interesting applied mathematics problem and for continued guidance throughout the workshop. We also gratefully acknowledge support from the Society for Industrial and Applied Mathematics, the National Science Foundation, and WPI's Center for Industrial Mathematics and Statistics. Arkady Wey additionally acknowledges support from the EPSRC Centre for Doctoral Training in Industrially Focused Mathematical Modelling.

References

[1] R. Bellman, *Dynamic Programming*, Dover, 2003.
[2] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, Athena Scientific, 1996.
[3] S. Dreyfus, *Richard Bellman on the birth of dynamic programming*, Operations Research, 50 (2002), pp. 48–51.
[4] A. Seierstad, *Stochastic Control in Discrete and Continuous Time*, Springer, 2009.
[5] D. Takahashi and S. Alexander, *Illiquid alternative asset fund modeling*, The Journal of Portfolio Management, 28 (2002), pp. 90–100.

Appendix A Codes

A.1 Code Used to Analyze Yale Model (Python)

```python
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy as sp

%matplotlib inline
%config InlineBackend.figure_format = 'svg'

diri='./'
sns.set(rc={"lines.linewidth": 2.5,"figure.figsize":(6,6)},font_scale=1.25)
sns.set_context("notebook")
sns.set_style("white")

target = 8.166

def timeprop(L=10, B=4, Y=0, CC=1, RC=[1/3, 1/2, 1], vintages=1000,
growth=0.13, Sspread=0.1*np.sqrt(2), Ispread=0.1*np.sqrt(2), endyear=20,
strategy=None, strategyinputs=None):
    # default values
    # L = 10 # life of fund
    # B = 4 # bow
    # Y = 0 # zero yield
    # CC = 1 # 1 total capital contributions
    # 1/3rd CC contribution for 3 years for RC
    # 1000 vintages
    # 13% growth rate, evenly split between S and I
    # end in year 20
    # target = 8.166 # from deterministic case

    # growth rate is evenly split among systematic and idiosyncratic parts
    Gpart = growth / 2
    pic = np.zeros((endyear, vintages)) # paid in capital
    c = np.zeros((endyear, vintages)) # amount called in
    d = np.zeros((endyear, vintages)) # amount distributed
    nav = np.zeros((endyear, vintages)) # net asset value
    RD = np.zeros((endyear, vintages)) # rate of distribution
    poc = np.zeros((endyear, vintages)) # paid out capital
    Gvals = np.zeros((endyear, vintages)) # growth rate for vintage and year
    CC = CC * np.ones(endyear + 1) # base capital contribution, reshape correctly
```

```
allvintages = list(range(vintages)) # all the vintages
activevintages = [] # vintages that are currently active

# reshape input RC correctly
RC = np.transpose(np.tile(np.array(np.pad(RC, (0, endyear-len(RC)),
mode='constant', constant_values=(0, 0))), (vintages, 1)))

rng = np.random.default_rng() # make RNG generator

# generate systematic growth rates
Svals = rng.normal(loc=Gpart, scale=Sspread, size=endyear)
# generate idiosyncratic growth rates
Ivals = rng.normal(loc=Gpart, scale=Ispread, size=(endyear, vintages))
# compute growth rate for each vintage in each year
Gvals = np.transpose(np.tile(Svals, (vintages, 1))) + Ivals

for j in range(endyear): # calendar year
    if allvintages:
    # check that there are vintages remaining to be added
        activevintages.append(allvintages.pop(0)) # add a new vintage
    for i in activevintages:
        # print(j-i)
        pic[j,i] = np.sum(c[:j,i])
        c[j,i] = RC[j-i,i] * (CC[i] - pic[j,i])
        RD[j,i] = max(Y, ((j - i + 1) / L) ** B)
        d[j,i] = RD[j,i] * nav[j-1,i] * (1 + Gvals[j,i])
        nav[j,i] = nav[j-1,i] * (1 + Gvals[j,i]) + c[j,i] - d[j,i]
    if j > L:
    # vintages starting dying after L years, start strategy after L years
        if activevintages:
            activevintages.pop(0) # oldest vintage dies
        if strategy:
            prevtot = np.sum(nav, axis=1)[j]
            CC[j+1] = strategy(j, target, np.sum(nav, axis=1), CC, strategyinputs)

return np.sum(nav, axis=1)
```

A.2 Numerical Optimzation Performed in Sections 6.1 and 6.2 (Python)

```
def lossfunc(navs, targetval=target):
return np.sum(np.square(navs - targetval)) / navs.size

def manyrealizations(realizations=1000, endyear=20, L=10, **kwargs):
reals = np.zeros((realizations, endyear))
for i in range(realizations):
    reals[i,:] = timeprop(endyear=endyear, L=L, **kwargs)

return reals[-1]

# single vintage, no stochasticity
```

```
plt.plot(timeprop(vintages=1,endyear=10, Sspread=0, Ispread=0), label='NAV')
plt.xlim(0, 9)
plt.ylim(0, 1.4)
plt.legend()
plt.xlabel('Year')
plt.ylabel('Value [$]')
plt.title('NAV for a single vintage with no stochasticity')
plt.show()

# many vintages, no stochasticity
plt.plot(timeprop(vintages=30,endyear=20,Sspread=0,Ispread=0), label='NAV')
plt.xlim(0, 19)
plt.ylim(0, 8.5)
plt.legend()
plt.xlabel('Year')
plt.ylabel('Value [$]')
plt.title('NAV for many vintages across 20 years with no stochasticity')
plt.show()

# many vintages, stochasticity
plt.plot(timeprop(vintages=30,endyear=20), label='NAV')
plt.xlim(0, 19)
plt.ylim(0, 15)
plt.axhline(target, label='Target Value', color='k')
plt.legend()
plt.xlabel('Year')
plt.ylabel('Value [$]')
plt.title('NAV for many vintages across 20 years with stochasticity')
plt.show()

# 10000 realizations without control, 27s
things = manyrealizations(realizations=10000)
baseloss = lossfunc(things)
baseloss

plt.hist(things[:,-1], histtype='stepfilled', bins=20, density=True,
label='Realizations', alpha=0.5)
plt.axvline(np.mean(things[:,-1]), label='Mean NAV', color='orange')
plt.axvline(target, label='Target value', color='k')
plt.legend()
plt.xlabel('NAV [$] at 20 years')
plt.ylabel('Frequency')
plt.title('Histogram of NAV value frequencies at 20 years')
plt.show()

def linearcontrol(j, target, totals, CCs, inputs):
    alpha = inputs[0]
    beta = inputs[1]
    return 1 + alpha * (target - totals[j]) + beta
```

```
    linconttest = manyrealizations(strategy=linearcontrol, strategyinputs=[0.06, 0.03])

plt.hist(linconttest[:,-1], histtype='stepfilled',
bins=20, density=True, label='Realizations', alpha=0.5)
plt.axvline(np.mean(linconttest[:,-1]), label='Mean NAV', color='orange')
plt.axvline(target, label='Target value', color='k')
plt.legend()
plt.xlabel('NAV [$] at 20 years')
plt.ylabel('Frequency')
plt.title('Histogram of NAV value frequencies at 20 years with linear control')
plt.show()

# vary alphas and beta in the linear control

alphas = np.mgrid[-1:1:51j]
betas = np.mgrid[-1:1:41j]

losses = np.zeros((len(alphas), len(betas)))

for i in range(len(alphas)):
    for j in range(len(betas)):
        losses[i, j] = lossfunc(manyrealizations(realizations=1000,
        strategy=linearcontrol, strategyinputs=[alphas[i], betas[j]]))

        plt.contourf(betas, alphas[20:], losses[20:,])
plt.colorbar()
plt.xlabel(r'$\beta$')
plt.ylabel(r'$\alpha$')
plt.title(r'Loss when varying $\alpha$ and $\beta$')
plt.tight_layout()
plt.show()

# next vary them independently

alphas = np.mgrid[-0.2:1:51j]

losses = np.zeros(len(alphas))

for i in range(len(alphas)):
    losses[i] = lossfunc(manyrealizations(realizations=1000, strategy=linearcontrol,
    strategyinputs=[alphas[i], -0.05]))

plt.plot(alphas, losses)
plt.xlim(np.min(alphas), np.max(alphas))
plt.show()

# next vary them independently

betas = np.mgrid[-0.5:0.5:51j]
```

```
losses = np.zeros(len(betas))

for i in range(len(betas)):
    losses[i] = lossfunc(manyrealizations(realizations=1000,
    strategy=linearcontrol, strategyinputs=[0.15, betas[i]]))

plt.plot(betas, losses)
plt.xlim(np.min(betas), np.max(betas))
plt.show()
```

A.3 Dynamic Programming (MATLAB)

```
%% Dynamic Programming on Reduced Model
opt=optimoptions('fmincon','Display','off','Algorithm','sqp');

% steady state phi thanks to Anthony's work
phi=([0.         ; 0.03529382; 0.09082137; 0.13613295; 0.15853617;
       0.16118738; 0.15215328; 0.1381865; 0.08686989; 0.04081864]);

% NAV domain
a=0;b=6;N=20;
NAV=linspace(a,b,N);

% time domain
M=19;
t=linspace(0,M,M+1);

% control domain
lb=0;ub=8;U=20;
controlstate1=linspace(lb,ub,U);
controlstate2=linspace(lb,ub,U);

% space of controls
controlspace=ones(M,N,U,U);

% model parameters for systematic and idiosyncratic risk
meanSt=0.065;
varSt=0.02;
meanIota=0.065;

% precompute the factors for effective distribution rate
% and idiosyncratic risk
deltat=0;
iotaVarSumFactor=0;

% the bow
B=4;

L=length(phi);
```

```
for v=1:L

    % rate of distribution
    RD=(v/(L))^B;

    % effective distribution rate
    deltat=deltat+RD*phi(v);

    % an important factor in computing idiosyncratic risk
    iotaVarSumFactor=iotaVarSumFactor+(phi(v))^2*(1-RD)^2;

end

% this is the variance for the idiosyncratic risk iota
varIota=varSt/((1-deltat)^2)*iotaVarSumFactor;

mu=1.13;
xi=1+(meanSt^2+varSt)+...
    (meanIota^2+varIota)+2*meanSt*meanIota+2*meanSt+2*meanIota;

% Target
T=3;

% for each moment in time
for i=1:M

    % for each point in NAV space
     for j=1:N

         % for each point in CC_t-2 space
         for k=1:U

             % for each point in CC_t-1  space
             for l=1:U

                % perform a backward program
                % on a (problem-dependent) Objective
                index=M+1-i;
                X=NAV(j);
                Y=controlstate1(k);
                Z=controlstate2(l);
                Objective=...
                @(u)X^2-2*T*X+2*T^2+xi*X^2*...
                (1-deltat)^2+1/9*(Y+Z+u)^2-2*mu*(1-deltat)*X*T...
                +2/3*mu*(1-deltat)*X*(Y+Z+u)-T/3*(Y+Z+u);

                % use a black box constrained optimization algorithm
                controlspace(index,j,k,l)=...
                    fmincon(Objective,.5,[],[],[],[],lb,ub,[],opt);
```

```
            end

        end

    end

end
```

A.4 Forward Iteration using Control Law found Via Dynamic Programming (MATLAB)

```
% initialization
Xt=0;
NAVs=zeros(1,M+1);
NAVs(1)=Xt;
controlvector=zeros(1,M);
controlminus1=0;
controlminus2=0;

% interpolation grid
[Xq,Yq,Zq] = ndgrid(NAV,controlstate1,controlstate2);

for j=1:M

    % pick out the controls for the current time step
    controlslice=squeeze(controlspace(j,:,:,:));

    % use linear interpolation
    interpcontrol=griddedInterpolant(Xq,Yq,Zq,controlslice);

    % evaluate the piecewise polynomial at the current state
    optimalcontrol=interpcontrol(Xt,controlminus1,controlminus2);

    % realize the systematic risk St
    St=normrnd(meanSt,sqrt(varSt));

    % realize the idiosyncratic risk Iotat
    Iotat=normrnd(meanIota,sqrt(varIota));

    % update the net asset value
    Xt=NAVs(j)*(1+St+Iotat)*(1-deltat)+...
        (controlminus1+controlminus2+optimalcontrol)/3;
    NAVs(j+1)=Xt;
    controlvector(j)=optimalcontrol;
    controlminus2=controlminus1;
    controlminus1=optimalcontrol;
end
```

```
% Visualize the state and the controls over some time window
h=subplot(2,1,1);
plot(0:M-1,NAVs(1:M)), axis tight
xlabel(time (years)'),ylabel('Net Asset Value'), ylim([0 9])
hold on
plot(0:M-1,T*ones(1,M),'--'), hold off

subplot(2,1,2)
plot(0:M-1,controlvector(1:M)), axis tight
xlabel('time (years)'),ylabel('Capital Commitment'), ylim([0 8])
```