A valuation framework for compound real options

Steinar Ekern (steinar.ekern@nhh.no), NHH
Norwegian School of Economics, Helleveien 30, 5045 Bergen, Norway

Mark B. Shackleton (corresponding author, m.shackleton@lancaster.ac.uk)
Lancaster University Management School, Bailrigg, Lancaster, LA1 4YX, UK.
+44 1524 594131 (847321 fax)

Sigbjørn Sødal (sigbjorn.sodal@uia.no), University of Agder
School of Business and Law, PO 422, Kristiansand 4604, Norway.

May 2021

A valuation framework for compound real options

Abstract: If a firm knows the decision cost of an operational change, an investment policy is possible. Such flexibility, e.g. to switch cashflows, is valued by numerically customizing real option solutions at policy rules consistent with decision costs. Comparative statics have also proceeded numerically but we analyse these *analytically*. By presenting all boundary conditions as a *linear system*, for different stochastic processes, we *calculate decision costs* given assumed policies. To infer the one policy that matches actual decision costs, we demonstrate an iterative search algorithm using the *analytical* power of our comparative statics in investment and switching decisions (99 words).

Keywords: Decision costs, optimal policies, comparative statics, smooth pasting and discount functions. EFM 430, C61, G31.

- Highlights: Describes optimality conditions for switching in arithmetic and geometric economies that are consistent with caps and floors.

- Lays out a framework for switching at *assumed* policy rules and solves for decision costs that are consistent with these initial *inputs*.

- Evaluates the sensitivity and comparative statics of decision costs to input rule levels and suggests an initial choice for policy rules suitable for starting an *iterative search*.

- With the difference between calculated and *target decision costs*, we use the analytical sensitivity in an iteration algorithm to find the *target policy* and comparative statics that are consistent with *known decision costs*.

1 Introduction

Since the breakthrough of Black and Scholes [5] and Merton [27], option pricing has made great strides in financial markets through the use of stochastic processes and mathematics. In particular the risk neutral valuation method of Cox, Ross and Rubinstein [11] etc. has allowed option claims to be valued on a no arbitrage basis consistent with market prices.

This thinking permeated into corporate finance with the area of so called "real options" (Myers [28]) mirroring the developments of traded options with e.g. applications to operational switching given for example in Brennan and Schwartz [6] and McDonald and Siegel [25]. This change in corporate decision analysis was supported by texts following Dixit and Pindyck [15] (e.g. Trigeorgis [35] and Brennan and Trigeorgis [7]) further developing and refining the analysis underlying valuation in this area.

This stimulated work on areas such as industrial capacity (Pindyck [30]) and valuation of land (Capozza and Li [10]) but real options have also been applied to marginal cost of capital (Abel, Dixit, Eberly and Pindyck [1]), capital structure (Sarkar and Zapatero [32]) and mergers and acquisitions (Lambrecht [23]). The analysis of competition, game theory and monopoly has been subjected to real options thinking (e.g. Smit and Ankum [33] and Pereira and Rodrigues [29]) as has thinking at all stages from growth (Kraft, Schwartz and Weiss [22]) to exit (Alvarez [4]).

Although less has been written on flexible investment sequences, real compound options and modularity have been studied (Ekern [19] modelled hysteresis with a finite number of repeated switches and Gamba and Fusari [20] motivate and value project design using six modularity principles). Guthrie [21] discussed the tradeoffs between scale and flexibility and similarly Dahlgren and Leung [12] applied this to infrastructure.

Overall the real options approach has become widespread and this introduction can only touch on the many articles in this area (see Lambrecht [24] for a recent comprehensive survey and critique).

However, the conclusions of many of these papers rest upon comparisons of investment dynamics and comparative statics that are derived for specific examples and numerical solutions to complex systems. This is because it is rarely the case that the investment situation can be solved directly. Furthermore the results may depend on the choice of the stochastic process used to drive the state variable (most papers use geometric Brownian motion but some e.g. Alexander, Mo and Stent [3], use an arithmetic process).

We wish to progress *analytically*, particularly for *comparative statics*. To do so, we proceed using the discount factor approach of Dixit, Pindyck and Sødal [16] and Sødal [34]. This relates the value an option at any arbitrary state (or time) to its value at a boundary condition, using a function of the separation of that state from the boundary (which proxies for the stopping time taken).

We extend this approach to embrace the first order condition known as smooth pasting and we consider the second order condition of double smooth pasting. The easiest situations in which to develop the method are that of reversible caps/floors which are limiting cases of investment hysteresis Dixit [14] or costly reversibility (Abel and Eberly [2] and Eberly and van Mieghem [18]). However, we show how to extend this method to many levels.

Our contribution is an investment framework that solves for compound option values and decision costs directly from *assumed or input* policy rules. By taking care with the necessary conditions at policy points (boundaries) this method is applied to multiple switching points.

Our method allows the comparative statics to be evaluated analytically for any input policy, not numerically as is the case in the literature. We use the analytical sensitivity of decision costs to policy assumptions in an iterative algorithm that seeks the single policy that matches *known decision costs*. This allows a wider range of real option situations to be explored with greater analytical tractability.

We start with one policy rule with two way switching, under arithmetic and geometric processes (Section 2). Then we extend to two policy rules and decision costs and solve for the latter as a function of the former, including comparative statics (Section 3). Then we tackle the search iteratively by suggesting a policy start point as well as an updating algorithm in Section 4 that converges to the particularly policy that matches target costs. In section 5 we detail two way switching with a long floor, short cap, i.e. collar example. We conclude in Section 6.

2 One policy rule R & decision cost X

2.1 Method

The first policy decision we consider is simple. If investment or divestment of a known present value cost of magnitude $X^\dagger$ gains or releases an asset with stochastic value, at what value threshold $R^\dagger$ of this stochastic asset is it optimal for *continuous and fully reversible switching to occur*? The dagger symbol† is used to label both the *known* decision cost $X^\dagger$ and *target i.e. unknown* policy threshold $R^\dagger$.

Typically the analysis of investment and divestment situations presents the actual decision cost $X^\dagger$ and expects to determine the value maximising policy threshold $R^\dagger$. In some situations this is possible directly but generally it is not, so we frame the process the other way around.

If investment and divestment decisions were taken at a *general* rule R at which value maximisation was ensured by certain conditions, what present value decision costs $X(R)$ would be consistent with that choice of R for the policy rule? To do this we employ the *necessary* conditions for the policy R to optimal given X (during the numerical example in section 4 we also test for *sufficient* conditions for R to maximise value given X).

Knowledge of the explicit function $X(R)$ using these conditions allows comparative statics such as $\frac{\partial X(R)}{\partial R}$ to be evaluated explicitly.

Although R represents one of many optimal policies and $X(R)$ its cost, they do not carry the dagger symbol† because they are different to the special *target policy* pair $X^\dagger, R^\dagger$. However if we can calculate values $X(R)$ that are consistent with any input R, this can be used *iteratively* until $X(R^\dagger) = X^\dagger$ is found.

For any input policy rule R, we ensure it follows *necessary conditions* to solve for the function $X(R)$ (other conditions can be tested whilst searching). This also allows determination of the sensitivity of X to each choice R.

An initial guess R will not generally be consistent with the actual decision costs $X^\dagger$ but the *calculated error* $X(R) - X^\dagger$ is informative of the unknown difference $R - R^\dagger$.

Using this sign and magnitude of this error we *iterate* from that choice of R toward the *target* value $R^\dagger$ using the sensitivity information in the function $\frac{\partial X(R)}{\partial R}$. This allows us to converge on the solution to the function $X(R^\dagger) = X^\dagger$ even though it is not invertible i.e. $R(X)$ cannot be found explicitly.

2.2 Setup

We suppose there is a *stochastic value* Π_t (derived from a stochastic cashflow π_t) that evolves over time t. Initially, we include the time subscript t as a reminder that Π_t is dynamic.

In this section there is just one policy rule for value switching, R (without subscript[1]), and the state variable can encounter this one policy value at many possible times t i.e. $\Pi_t = R$.

This section is concerned with the proximity of the stochastic process Π_t to the single *policy value* R and its crossing. Each time the policy rule is crossed, a fixed lump sum decision cost X is either invested, or retrieved (equivalent to valuing caps and floors on flows). This enables value to be derived from continuously switching between the stochastic value Π_t (and its dividend) and the *non-stochastic lump sum* X (and its risk free funding) on every occasion that $\Pi_t = R$. This flexibility can be represented using two switching options above and below R, both of which are exercised at R.

This approach is consistent with a cashflow approach. At $\Pi_t = R$ the cashflow on the stochastic asset is a percentage dividend yield on value R and the cashflow on X is the continuous risk free rate r on X, i.e. rX. If the yield rate on the asset and the risk free rate differ, then the cashflows can differ upon switching even if R, X are equal.

If the target policy is to switch when the cashflow on Π_t at R matches

[1]In later sections once the method is established and it necessary to track multiple policy thresholds, subscripts are used to indicate their **level** e.g. $R_1 < R_2$.. etc. To avoid confusion, the state variable is then simply Π. It is important to note that the subscripts on R are not used to track individual times or sequences.

that of a target fixed cashflow rate, $rX^\dagger$ (the risk free rate r on $X^\dagger$), that can also be accommodated.

When matching values or cashflows, the option values for switching above and below the decision rule R may differ but total value maximisation including options can be achieved by optimal switching at any general level.

2.3 Policy input

Therefore, for any given value choice e.g. an initial guess of rule $R = \$1$ for the policy, the objective of this paper is to calculate the non-stochastic decision cost X that is consistent with that R being optimal, i.e. to find decision cost $X(R)$ as a function of policy choice R.

With an instantaneous cashflow rate of $\pi_t dt$, the value Π_t is consistent with a perpetual integral of the expected future (risk neutral) of these flows discounted, with the asset growth and yield rates constrained by the risk free rate. Also in terms of flows, X can be represented as a perpetual integral of fixed risk free flows (flows within the integral for X are $rXdt$ for every infinitesimal time dt).

The value of such flexibility anticipates each crossing that Π_t makes over the fixed value threshold R, i.e. it captures the expected present value benefit from the cashflow changes before they actually occur.

These options can be thought of as purchases and sales conditional upon the stochastic process value reaching the policy point R.

In this section, we define the value that anticipates flexibility $V(\Pi_t)$ as that *in excess of* the immediate maximum of the two local payoffs $\max(\Pi_t, X)$ and the total value $V(\Pi_t) + \max(\Pi_t, X)$, i.e. flex plus payoff, is given in equation (1).

$$\begin{aligned} V(\Pi_t) + \max(\Pi_t, X) &= C(\Pi_t) + X \quad \text{for } \Pi_t \text{ below or at } R \\ &= P(\Pi_t) + \Pi_t \quad \text{for } \Pi_t \text{ above or at } R \end{aligned} \tag{1}$$

When $\Pi_t \geq R$, the flexibility value is derived from the ability to *Put* Π_t and receive X so we label it $P(\Pi_t)$. When $\Pi_t \leq R$, its value is derived from the ability to *Call* Π_t by giving X so we label this $C(\Pi_t)$. These put and call options have no final expiry date and are potentially perpetual. Whilst they have zero value at a limiting boundary (zero or infinity), their value depends upon their early exercise *switching payoff* at the policy boundary R. In this section, they have a common decision cost X which is viewed as the *exercise price* on both put and call option.

In this perfectly reversible situation, the put and the call only exist at the same time at the instant $\Pi_t = R$ (in the next section with R_1, R_2 they can both exist with a region) but at the single threshold R they are *exchanged* for each other.

If a policy R is to be optimal and consistent with decision costs X, then total value either side of the policy threshold must match and the two lines of equation (1) will be equal at $\Pi_t = R$. The necessary first order condition is *smooth pasting* (Dixit and Pindyck [15]), which we address. Because in this section it is perfectly reversible, a second order condition will hold too also.

Only in special cases will the value of the put and call be equal at the policy rule $\Pi_t = R$, i.e. generally $C(R) \neq P(R)$. This is because the options account for immediate cashflow and other differences. The impact can be seen in the value matching equation (2) which makes the right hand sides of equation (1) explicit by tracking value below R (where the call is un–exercised) matching values above R, where the put is un–exercised. At this rule point, both options are exercised as a passing exchange of their values $C(R), P(R)$.

$$C(\Pi_t) + X = P(\Pi_t) + \Pi_t \quad \text{at } \Pi_t = R \text{ i.e.} \quad C(R) + X = P(R) + R \qquad (2)$$

Now we can see that even with the perfect reversibility in this section, if the options have different values at the policy rule, the fixed sum decision cost X will not equal the policy rule R i.e. $C(R) - P(R) = R - X \neq 0$.

2.4 Conditions for optimal exercise at $\Pi_t = R$

For equation (2) to represent optimal exercise of both the call on the way up *and* the put on the way down, first and second order conditions must hold.

Firstly, the equation must smooth paste, that is have the same sensitivity to Π_t on either side. Secondly, if reversible, i.e. can go both ways at the *same* threshold, it must *doubly smooth paste* (Dumas [17]) not just in the first derivative with respect to Π_t but in the second derivative too.

The first order condition for maximal total value on both sides of equation (2) i.e. smooth pasting, ensures an equivalent response to Π_t before and after instantaneous action at R. That is to say that when the put flexibility $P(\Pi_t)$ is used to change Π_t to X (when crossing R from above), the net sensitivity with respect to Π_t should balance, including that of the call flexibility $C(\Pi_t)$ generated (and vice versa on the way up).

How to implement smooth and double smooth pasting depends on the type of economy that is modelled. Firstly we do this for arithmetic flows and values.

2.5 Arithmetic flows

The simplest choice for stochastic value Π_t would have it depend in a linear manner on a *driftless* arithmetic Brownian motion for a *cashflow rate* π_t. The rate π_t has units of dollars *per year* and for the driftless situation it gives a stochastic perpetual capital value $\Pi_t = \pi_t/r$ measured in dollars (r, the continuous risk free rate, is measured in reciprocal years).

For risk neutral Brownian increments dW_t, equation (3) shows; the stochastic driver (a function of dollar risk rate Σ and Brownian increments dW_t), its driftless risk neutral expectation, its squared change and the no–arbitrage condition for a time homogeneous option claim $V(\Pi_t)$ (where we have used $V''(\Pi_t)$ for the second derivative with respect to Π_t).[2]

$$\frac{d\pi_t}{r} = d\Pi_t = \Sigma dW_t \quad E^{RN}[d\Pi_t] = 0 \quad (d\Pi_t)^2 = \Sigma^2 dt \quad rV(\Pi_t) = \frac{1}{2}\Sigma^2 V''(\Pi_t) \tag{3}$$

[2] The condition at the end of equation (3) is a Bellman equation from the self funding criteria $rV(\Pi_t)dt = E^{RN}[dV(\Pi_t)]$ and Ito's Lemma $dV(\Pi_t) = \frac{\partial V(\Pi_t)}{\partial \Pi_t} d\Pi_t + \frac{1}{2}\frac{\partial^2 V(\Pi_t)}{\partial \Pi_t^2}(d\Pi_t)^2$.

These are solved by general solutions $V(\Pi_t) \propto e^{\pm b\Pi_t}$. Thus the options $C(\Pi_t)$ and $P(\Pi_t)$ have value proportional to $e^{b\Pi_t}$ and $e^{-b\Pi_t}$ where the coefficients $\pm b$ satisfy $\frac{1}{2}\Sigma^2 b^2 = r$.

Note that the units of b are *reciprocal dollars* and the units of risk Σ^2 are *dollars squared per year* (unlike σ which we use in the next geometric section, expressed as % per year). We assign a value of $\Sigma^2 = (\$1)^2$ p.a. which leaves $\Sigma = \$1$ comparable over a year to the level at which we also set $R = \$1$. This means that the two option sensitivities $\pm b = \pm 0.20\$^{-1}$ are consistent with a continuous risk free rate of $r = 2\%$.

2.6 Discounting options for arithmetic processes

We chose to represent option values in a manner consistent with the *discount factor* approach of Dixit, Pindyck and Sødal [16]. This means that any prior option value is considered to be a discounted fraction of its payoff at a boundary rule. The payoffs are the non-stochastic values $C(R), P(R)$ and the discounted values of a dollar at R, presented as a function of the stochastic value Π_t, are $e^{-b(R-\Pi_t)}, e^{-b(\Pi_t-R)}$ respectively for call and put.

This means that before the boundary encounter (far away from their common boundary R, options tend to zero) the options have stochastic values given by:-

$$C(\Pi_t) = C(R)e^{-b(R-\Pi_t)} \quad \Pi_t \leq R \quad \text{and} \quad P(\Pi_t) = P(R)e^{-b(\Pi_t-R)} \quad \Pi_t \geq R$$

2.7 Arithmetic single and double smooth pasting

We label single and double differentiation with respect to the stochastic state variable with a dash$'$ and double dash$''$ e.g. $\frac{\partial C(\Pi_t)}{\partial \Pi_t} = C'(\Pi_t)$ and $\frac{\partial^2 C(\Pi_t)}{\partial \Pi_t^2} = C''(\Pi_t)$. Single and double differentiation of the state variable with respect to itself give 1,0 i.e. $\Pi_t' = 1$ and $\Pi_t'' = 0$.

Next in equation (4) we present all necessary conditions in three columns; value matching, smooth pasting and double smooth pasting, from left to right. The two rows do this for dynamic Π_t in the first line and *evaluated at*

$\Pi_t = R$ in the second lines of equation (4).

$$\begin{array}{lll} C(\Pi_t) + X = P(\Pi_t) + \Pi_t & C'(\Pi_t) = P'(\Pi_t) + 1 & C''(\Pi_t) = P''(\Pi_t) \\ C(R) + X = P(R) + R & bC(R) = -bP(R) + 1 & b^2 C(R) = b^2 P(R) \end{array} \tag{4}$$

The first order condition (centre column) brings a multiplier b in front of the option values and renders the reference asset to unity and the second differentiation (right column) brings another factor b in front of the options but eliminates the reference asset (its double differential is zero). Note that the call has positive slope $b > 0$ and the put negative slope $-b < 0$ on Π_t, but they both have positive convexity b^2 on Π_t. Also the first and second order conditions have units other than dollars (the units in value matching) for these arithmetic flows (in the geometric section we ensure that all conditions are expressed in dollar units).

Solving backwards in this symmetric and driftless situation, the call and the put have equal value (from the second order condition, last column of equation (4)) because their second derivative and convexities are equal. From the condition in the middle column, this value is a half of $1/b$ (a dollar value), i.e. \$2.5 in this case, and finally $X = R = \$1$, again because of symmetry.

For $R = \$1$ the solutions to the call and put $(C(R), P(R)$ which are symmetric and equal in value) are both \$2.5 and the option "strike" prices or decision costs consistent with this reversible setup are also $X = \$1$. Using a drifted motion would break the symmetry of put and call, this comes up in the next section where we also put each of the three conditions into the same dollar units within the geometric economy.

2.8 Geometric economy

In the arithmetic economy, the reference state variable could become zero or negative on bad news. It is therefore not suitable for representing an asset since it could become a liability. Following Samuelson [31] and McKean [26], option pricing has worked with geometric processes for positive assets which are incapable of becoming liabilities. Shares and in particular stock indices are bounded at zero and investment analysis uses regressions of *percentage changes* (not dollar changes) to test for return sensitivity.

Consequently the majority of real options papers use a geometric Brownian motion to drive values from capitalised flows i.e. $\Pi_t = \frac{\pi_t}{\delta}$, with a dividend yield parameter δ and risk neutral drift rate $r - \delta$. Equation (5) shows; percentage changes in the state variable(s), the expected risk neutral drift and total volatility rates in the first row.

$$\frac{d\pi_t}{\pi_t} = \frac{d\Pi_t}{\Pi_t} = (r-\delta)dt + \sigma dW_t \quad \frac{E^{RN}[d\Pi_t]}{dt} = (r-\delta)\Pi_t \quad \frac{(d\Pi_t)^2}{dt} = \sigma^2\Pi_t^2$$
$$rV(\Pi_t) = (r-\delta)\Pi_t V'(\Pi_t) + \frac{1}{2}\sigma^2\Pi_t^2 V''(\Pi_t) \tag{5}$$

The second line of equation (5) is the time homogeneous Bellman condition derived from the self financing condition (in footnote (2)). Now the option solutions are given by equation (6) along with the condition on their *beta* parameters β.

$$C(\Pi_t) = C(R)\left(\frac{\Pi_t}{R}\right)^{\beta_C} \qquad \text{for } \Pi_t \le R$$
$$P(\Pi_t) = P(R)\left(\frac{\Pi_t}{R}\right)^{\beta_P} \qquad \text{for } \Pi_t \ge R$$
$$\frac{1}{2}\sigma^2\beta(\beta-1) = r - (r-\delta)\beta \qquad \text{for both betas } \beta = \beta_C, \beta_P \tag{6}$$

The widely used option constants are explicitly labelled betas, $\beta_C > 1$ for the call and $\beta_P < 0$ for the put.

These are betas[3] because they represent the % response on each option (not dollar as before) to a % change in the state variable (previously b was linked to the dollar response and had units of $\$^{-1}$) and these betas β_C, β_P

[3] We assume the geometric reference process Π_t is traded and can be used for replication of the call, put or other option claim. E.g. if Π_t is the unit beta market portfolio ($\beta_\Pi = 1$), then β_V is the market beta of $V(\Pi_t)$. Otherwise β_V is the beta *relative* to the asset Π_t and other betas would be scaled by $\beta_\Pi \neq 1$. Without losing generality we assume Π_t to have a beta of one, i.e. Π_t is the tradeable reference pricing asset and β_C, β_P are relative to it.

are dimensionless.[4]

To represent option values in a manner consistent with the *discount factor* approach of Dixit, Pindyck and Sødal [16] where any prior option value is considered to be a discounted fraction of its payoff at a boundary, the payoffs are again $C(R), P(R)$ but the discounts presented as a function of the stochastic value Π_t are now given in equation (7).

$$\begin{aligned} C(\Pi_t) = D_C(\Pi_t, R)C(R) \quad \text{where } D_C(\Pi_t, R) &= \left(\frac{\Pi_t}{R}\right)^{\beta_C} \quad \text{for } \Pi_t \leq R \\ P(\Pi_t) = D_P(\Pi_t, R)P(R) \quad \text{where } D_P(\Pi_t, R) &= \left(\frac{\Pi_t}{R}\right)^{\beta_P} \quad \text{for } \Pi_t \geq R \end{aligned} \tag{7}$$

We label these functions $D_C(\Pi_t, R)$ for the call and $D_P(\Pi_t, R)$ for the put discount; note that the function's argument contains the rule at which the discount attains unit value (i.e. $D_C(R, R) = D_P(R, R) = 1$). Unlike the exponential discounts in the arithmetic section, when these are differentiated with respect to Π_t, their dependence on Π_t changes, e.g. in the first line of equation (8).

$$\begin{aligned} C'(\Pi_t) = D'_C(\Pi_t, R)C(R) = \beta_C \frac{\Pi_t^{\beta_C - 1}}{R^{\beta_C}} C(R) \quad \text{or} \\ C'(\Pi_t)\Pi_t = D'_C(\Pi_t, R)C(R)\Pi_t = \beta_C \frac{\Pi_t^{\beta_C}}{R^{\beta_C}} C(R) = \beta_C C(\Pi_t) \end{aligned} \tag{8}$$

However, if we differentiate *and* multiply by Π_t as in the second line of equation (8), then the resulting quantity can be interpreted as the beta of the call times its value *at any level*. If smooth pasting is implemented this way, the constants β_C, β_P multiply *option values in dollars* and the smooth pasted condition has the *same units* as the value matching condition.[5]

[4] The betas satisfy $\beta_C, \beta_P = \frac{1}{2} - \frac{r-\delta}{\sigma^2} \pm \sqrt{\left(\frac{r-\delta}{\sigma^2} - \frac{1}{2}\right)^2 + \frac{2r}{\sigma^2}}$. Setting $r = \delta = \sigma^2$ simplifies the two solution betas to $\beta_C = 2, \beta_P = -1$ and equates the convexities of both to $\gamma = \beta(\beta - 1) = 2$. However setting $\sigma^2 = 2r = \frac{4\delta}{3}$ gives $\beta_C = 2, \beta_P = -0.5$ and $\gamma_C = 2, \gamma_P = 0.75$. We prefer to fix β_c, β_P rather than choose base parameters r, δ, σ since betas directly affect single and double smooth pasting. Note that one degree of freedom remains.

[5] We aim for coefficients in value matching, smooth and double smooth pasting that

2.9 Smooth and double smooth pasting

Therefore in the geometric economy, we effect smooth pasting by differentiating items on both sides of equation (2) with respect to Π_t and then *multiplying across all elements* by Π_t to re–scale. That is to say instead of taking $\partial[.]/\partial\Pi_t$ across components as for the arithmetic case, we take $\partial[.]/\partial\Pi_t \times \Pi_t$ labelled $[.]'\Pi_t$ (where $[.]'$ indicates differentiation of bracket contents with respect to Π_t).

It is important to note that this rescaling is only evaluated at the instant that smooth pasting occurs $\Pi_t = R$. Each switching option has a beta at levels other than a policy rule but it only the measurement of betas at a rule determine optimality. Although smooth pasting is implemented as $[.]'R$, conceptually it is $[.]'\Pi_t$ at the moment when $\Pi_t = R$.

When this is applied to non-stochastic X, a zero results $[X]'\Pi_t = 0$ and when it is applied to Π_t the result is $[\Pi_t]'\Pi_t = \Pi_t$ itself because these two have sensitivities of 0,1 respectively. We interpret these sensitivities as *betas consistent with market percentage returns.* This works for any asset $V(\Pi_t)$ because β_V is $\partial V(\Pi_t)/V(\Pi_t)$ divided by $\partial\Pi_t/\Pi_t$, or it is the relative sensitivity of percent changes in $V(\Pi_t)$ divided by percent changes in Π_t.

Applying scaled differentiation to both representations of value means taking $[V(\Pi_t)+\max(\Pi_t, X)]'\Pi_t$. Applying this to equation (2) (which was not specific to arithmetic Brownian motion) generates the first order condition, equation (9), which contains products of options and their betas at stochastic level Π_t and rule R consistent with geometric Brownian motion.

$$\Pi_t + \beta_P P(\Pi_t) = \beta_C C(\Pi_t) \quad \text{at } \Pi_t = R \text{ i.e.} \quad R + \beta_P P(R) = \beta_C C(R) \tag{9}$$

In the geometric economy, the advantage of equation (9) compared to

are consistent with a second order expansion of percentages in $V(\Pi_t)$ against percentages in the reference asset Π_t (N.B. $\gamma_V = \beta_V(\beta_V - 1)$ for GBM).

$$\frac{dV(\Pi_t)}{V(\Pi_t)} \approx \frac{V'(\Pi_t)\Pi_t}{V(\Pi_t)}\frac{d\Pi_t}{\Pi_t} + \frac{1}{2}\frac{V''(\Pi_t)\Pi_t^2}{V(\Pi_t)}\left(\frac{d\Pi_t}{\Pi_t}\right)^2 \equiv \beta_V\frac{d\Pi_t}{\Pi_t} + \frac{\gamma_V}{2}\left(\frac{d\Pi_t}{\Pi_t}\right)^2$$
$$V'(\Pi_t)\Pi_t \equiv \beta_V V(\Pi_t) \qquad V''(\Pi_t)\Pi_t^2 \equiv \gamma_V V(\Pi_t)$$

differentiation without scaling, is that it gives $\beta_C C(\Pi_t)$ etc. which are *beta weighted values* that use the same terms (and units) as the value matching equation. When smooth pasting the lines of equation (2) at $\Pi_t = R$ this way, the betas β_C, β_P in equation (9) multiply values to equate the sum products of betas and values.

Where switching can occur frequently and *reversibly*, "double smooth pasting" must hold where the *second derivatives* across both sides of value matching align too Dumas ([17]). This is achieved using the second derivative, a gamma γ_V, with respect to Π_t i.e. $[.]''$. Again we chose to re-scale, this time by Π_t^2 i.e. taking $[.]''\Pi_t^2$, which produces a gamma multiplied by a value i.e. $\gamma_V V(\Pi_t)$ (where $V(\Pi_t)$ is either the put or the call). At the instant of switching, this is applied as $[.]''R^2$.

When double differentiation and scaling is applied to the value matching equation, not only does this knock out the X but it also knocks out the linear claim Π_t too i.e. $[\Pi_t]''\Pi_t^2 = 0$. This means that the double differentiated version of value matching contains the options alone. The double smooth pasting elements, weighted by gamma convexity, are given in equation (10).

$$\gamma_P P(\Pi_t) = \gamma_C C(\Pi_t) \quad \text{at } \Pi_t = R \text{ i.e.} \quad \gamma_P P(R) = \gamma_C C(R) \tag{10}$$

Now under GBM, equations (2), (9) and (10) present *linear* combinations of option values with the betas playing the role of weights on the same values $C(R), P(R)$ at the policy rule. Their solutions including the analytic comparative static for $X(R)$ are:-

$$C(R) = \frac{\beta_P - 1}{\beta_C(\beta_P - \beta_C)} R \qquad P(R) = \frac{\beta_C - 1}{\beta_P(\beta_P - \beta_C)} R$$
$$X(R) = \frac{\gamma_C - \gamma_P}{\beta_C \beta_P(\beta_P - \beta_C)} R + R \qquad \frac{\partial X(R)}{\partial R} = 1 + \frac{\gamma_C - \gamma_P}{\beta_C \beta_P(\beta_P - \beta_C)}$$

and in the symmetric case when convexities $\beta_P = \beta_C$ are equal, $X = R$ and $\frac{\partial X(R)}{\partial R} = 1$.

Using investment betas and gammas at the boundary, we found the value $X(R)$ and sensitivity to the choice of R $\frac{\partial X(R)}{\partial R}$ along with option constants

$C(R), P(R)$. Here these are all in a linear relationship, where the choice of R determines the scale. Later we will show other situations with multiple costs and thresholds, e.g. $X_1(R_1, R_2), X_1(R_1, R_2)$ with their cross dependencies.

2.10 $X(R)$ for different inputs

To complete the illustration here using the linear system for three unknowns $X(R), P(R), C(R)$ with three equations, we show two sets of inputs for symmetric and non–symmetric cases. Firstly we show $\beta_C = 2, \beta_P = -1$ and $\gamma_C = \gamma_P = 2$ so that:-

$$C(R) + X = P(R) + R \qquad 2C(R) = -P(R) + R \qquad 2C(R) = 2P(R) \tag{11}$$

i.e. the put and the call are symmetric and equal, both equal to a third of the rule level R and due to the symmetry, the decision cost $X(R) = R$.

However for $\beta_C = 2, \beta_P = -\frac{1}{2}$ and $\gamma_C = 2, \gamma_P = \frac{3}{4}$ (where the cash dividend yield δ and risk free r rate differ), the values satisfy equation (12).

$$C(R) + X = P(R) + R \qquad 2C(R) = -\frac{1}{2}P(R) + R \qquad 2C(R) = \frac{3}{4}P(R). \tag{12}$$

The *relative value* of the put and the call are established from the last column of each set, which say that the dollar weighted convexity on the call must match that on the put $\gamma_C C(R) = \gamma_P P(R)$. The *absolute value* of put and call are then determined in the middle column of these equations by the choice of R that fixes the *scale* of the setup.

This sheds light on why decision costs are subordinated to option values and thresholds in the solution process, they appear in the value matching conditions alone (left hand column) and drop out of the smooth pasting and double smooth pasting conditions.

For the second situation, Figure 1 shows the values of the claims (y axis) against the stochastic value Π_t on the x axis. $R = 1$ was an arbitrary choice input and $X = 1.5$ its output. On the left when $\Pi_t < R = 1.0$, the flexible

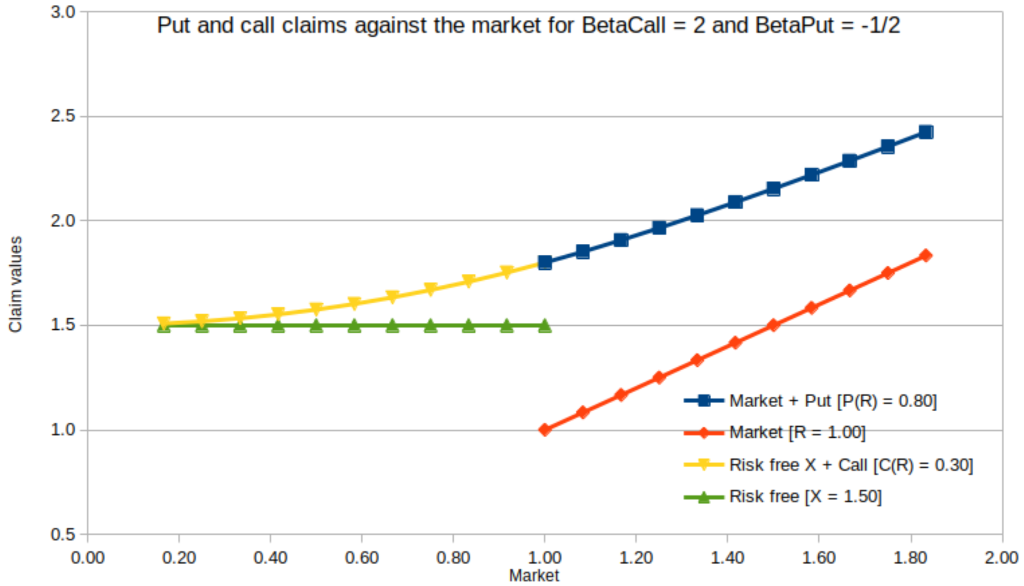

Figure 1: Put & call claims ($\beta_C = 2, \beta_P = -\frac{1}{2}$) for $R = 1$ implying $X = 1.5$.

situation holds $X = 1.5$ in cash (green, earning r) and benefits additionally from the call (yellow). On the right of Figure 1 when $\Pi > R = 1.0$ the flexible situation holds with the stochastic asset (red) and in addition the put (in blue). Due to the asymmetry and different gammas, when switching, the put (\$0.8) is worth more than the call (\$0.3), but the resulting $X = 1.5$ takes up the difference at $R = 1.0$.

The next sub–section (and Figure 2) show a choice that matches a target value of $X = 1.0$

2.11 Finding $R^\dagger$ given $X^\dagger$

For the symmetric option inputs the call and put where equal.[6] For the option inputs ($\beta_C = 2, \beta_P = -0.5$) and $R = \$1$ the system was solved by $P(1) = \$0.8, C(1) = \$0.3, X(1) = \$1.5$.

When we wish to find what policy is consistent with $X^\dagger = \$1$ instead, here it is trivial to find $R^\dagger$. The relative values of the options do not change but their absolute values must change to accommodate a different $R^\dagger$. In the first line we can see that if X is to be scaled down from \$1.5 to $X^\dagger = \$1.0$, R must be scaled from \$1.0 to $R^\dagger = \$\frac{2}{3}$ so that $C^\dagger(\frac{2}{3}) = \$\frac{1}{5}$ and $P^\dagger(\frac{2}{3}) = \$\frac{8}{15}$.

This situation is shown in Figure 2 which has the same betas and gammas as Figure 1, but has shifted the decision rule left to $R = \frac{2}{3}$. This lowers the imputed value of X to its desired value of \$1.

It is important to note that both decision rules $\$R = 1$ (Fig 1) and $R^\dagger = \$\frac{2}{3}$ (Figure 2) meet necessary smooth and double smooth pasting conditions, but only the second is consistent with the target decision costs $X^\dagger = \$1.0$ (the first is consistent with $X = \$1.5$). The smooth and double smooth properties of flexibility value can be seen in both figures where the yellow and blue lines meet.

From an assumed policy rule R, we have solved for option values at that decision rule or boundary ($C(R), P(R)$ represent the solution constants, e.g. as recommended in Dixit and Pindyck [15]). *Between* policy rules the discount functions give option values for stochastic value $\Pi_t \neq R$. In the examples given so far, the constants e.g. $C(R)$ scale with R, but this is not to say that the option values themselves increase with Π_t. As mentioned, away

[6] For the single threshold case when $\gamma_C = \gamma_P$, the dividend yield δ and the risk free rate r were equal. Then call $C(R)$ and put $P(R)$ at the rule were and the implied decision cost X the same as the decision rule R. Equation (1) is interpreted as an integral of discounted expected cashflows that are conditioned on the stochastic level Π_t using indicator functions $\mathbb{1}_{\Pi_t>R}$ and $\mathbb{1}_{\Pi_t<R}$ in an expression like $\int_t^\infty E_t^{RN}\left[\pi_s \mathbb{1}_{\Pi_s>R} + x\mathbb{1}_{\Pi_s<R}\right] e^{-r(s-t)}ds = \int_t^\infty E_t^{RN}\left[\max(\pi_s, x)\right] e^{-r(s-t)}ds$ where $x = rX$ is the running interest expense on decision cost X and $\pi_t = \delta\Pi_t$ is the cash dividend rate on stochastic asset Π_t. This is an integral of (out of the money) caplets or floorlets and the call and put options $C(\Pi_t)$ and $P(\Pi_t)$ can be thought of as (out of the money) Cap and a Floor i.e. $V(\Pi_t) + \max(\Pi_t, X) = Floor(\delta\Pi_t, rX) + \Pi_t$ or $Cap(\delta\Pi_t, rX) + X$ with $r = \delta$. If $r \neq \delta$ then $X \neq R$ and switching occurs at a different cashflow condition where a different balancing payment is required on switching ($\gamma_C \neq \gamma_P$).

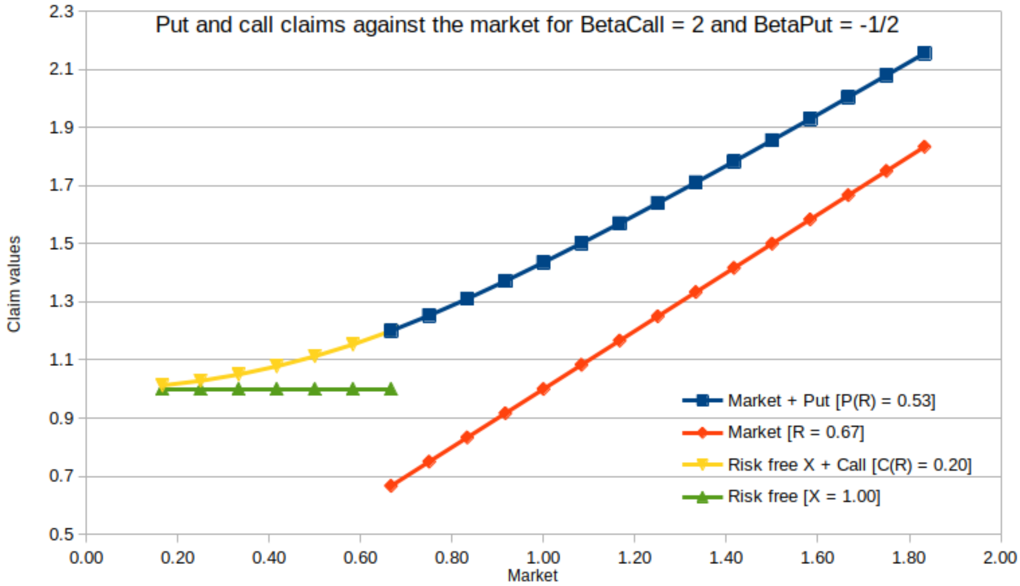

Figure 2: Put and call claims ($\beta_C = 2, \beta_P = -\frac{1}{2}$) for $R = \frac{2}{3}$ implying $X = 1$.

from rule R options are non-linear (discount) functions of the state variable e.g. $C(\Pi_t) = D_C(\Pi_t, R)C(R)$.

It is the *scale* of the set up and all options that increases linearly with R; passage within the system is tracked by Π_t.

With two thresholds R_1, R_2, the situation would scale if both move together but if only one moves, it would not. Since we require multiple thresholds, we will rely on matrix algebra to solve this linear decision system. Therefore we need to conduct multiple value matching and smooth pasting using conditions place in *vectors* e.g. $\mathbf{R} = [R_2; R_1]$.

3 Policy rules R_1, R_2 & decision costs X_1, X_2

Here we drop the time subscript on Π_t using Π instead and reserve subscripts for the decision rules R_1, R_2 etc. These rules increase in level with their subscript e.g. $R_1 < R_2$. Although the stochastic process is now labelled Π, it should not be forgotten that it is dynamic when comparing its value to the two threshold rules. As was the case in the last section, the focus is on proximity of Π to each rule and not the times at which contact with the boundary occurs $\Pi = R_1$ or $\Pi = R_2$.

In this section, the prior result is extended by *separating the up and down* switching rules, an approach that is common in hysteresis (Dixit [14]). So far they both occur at the same rule R but now we explicitly separate the transitions, *up at* R_2 and *down at* R_1. This allows the compound option approach for $V(\Pi)$ to include an overlap region between the two rules $R_1 < \Pi < R_2$, in this region we assume that the prior state from last boundary contact persists.[7]

Thus we examine *general rules* R_1, R_2 at which flexibility is exercised, i.e. $\Pi = R_1$ or R_2 aiming to solve for the decision costs $X_1(R_1, R_2), X_2(R_1, R_2)$ that would be committed at R_1, R_2 resp. It could be the case that exercising the cap or the floor incurs a sunk cost at each switch time; if this requires funding it *increases* the required benefit upon exercise, delaying it.

With $R_1 < R$ and $R < R_2$, we can think of the last section as a limiting case when both new rule levels converge to $R_1 = R = R_2$ so this method should be able to recover the same result as in the prior section. However with separation, the target decision costs $X_1^\dagger < X^\dagger < X_2^\dagger$ at the rules will be different to the X in the prior section, e.g. they might follow $X_2^\dagger = X^\dagger + K$ and $X_1^\dagger = X^\dagger - K$ where $K > 0$ represents a known frictional present value cost payable on both transitions.

This approach requires *two* value matching equations; in equation (13), we place flexibility value immediately *before* a transition on the *left* and immediately *after*, on the *right*. The value difference associated with cash flow changes go on the right, with positive values for sums gained and negative for those lost. Since we are fully switching between floating (Π) and fixed (X), we either gain R_2 or lose R_1 and $-X_2, +X_1$ are interpreted as the fixed decision costs incurred there. We also double the number of call and put option constants that must be tracked; $C(R_1), C(R_2), P(R_1), P(R_2)$.

[7] With separate rules, the representation is $\int_t^\infty E_t^{RN}\left[\delta\Pi_s \mathbb{1}_{\Pi_s > R_1} + rX\mathbb{1}_{\Pi_s < R_2}\right] e^{-r(s-t)} ds$. Where the state variable is a region that satisfies two indicator functions, it is the last one with which it had contact that takes precedent (so they remain mutually exclusive). With different switching levels, it becomes harder to represent this with cap and floor notation.

Since $\Pi_{(t)}$ cannot equal both R_2 and R_1 at the same time, these conditions occur at different times (t) but they have the same economic meaning (value matching) so grouping them is beneficial. Note that $\mathbf{\Delta}$ is a matrix (defined in footnote 8) that produces changes in cashflow patterns at the policy rules.

$$\begin{array}{ccccc} C(R_2) & = & R_2 - X_2 & + & P(R_2) \\ P(R_1) & = & X_1 - R_1 & + & C(R_1) \\ \mathbf{H} & = & \mathbf{\Delta}(\mathbf{R} - \mathbf{X}) & + & \mathbf{S} \end{array} \tag{13}$$

This stacking[8] also requires option values at the rules, these are placed in vectors labelled *Harvest* values in $\mathbf{H}$ (on the left) and *Seeds* in $\mathbf{S}$ on the right of equation (13). These contain call and put option values $C(\Pi)$, $P(\Pi)$ *evaluated at the relevant value* of Π, i.e. $\Pi = R_2$ and $\Pi = R_1$ giving $C(R_2)$, and $P(R_2)$ and $C(R_1)$, $P(R_1)$.

Vector $\mathbf{\Delta}(\mathbf{R} - \mathbf{X})$ carries both the values of the floating flows gained or lost and the fixed flows too i.e. a net gain on the switch going up of $R_2 - X_2$ and recovery of $X_1 - R_1$ on the down switch.

The top row of equation (13) shows the call option with value $C(R_2)$ (in $\mathbf{H}$) being harvested (on the left) to change the asset values $R_2 - X_2$ on the right plus seeding the put option $P(R_2)$ (which is less than $P(R_1)$ in the next row).

The bottom row of each vector is the application of value matching at $\Pi = R_1$, the put (in $\mathbf{H}$) is harvested with value $P(R_1)$ to release Π and lose value R_1, the present value of costs $-X_1$ are spared (in $\mathbf{X}$) and option $C(R_1)$

[8]Vectors below their labels are designed to display value matching when read across corresponding entries in each row.

$$\begin{array}{ccccc} \mathbf{H} & = & \mathbf{\Delta R} - \mathbf{\Delta X} & + & \mathbf{S} \\ \begin{bmatrix} C(R_2) \\ P(R_1) \end{bmatrix} & = & \begin{bmatrix} R_2 - X_2 \\ -R_1 + X_1 \end{bmatrix} & + & \begin{bmatrix} P(R_2) \\ C(R_1) \end{bmatrix} \end{array}$$

Since the change in value of assets occurs at known asset values, these can be expressed through matrix, vector $\mathbf{\Delta}, \mathbf{R}$ where $\mathbf{\Delta}$ is a matrix that captures the changes at the vector of rules $\mathbf{R}$.

$$\begin{array}{cccc} \mathbf{\Delta R} & = & \mathbf{\Delta} & \mathbf{R} \\ \begin{bmatrix} R_2 \\ -R_1 \end{bmatrix} & = & \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} R_2 \\ R_1 \end{bmatrix} \end{array}$$

(in $\mathbf{S}$) is re–seeded with lesser value than its harvest $C(R_2)$.

We have taken care to make each row carry items relating to rules at R_2, R_1 respectively and also to place options into one of two vectors that identify if they are harvested or seeded. Having $\mathbf{H}, \mathbf{S}$ like this is useful (more so than carrying the calls and puts in separate vectors, say $\mathbf{C}, \mathbf{P}$) because we now relate the value of option seeds to those at their harvesting using a 2×2 discount matrix.

In equation (14), the discount factors from the right of these equations (7) have been fixed and placed into a *discount matrix* $\mathbf{D}$ that allows option seed and harvest values in vectors $\mathbf{S}, \mathbf{H}$ from equation (13) to be related.

$$\begin{array}{ccccc} \mathbf{S} & = & \mathbf{D} & & \mathbf{H} \\ \begin{bmatrix} P(R_2) \\ C(R_1) \end{bmatrix} & = & \begin{bmatrix} 0 & D_P(R_2, R_1) \\ D_C(R_1, R_2) & 0 \end{bmatrix} & & \begin{bmatrix} C(R_2) \\ P(R_1) \end{bmatrix} \end{array} \tag{14}$$

This product between a matrix and a vector in equation (14) expresses the growth process from seed to harvest. Due to the rules of matrix multiplication, unlike equation (13) it is not read directly row wise ($C(R_2)$ does not multiply $D_P(R_2, R_1)$), but it ensures that seed values of each option are discounted versions of values when harvested.

Harvesting of each option always occurs after seeding (or at the same time in the limit as rules converge) but the discount factor takes into account the time value associated with waiting until the next action.[9]

Although we doubled the number of option contants by taking their value at two points, equation (14) compensates with two more conditions that must be satisfied.

[9] For time t and value Π_t (below the call or above the put rules R_2, R_1) if the harvests $t_H > t$ occur when Π_t hits the next put rule $R_1 = \Pi_{t_H}$ or next call rule $R_2 = \Pi_{t_H}$ the discount factors can also be derived using stopping times but discount factors *integrated out* random harvest (stopping) times t_H.

$$\begin{aligned} D_C(\Pi_t, R_2) &= E_t^{RN}\left[e^{-r(t_H - t)}\middle| \Pi_{t_H} = R_2\right] \\ D_P(\Pi_t, R_1) &= E_t^{RN}\left[e^{-r(t_H - t)}\middle| \Pi_{t_H} = R_1\right] \end{aligned}$$

We also need two versions of equation (9) in vector form that match value weighted betas via scaled smooth pasting. To implement scaled smooth pasting across all rules, in equation (15) we place the betas into two matrices $\beta_{\mathbf{H}}, \beta_{\mathbf{S}}$ that pre-multiply the harvest and seed option vectors $\mathbf{H}, \mathbf{S}$.

$$\begin{array}{ccccccccc} \beta_{\mathbf{H}} & \mathbf{H} & = & \mathbf{\Delta R} & + & \beta_{\mathbf{S}} & \mathbf{S} \\ \begin{bmatrix} \beta_C & 0 \\ 0 & \beta_P \end{bmatrix} & \begin{bmatrix} C(R_2) \\ P(R_1) \end{bmatrix} & = & \begin{bmatrix} R_2 \\ -R_1 \end{bmatrix} & + & \begin{bmatrix} \beta_P & 0 \\ 0 & \beta_C \end{bmatrix} & \begin{bmatrix} P(R_2) \\ C(R_1) \end{bmatrix} \end{array} \tag{15}$$

Equation (15) captures scaled smooth pasting at the two rules because the top and bottom lines in it are equivalent to (9) evaluated at $\Pi = R_2$ and $\Pi = R_1$ respectively. The change in value vector on the right, which contained floating and fixed items, is now just $\mathbf{\Delta R}$ because the fixed elements have zero beta.

There is no perfect reversibility here and no double smooth pasting, but in the limit as R_2 approaches R_1 this system will return the same results as Section 2. The two smooth pasting conditions become identical but replacing one condition with $\gamma_C C(R) = \gamma_P P(R)$ for double smoothness restores the necessary number of conditions.

3.1 Solving for $\mathbf{S}, \mathbf{H}$

Now we have three matrix equations, equation (15) smooth pastes at both thresholds, equation (14) discounts both seed to harvest values and equation (13 and its footnote) matches values at both rules.

For input decision rules R_1, R_2 etc. $\mathbf{S}, \mathbf{H}, \mathbf{X}$ form a *linear system* in $\mathbf{R}$ and the option constants used in the other elements. Equation (16) also shows the solution for $\mathbf{X}$, the cost or option strike vector at the end.

$$\mathbf{H} = [\beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D}]^{-1}\Delta\mathbf{R} \quad \mathbf{S} = \left[\beta_{\mathbf{H}}\mathbf{D^{-1}} - \beta_{\mathbf{S}}\right]^{-1}\Delta\mathbf{R} \quad \Delta(\mathbf{X} - \mathbf{R}) = (\mathbf{D} - \mathbf{I})\mathbf{H} \tag{16}$$

These expressions are general and can hold even if there are more than two levels (see the Appendix). The harvest option (vector) is an inverse matrix multiplication of the change in cashflows at the boundaries.

$$\begin{array}{ccccc} \mathbf{H} & = & [\beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D}]^{-1} & & \mathbf{\Delta R} \\ \begin{bmatrix} C(R_2) \\ P(R_1) \end{bmatrix} & = & \begin{bmatrix} \beta_C & \beta_P D_P\left(R_2, R_1\right) \\ \beta_C D_C\left(R_1, R_2\right) & \beta_P \end{bmatrix}^{-1} & & \begin{bmatrix} R_2 \\ -R_1 \end{bmatrix} \end{array} \tag{17}$$

Equivalently a beta matrix incorporating future discounted beta, applied to the vector of options harvested, gives the change in cashflow vector.

$$\begin{array}{ccccc} \mathbf{\Delta R} & = & \beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D} & & \mathbf{H} \\ \begin{bmatrix} R_2 \\ -R_1 \end{bmatrix} & = & \begin{bmatrix} \beta_C & \beta_P D_P\left(R_2, R_1\right) \\ \beta_C D_C\left(R_1, R_2\right) & \beta_P \end{bmatrix} & & \begin{bmatrix} C(R_2) \\ P(R_1) \end{bmatrix} \end{array} \tag{18}$$

Either way it can be seen that smooth pasting at one boundary incorporates the beta of the immediate option but also *anticipates and incorporates* a discounted version or value weighted beta from the other boundary, e.g. $R_2 = \beta_C C(R_2) + \beta_P D_P(R_2, R_1) P(R_1)$. The same is true at the other rule R_1 where similar relationships hold for the seed vector $\mathbf{S}$ and $\mathbf{\Delta R}$.

The determinant of the inverse $[\beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D}]^{-1}$ is needed, which for the two level case is $\text{Det} = \beta_C\beta_P(1 - (\frac{R_1}{R_2})^{\beta_C - \beta_P})$.

For the two levels in this example, the solution for $\mathbf{\Delta}(\mathbf{X}(\mathbf{R}) - \mathbf{R})$ is shown below in equation (19) from the system matrix $\mathbf{M}(\mathbf{R}) = (\mathbf{D} - \mathbf{I})(\beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D})^{\mathbf{-1}}$ which includes this determinant.

$$\begin{array}{ccccc} \mathbf{\Delta}(\mathbf{X} - \mathbf{R}) & = & \mathbf{M}(\mathbf{R}) & & \mathbf{\Delta R} \\ \begin{bmatrix} X_2 - R_2 \\ R_1 - X_1 \end{bmatrix} & = & \frac{1}{\text{Det}} \begin{bmatrix} \beta_C(\frac{R_1}{R_2})^{\beta_C - \beta_P} - \beta_P & (\beta_C - \beta_P)(\frac{R_2}{R_1})^{\beta_P} \\ (\beta_P - \beta_C)(\frac{R_1}{R_2})^{\beta_C} & \beta_P(\frac{R_1}{R_2})^{\beta_C - \beta_P} - \beta_C \end{bmatrix} & & \begin{bmatrix} R_2 \\ -R_1 \end{bmatrix} \end{array} \tag{19}$$

For the decision costs explicitly, i.e. $\Delta(\mathbf{X}(\mathbf{R}) - \mathbf{R})$ it can be seen that although $\mathbf{X}(\mathbf{R})$ is possible to evaluate, the inverse $\mathbf{R}(\mathbf{X})$ (which we would like) is not possible to solve. Solving for $\mathbf{X}(\mathbf{R})$ in vector form gives equation (19).

$$\begin{bmatrix} X_2 - R_2 \\ R_1 - X_1 \end{bmatrix} = \frac{1}{\text{Det}} \begin{bmatrix} R_2(\beta_C(\frac{R_1}{R_2})^{\beta_C - \beta_P} - \beta_P) - R_1((\beta_C - \beta_P)(\frac{R_2}{R_1})^{\beta_P}) \\ R_2((\beta_P - \beta_C)(\frac{R_1}{R_2})^{\beta_C}) - R_1(\beta_P(\frac{R_1}{R_2})^{\beta_C - \beta_P} - \beta_C) \end{bmatrix} \tag{20}$$

or for the symmetric case $\beta_C = \mathbf{2}, \beta_P = -\mathbf{1}$ i.e. $\gamma = \mathbf{2}, \beta_C - \beta_P = \mathbf{3}$

$$\begin{bmatrix} X_2 \\ X_1 \end{bmatrix} = (2(R_1^2 + R_1 R_2 + R_2^2))^{-1} \begin{bmatrix} 4R_1^2 R_2 + R_1 R_2^2 + R_2^3 \\ 4R_1 R_2^2 + R_1^2 R_2 + R_1^3 \end{bmatrix} \tag{21}$$

Partial derivatives are needed for the numerical iteration search amongst optimal policies (not to optimise any one policy as with smooth pasting) so, we calculate the Jacobian[10] e.g. $\mathbf{J}(\mathbf{R}) = \begin{bmatrix} \frac{\partial X_2}{\partial R_2} & \frac{\partial X_2}{\partial R_1} \\ \frac{\partial X_1}{\partial R_2} & \frac{\partial X_1}{\partial R_1} \end{bmatrix} =$

$$\frac{1}{2}(R_1^2 + R_1 R_2 + R_2^2)^{-2} \begin{bmatrix} 4R_1^4 + 2R_1^3 R_2 + 2R_1 R_2^3 + R_2^4 & 3R_1^2 R_2(2R_1 + R_2) \\ 3R_1 R_2^2(R_1 + 2R_2) & R_1^4 + 2R_1^3 R_2 + 2R_1 R_2^3 + 4R_2^4 \end{bmatrix} \tag{22}$$

4 Iterating from $\mathbf{R}^0$ to $\mathbf{R}^\dagger$

It is important to re-iterate that for all sets of rules $\mathbf{R}$, the costs $\mathbf{X}(\mathbf{R})$ from equation (16) are associated with $\mathbf{R}$ being $\mathbf{X}$'s optimal policy in that the required conditions at the rules in $\mathbf{R}$ have been met. These are consistent with having maximised values by choosing $\mathbf{R}$ to match $\mathbf{X}$ (but not $\mathbf{X}^\dagger$).

What if we need to know this particular policy that generates $\mathbf{X}^\dagger \neq \mathbf{X}(\mathbf{R})$, i.e. how do we find $\mathbf{R}^\dagger$ that satisfies $\mathbf{X}(\mathbf{R}^\dagger) = \mathbf{X}^\dagger$ where $\mathbf{X}^\dagger$ are decision cost values *inputted as an iteration target*?

With the solution for $\mathbf{X}(\mathbf{R})$ as a function of $\mathbf{R}$ (equation 16) we now provide an algorithm to find $\mathbf{R}^\dagger$. The search proceeds along a path within the space of vectors of any $\mathbf{R}, \mathbf{X}$ toward the special $\mathbf{R}^\dagger, \mathbf{X}^\dagger$. Each point in the iteration tests an optimal policy, but only the final one will correspond to the target cost structure $\mathbf{X}^\dagger$.

The algorithm contains two parts; firstly the means to determine an initial rule estimate $\mathbf{R}^0$, from which $\mathbf{X}(\mathbf{R}^0)$ is direct. Secondly we show how to update these $\mathbf{R}^1, \mathbf{X}(\mathbf{R}^1)$, then $\mathbf{R}^2, \mathbf{X}(\mathbf{R}^2)$ etc. toward $\mathbf{R}^\dagger$ and $\mathbf{X}(\mathbf{R}^\dagger) = \mathbf{X}^\dagger$.

[10]In the limit as the two rules approach a common R, this Jacobian tends to a full two by two matrix of halves. In contrast the sensitivity of $X_2 - X_1 = \frac{(R_2 - R_1)^3}{2(R_1^2 + R_1 R_2 + R_2^2)}$ tends to zero in the difference $R_2 - R_1$ at zero which is consistent with the double smooth pasting used in Section 2. N.B. A matrix of second derivatives, the Hessian, determines the sufficient conditions for value maximisation at that set of rules in $\mathbf{R}$.

4.1 Initial $\mathbf{R}^0$ from asymptotic case

With only the information in a target $\mathbf{X}^\dagger$ and the structure of the problem (including betas that are fundamental constants for GBM), we need an initial rule $\mathbf{R}^0$ that determines an appropriate point to start the search for the final elements of $\mathbf{R}^\dagger$. In general, the fully reversible policy associated with $\mathbf{R} = \mathbf{X}$ is not a good start point for the iterative search.

For the two rule system in Section 3, we do this by supposing that each of the decisions rules in $\mathbf{R}^0$ were *terminal*, without further option seeds or role for onward discounting, i.e. $\mathbf{S} = \mathbf{0}$. Under this scenario, the relationship between value matching for harvest option values $\mathbf{H}$, changes $\mathbf{\Delta R}$ and decision costs $\mathbf{X}$ would be replaced by $\mathbf{H}^0$, $\mathbf{\Delta R}^0$, $\mathbf{X}^\dagger$ in equation (23) below.

$$\begin{array}{ccccc} \mathbf{H}^0 & = & \mathbf{\Delta R}^0 & - & \mathbf{\Delta X}^\dagger \\ \begin{bmatrix} C(R_2^0) \\ P(R_1^0) \end{bmatrix} & = & \begin{bmatrix} R_2^0 \\ -R_1^0 \end{bmatrix} & - & \begin{bmatrix} X_2^\dagger \\ -X_1^\dagger \end{bmatrix} \end{array} \tag{23}$$

Since all the harvested options are final, there are no betas from $\beta_{\mathbf{S}}^{\mathbf{0}}$ (i.e. ignores the effect of options at other rules), so equation (23) shows scaled smooth pasting conditions at $\mathbf{R}^0$ for each of four hypothetical terminal decisions.

$$\begin{array}{cccc} \beta_{\mathbf{H}}^0 & \mathbf{H}^0 & = & \mathbf{\Delta R}^0 \\ \begin{bmatrix} \beta_C & 0 \\ 0 & \beta_P \end{bmatrix} & \begin{bmatrix} C(R_2^0) \\ P(R_1^0) \end{bmatrix} & = & \begin{bmatrix} R_2^0 \\ -R_1^0 \end{bmatrix} \end{array} \tag{24}$$

Using equations (23 & 24) to eliminate $\mathbf{H}^0$ and solve for $\mathbf{R}^0$ from $\mathbf{X}^\dagger$ gives equation (25) (the difference matrix is self inverse i.e. $\Delta^{-1} = \Delta$).

$$\mathbf{R}^0 = (\mathbf{I} - \left(\beta_{\mathbf{H}}^{\mathbf{0}}\right)^{-1})^{-1}\mathbf{X}^\dagger = \begin{bmatrix} \frac{\beta_C}{\beta_C - 1} & 0 \\ 0 & \frac{\beta_P}{\beta_P - 1} \end{bmatrix} \begin{bmatrix} X_2^\dagger \\ -X_1^\dagger \end{bmatrix} = \begin{bmatrix} \frac{\beta_C}{\beta_C - 1} X_2^\dagger \\ \frac{\beta_P}{\beta_P - 1} X_1^\dagger \end{bmatrix} \tag{25}$$

Note that the rules in $\mathbf{R}^\dagger$ (which include further seeds) are not terminal so $\mathbf{R}^0$ only approximates $\mathbf{R}^\dagger$, i.e. $\mathbf{X}(\mathbf{R}^0) \neq \mathbf{X}^\dagger$.

However, if the threshold rules in $\mathbf{R}^\dagger$ are *widely separated* in level, it takes the diffusion Π longer to travel between them and actions at other levels are discounted more highly (i.e. discounts and option seeds tend to zero $\mathbf{D}, \mathbf{S} \to \mathbf{0}$) so we would expect $\mathbf{R}^0$ to be more accurate the more disperse the rules in $\mathbf{R}^{\dagger}$.[11] Even when iterating for more closely spaced rules, $\mathbf{R}^0$ gave good start points for finding $\mathbf{R}^{\dagger}$ in the numerical searches documented here. The first iteration step depends on the initial estimation error $\mathbf{X}^{\dagger} - \mathbf{X}(\mathbf{R}^0)$.

4.2 Updating

From any trial point $\mathbf{R}$, the local change in decision costs to improve the estimate of $R^{\dagger}$ are given by $\mathbf{dX}(\mathbf{R}) = \mathbf{J}(\mathbf{R})\mathbf{dR}$ where the Jacobian $\mathbf{J}(\mathbf{R})$ shown in equation (26) is derived from equations like (22).

$$\begin{aligned} \begin{bmatrix} dX_2 \\ dX_1 \end{bmatrix} &= \mathbf{dX}(\mathbf{R}) = \mathbf{J}(\mathbf{R})\mathbf{dR} = \begin{bmatrix} \frac{\partial X_2}{\partial R_2} & \frac{\partial X_2}{\partial R_1} \\ \frac{\partial X_1}{\partial R_2} & \frac{\partial X_1}{\partial R_1} \end{bmatrix} \begin{bmatrix} dR_2 \\ dR_1 \end{bmatrix} \\ \mathbf{R}^1 &= \mathbf{R}^0 + \mathbf{J}^{-1}(\mathbf{X}^{\dagger} - \mathbf{X}(\mathbf{R}^0)) \qquad \text{etc. until } \mathbf{R}^{\dagger} \text{ gives} \qquad \mathbf{X}(\mathbf{R}^{\dagger}) = \mathbf{X}^{\dagger} \end{aligned} \tag{26}$$

The second line of equation (26) also shows the updating equation that forms $\mathbf{R}^1$ from $\mathbf{R}^0$. This is repeated until convergence is achieved, typically in a few steps (see the Appendix for a worked example).

4.3 Constraints

Whilst updating it is important not to move from $\mathbf{R}^0$ to new rules $\mathbf{R}^1$ etc. that violate any of the embedded constraints, e.g. $R_2^1 > R_1^1$ (these imply constraints on $X_2^1 > X_1^1$). If this occurs, the discounts in $\mathbf{D}$ can exceed their regular bounds between zero and unity and lose economic meaning. For the iterations shown in the Appendix, neither $\mathbf{R}^0$ nor $\mathbf{R}^1$ etc. exited feasible regions and convergence occurred without violating constraints.

[11] In the limit as discounts become zero, $\mathbf{X}(\mathbf{R}^0) = \mathbf{X}^{\dagger}$ because

$$\begin{aligned} \mathbf{X}(\mathbf{R}) &= (\mathbf{I} + (\mathbf{D} - \mathbf{I}) \left[\beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D}\right]^{-1})\mathbf{R} \\ \lim_{\mathbf{D} \to \mathbf{0}} \mathbf{X}(\mathbf{R}^0) &= (\mathbf{I} - \left(\beta_{\mathbf{H}}^{\mathbf{0}}\right)^{-1})\mathbf{R}^0 = \mathbf{X}^{\dagger}. \end{aligned}$$

4.4 Stability of fixed point

To ensure that the fixed point of convergence is stable i.e. can be found via a cobweb method, the modulus of Jacobian entries were summed and tested e.g. $\left|\frac{\partial X_2}{\partial R_2}\right| + \left|\frac{\partial X_2}{\partial R_1}\right| < 1$)(this was confirmed for the numerics in the appendix). Updating and repeating worked well with a small number of steps achieving four decimal place accuracy.

4.5 Maxima, minima and points of inflection

Finally, to check that the first order conditions (smooth pasting) do not represent a minimum value (or point of inflection), for the values of $\mathbf{X}^{\dagger}$ used, e.g. $R_1^{\dagger}$ within $\mathbf{R}^{\dagger}$, was perturbed and option values at R_2 were seen to decline (those at R_1 remain smooth pasted). That occurred because these naturally convex options satisfy the second order conditions for optimality.

5 Two way options and reversibility

In this section we look at another way to use a pair of rules R_1, R_2 other than to move from one state to another with frictional reversibility. In the last section we considered having two rules converge at one common R.

Here we assume that at both of these rules reversible action occurs (i.e. that they each represent the convergence of two other unnamed rules).

Rather than use discounting, this reversibility at two rules requires (each of) value matching, smooth pasting *double smooth pasting* and therefore the gamma of options at R_1, R_2 will be required.

We will use R_1 to turn the first cap flow on and the second higher rule R_2 to switch the flow off, i.e. to exercise a short cap; the second cap has a higher strike than the first and together the two create a *collar or bull spread* between $\Pi = R_1$ and R_2.[12]

In the region between the two policy points, the flexibility value has functional form $K(\Pi)$ denoting a combination. This requires two way discounting

[12] Between the rules R_1 and R_2 the flexible flow is floating, with cap and floor terminology and boundaries x_1 and x_2 the definition of flexibility value here is consistent with $V(\Pi_t)$

to value it. Starting within the region $R_1 < \Pi < R_2$ if the process hits the upper rule, the flow is switched off by the party who is long this cap who effectively calls the value in return for paying a fixed sum X_2. However if the process hits the lower boundary first, the bull spread owner uses the combination option to put the floating value over to the party that is short the lower cap, this means using the combination option to put the floating and gain the fixed X_1 at R_1.

To reverse the exit from the region, reversibly, on hitting R_1 from below, the collar gains the floating value in return for paying X_1. When hitting the rule R_2 from above, X_2 is paid in return for obtaining the floating flow. Using $K(R_2)$ and $K(R_1)$ for the value of the combination flex value at these two rules ($P(R), C(R)$ retain their notation for one way options), value matching at these two levels implies equation (27).

$$\begin{array}{cccccc} \mathbf{H} & = & \mathbf{\Delta}(\mathbf{R}-\mathbf{X}) & + & \mathbf{S} \\ \begin{bmatrix} K(R_2) \\ C(R_1) \end{bmatrix} & = & \begin{bmatrix} X_2 - R_2 \\ R_1 - X_1 \end{bmatrix} & + & \begin{bmatrix} P(R_2) \\ K(R_1) \end{bmatrix} \end{array} \tag{27}$$

Again note that the fixed and floating changes $\mathbf{\Delta}(\mathbf{R}-\mathbf{X})$ and its beta product both equal $\mathbf{\Delta R}$ (where $\mathbf{\Delta} = \text{diag}[-1, 1]$).

We now need to extend the discounting relationship to accommodate two possible outcomes between R_2 when R_1 are met; the discount factors required are linear combinations of put and call discounts including *knock out* features.

5.1 Two way discounting

When flexibility indicates that there are two options that can be gained, one if the state variables increases and one if it decreases, then we label the dual option a combination, $K(\Pi)$. This is a *linear combination* of a special call

---

of:-

$$\int_t^{\infty} E_t^{RN}\left[\max(\pi_s, x_1) - \max(\pi_s, x_2)\right] e^{-r(s-t)} ds = \Pi_t + Floor(\pi_t, x_1) - Cap(\pi_t, x_2).$$

Above R_2 the flow is fixed but two floors are present, and beneath R_1, the flow is also fixed but two caps are present in long/short pairs.

on an upper payoff $K(R_2)$ and a special put on a lower one $K(R_1)$. This can be seen in equation (28) which shows $K(\Pi)$ as a discounted combination of $K(R_2)$ and $K(R_1)$.

$$K(\Pi) = D_C(\Pi, R_2, R_1) K(R_2) + D_P(\Pi, R_1, R_2) K(R_1) \qquad (28)$$

Unlike before, in equation (28) the discount factors D_C, D_P have three arguments; the first is the current level (dynamic, if labelled Π but static if fixed at a rule), the second is the level at which the discount factor achieves unity and the third the level at which it attains a zero value; that is to say these special options possess a *knock out* feature at the third rule in the argument. This condition of achieving zero worth ensures that when used, $K(\Pi)$ depends fully on one harvest value and not on the other in equation (28).

At $\Pi = R_2$ the compound option is worth $K(R_2)$ (and $K(R_1)$ at R_1), but we need the beta, weighted by the re–scaled value at each rule, i.e. $K'(R_2)R_2 = \beta_{K(R_2)}K(R_2)$ (etc. R_1). Unlike the simple options, for this compound option, its beta is not constant but depends upon a discounted combination of β_C, β_P and the separation of the two thresholds.

The two new discounts in equation (28) can themselves be simplified in equation (29) as linear combinations of the simple call and put discounts, here expressed for the GBM discount factors.

$$\begin{aligned} D_C(\Pi, R_2, R_1) &= nD_C(\Pi, R_2) - lD_P(\Pi, R_1) \\ D_P(\Pi, R_1, R_2) &= nD_P(\Pi, R_1) - mD_C(\Pi, R_2) \end{aligned} \qquad (29)$$

Equation (29) includes normalization constants $n = \left(1 - \left(\frac{R_1}{R_2}\right)^{\beta_C - \beta_P}\right)^{-1}$, $l = n \times \left(\frac{R_1}{R_2}\right)^{\beta_C}$ and $m = n \times \left(\frac{R_2}{R_1}\right)^{\beta_P}$ that depend on the envelope (R_2, R_1) but not the state Π. These ensure the complementary boundary conditions at both rules are met.[13]

[13]See Darling and Siegert [13] for the treatment of stopping time expectations conditioned on Π avoiding another rule. For Π_t within a pair of rules i.e. $R_1 < \Pi_t < R_2$, from time t stopping times $t_H > t$ at $\Pi_{t_H} = R_j$ to hit one, conditional upon avoiding the other,

5.2 Beta of options with knock–outs

To evaluate the betas of the knock–out options, we first take the dynamic equation (29) and apply differentiation and re–scale the result at $\Pi = R_2$ and $\Pi = R_1$, that is to say we require $D'_{C,P}(\Pi,.,.)\Pi$ as was derived for the one way case in equations (7, 8).

$$\begin{aligned} D'_C(\Pi, R_2, R_1)\,\Pi &= n\beta_C D_C(\Pi, R_2) - l\beta_P D_P(\Pi, R_1) \\ D'_P(\Pi, R_1, R_2)\,\Pi &= n\beta_P D_P(\Pi, R_1) - m\beta_C D_C(\Pi, R_2) \end{aligned} \tag{30}$$

Now the general dynamic versions of the call and put in the last equation can be specialised to $\Pi = R_2, R_1$ respectively so as to evaluate the betas there on $K(R_2), K(R_1)$.

$$\begin{aligned} \beta_{K(R_2)} &= D'_C(R_2, R_2, R_1)\,R_2 = \beta_C + \frac{\beta_C - \beta_P}{\left(\frac{R_2}{R_1}\right)^{\beta_C - \beta_P} - 1} > \beta_C \\ \beta_{K(R_1)} &= D'_P(R_1, R_1, R_2)\,R_1 = \beta_P + \frac{\beta_P - \beta_C}{\left(\frac{R_2}{R_1}\right)^{\beta_C - \beta_P} - 1} < \beta_P \end{aligned} \tag{31}$$

Due to the knock out features, these are more extreme betas than the simple constants β_C, β_P. These knock out features diminish as the separation R_2/R_1 increases; in the limit as the separation is very great, they revert back to the constant ones, otherwise the expected loss of the (short) option that is knocked out, increases the beta compared to the non knock out version.

---

are as follows:-

$$\begin{aligned} D_C(\Pi_t, R_2, R_1) &= E_t^{RN}\left[e^{-r(t_H - t)}\middle|\,\Pi_{t_H} = R_2, \min(\Pi_t..\Pi_{t_H}) > R_1\right] \\ D_P(\Pi_t, R_1, R_2) &= E_t^{RN}\left[e^{-r(t_H - t)}\middle|\,\Pi_{t_H} = R_1, \max(\Pi_t..\Pi_{t_H}) < R_2\right] \end{aligned}$$

They encompass one way factors as a special case (left panel with boundaries at $\infty, 0$) and two way bounds (right panels).

$$\begin{array}{lll} D_C(\Pi_t, R_2, 0) = D_C(\Pi_t, R_2) & D_C(R_1, R_2, R_1) = 0 & D_C(R_2, R_2, R_1) = 1 \\ D_P(\Pi_t, R_1, \infty) = D_P(\Pi_t, R_1) & D_P(R_1, R_1, R_2) = 1 & D_P(R_2, R_1, R_2) = 0 \end{array}$$

5.3 Gammas of options with knock–outs

For double smooth pasting, we also need the gammas defined in the next equation,

$$\begin{aligned} D''_C(\Pi, R_2, R_1)\,\Pi^2 &= n\gamma_C D_C(\Pi, R_2) - l\gamma_P D_P(\Pi, R_1) \\ D''_P(\Pi, R_1, R_2)\,\Pi^2 &= n\gamma_P D_P(\Pi, R_1) - m\gamma_C D_C(\Pi, R_2) \end{aligned} \tag{32}$$

so that

$$\begin{aligned} \gamma_{K(R_2)} &= D''_C(R_2, R_2, R_1)\,R_2^2 = \gamma_C + \frac{\gamma_C - \gamma_P}{\left(\frac{R_2}{R_1}\right)^{\beta_C - \beta_P} - 1} \gtrless \gamma_C \\ \gamma_{K(R_1)} &= D''_P(R_1, R_1, R_2)\,R_1^2 = \gamma_P + \frac{\gamma_P - \gamma_C}{\left(\frac{R_2}{R_1}\right)^{\beta_C - \beta_P} - 1} \gtrless \gamma_P. \end{aligned}$$

Again we get a basic beta with an envelope enhanced difference but since both gammas are positive, this does not amplify and can cancel out if the gammas are equal.

5.4 Smooth pasting with knockouts

We can use these betas to move from value matching in equation (27) to smooth pasting here.

$$\begin{array}{ccccc} \beta_{\mathbf{K}}\mathbf{H} & = & \mathbf{\Delta R} & + & \beta_{\mathbf{K}}\mathbf{S} \\ \begin{bmatrix} \beta_{K(R_2)}K(R_2) \\ \beta_C C(R_1) \end{bmatrix} & = & \begin{bmatrix} -R_2 \\ R_1 \end{bmatrix} & + & \begin{bmatrix} \beta_P P(R_2) \\ \beta_{K(R_1)}K(R_1) \end{bmatrix} \end{array} \tag{33}$$

This equation fixes the absolute scale of the options to that of the choices for $\mathbf{R}$. However, we need the double smooth pasting conditions for reversible action at R_1, R_2 to fix the relative sizes of the the options there. This involves the option gammas, and not the gamma of the floating leg which is zero.

5.5 Double smooth pasting

We need the double differential, double scaled for the gammas.

$$
\begin{aligned}
D''_C(\Pi, R_2, R_1)\,\Pi^2 &= n\gamma_C D_C(\Pi, R_2) - l\gamma_P D_P(\Pi, R_1) \\
D''_P(\Pi, R_1, R_2)\,\Pi^2 &= n\gamma_P D_P(\Pi, R_1) - m\gamma_C D_C(\Pi, R_2)
\end{aligned} \tag{34}
$$

Now the general dynamic versions of the call and put in the last equation can be specialised to $\Pi = R_2, R_1$ respectively so as to evaluate the betas there on $K(R_2), K(R_1)$.

$$
\begin{aligned}
\gamma_{K(R_2)} &= D''_C(R_2, R_2, R_1)\,R_2^2 = n\gamma_C - n\gamma_P\left(\frac{R_1}{R_2}\right)^{\beta_C-\beta_P} \\
\gamma_{K(R_1)} &= D''_P(R_1, R_1, R_2)\,R_1^2 = n\gamma_P - n\gamma_C\left(\frac{R_1}{R_2}\right)^{\beta_C-\beta_P}
\end{aligned} \tag{35}
$$

We now use these betas to move from value matching in equation (27) to smooth pasting here.

$$
\begin{array}{ccc}
\gamma_{\mathbf{K}}\mathbf{H} & = & \gamma_{\mathbf{K}}\mathbf{S} \\
\left[\begin{array}{c} \gamma_{K(R_2)}K(R_2) \\ \gamma_C C(R_1) \end{array}\right] & = & \left[\begin{array}{c} \gamma_P P(R_2) \\ \gamma_{K(R_1)}K(R_1) \end{array}\right]
\end{array} \tag{36}
$$

This fixes the relative level of external put call to the internal combination option. The put and call can be eliminated, leaving the end values of the combination option, this can be valued from the two smooth pasting conditions which include the levels R_1, R_2 which is also embedded in the definitions of the combination betas.

5.6 Other applications

Other extensions include constructing perpetual leader–follower type games with players maximizing their own claim conditional on others doing the same. These may involve smooth pasting conditions for each player at the level where their action is relevant, i.e. a complete set of smooth pasting conditions are jointly determined by players.[14] Even though such games may be time invariant, their policies need further investigation for optimality, unlike the single agent problems here.

6 Conclusion

Discount factors dependent on diffusion dynamics have been used before but with limited interaction and without a beta interpretation. We extend their use so that compound options can interact within a decision framework. This was done by separating the beginning (seed) and end of life (harvest) values for each option, placing them into vectors and solving with discount and beta matrices whose size and composition reflect the scale and form of flexibility present.

Value associated with the option to change cashflows can be represented using these discount factors because they capture two features of the stochastic dynamics. Firstly, discounts quantify an expectation of the time–value separation between policy rules when options are created and used. Secondly, they determine the beta of the option which is key to the smooth pasting optimal first order conditions at each rule. Second order conditions for reversibility were derived from gammas (beta times excess beta over one). When solving this as a linear system, for geometric processes it is easier to present smooth pasting multiplied by its policy threshold since this produces option values weighted by their betas and gammas.

It is easier to identify candidate policy rules first and use discounting and scaled smooth pasting conditions to form explicit solutions for option values and decisions costs. *Comparative statics* were also be derived in this process. Since the fixed decision costs only appear in one set of conditions (value matching), it is only possible to infer them last.

Having derived the option values, the analytical comparative statics were used in the iteration process. From a robust iteration start point, an efficient algorithm was developed to search among optimal policies for the one that

[14]To see how smooth pasting can encompass leader follower situations, see Bustamente [8] and [9] for inclusion of a shadow cost of pre–emption.

matches target decision costs. This facilitates the solution of such systems and expands the range of problems that can be tackled.

References

[1] Andrew B. Abel, Avinash K. Dixit, Janice C. Eberly, and Robert S. Pindyck. Options, the value of capital, and investment. *The Quarterly Journal of Economics*, 111(3):753–777, 1996.

[2] Andrew B. Abel and Janice C. Eberly. Optimal investment with costly reversibility. *Review of Economic Studies*, 63:581–593, 1996.

[3] David R. Alexander, Mengjia Mo, and Alan F. Stent. Arithmetic Brownian motion and real options. *European Journal of Operational Research*, 219:114–122, 2012.

[4] Luis H.R. Alvarez. Optimal exit and valuation under demand uncertainty: A real options approach. *European Journal of Operational Research*, 114:320–329, 1999.

[5] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(May–June):637–659, 1973.

[6] Michael John Brennan and Eduardo S. Schwartz. Evaluating natural resource investments. *Journal of Business*, 58(2):135–157, 1985.

[7] Michael John Brennan and Lenos Trigeorgis. *Project flexibility, agency, and competition: New developments in the theory and application of real options*. Oxford University Press, 2000.

[8] M. Cecilia Bustamente. The dynamics of going public. *Review of Finance*, 16:5776–618, 2012.

[9] M. Cecilia Bustamente. Strategic investment and industry risk dynamics. *Review of Financial Studies*, 28:297–341, 2015.

[10] Dennis Capozza and Yuming Li. The intensity and timing of investment: the case of land. *American Economic Review*, 84(4):889–904, 1994.

[11] John C. Cox, Stephen A. Ross, and Mark Rubinstein. Option pricing: a simplified approach. *Journal of Financial Economics*, 7:229–263, 1979.

[12] Eric Dahlgren and Tim Leung. An optimal multiple stopping approach to infrastructure investment decisions. *Journal of Economic Dynamics and Control*, 53(Apr):251–267, 2015.

[13] D. A. Darling and A. J. F. Siegert. The first passage problem for a continuous markov process. *Annals of Mathematical Statistics*, 24:624–639, 1953.

[14] Avinash K. Dixit. Entry and exit decisions under uncertainty. *Journal of Political Economy*, 97:620–638, 1989.

[15] Avinash K. Dixit and Robert S. Pindyck. *Investment under Uncertainty*. Princeton University Press, 1994.

[16] Avinash K. Dixit, Robert S. Pindyck, and Sigbjørn Sødal. A markup interpretation of optimal investment rules. *The Economic Journal*, 109(455):179–189, 1999.

[17] Bernard Dumas. Super contact and related optimality conditions. *Journal of Economic Dynamics and Control*, 15(4):675–685, 1991.

[18] Janice C. Eberly and Jan A. Van Mieghem. Multi-factor dynamic investment under uncertainty. *Journal of Economic Theory*, 75:345–387, 1996.

[19] Steinar Ekern. Entry and exit decisions with restricted reversibility. NHH Working Paper 93-08, 1993.

[20] Andrea Gamba and Nicola Fusari. Valuing modularity as a Real Option. *Management Science*, 55(11):1877–1896, 2009.

[21] Graeme Guthrie. Uncertainty and the trade off between scale and flexibility. *Journal of Economic Dynamics and Control*, 36(11):1718–1728, 2012.

[22] Holger Kraft, Eduardo Schwartz, and Farina Weiss. Growth options and firm valuation. *European Financial Management*, 24(2):209–2386, 2017.

[23] Bart Lambrecht. The timing and terms of mergers motivated by economies of scale. *Journal of Financial Economics*, 72:41–62, 2004.

[24] Bart M. Lambrecht. Real options in finance. *Journal of Banking and Finance*, 81(August):166–171, 2017.

[25] Robert L. McDonald and Daniel R. Siegel. Investment and the valuation of firms when there is an option to shut down. *International Economic Review*, 26(2):331–349, 1985.

[26] H.P. McKean. Appendix: A free boundary problem for the heat equation arising from a problem in mathematical economics. *Industrial Management Review*, 6:32–39, 1965.

[27] Robert C. Merton. The theory of rational option pricing. *Bell Journal of Economics*, 4(1):141–183, 1973.

[28] Stewart C. Myers. Determinants of corporate borrowing. *Journal of Financial Economics*, 5:147–176, 1977.

[29] Paulo J. Pereira and Artur Rodrigues. Investment decisions in finite lived monopolies. *Journal of Economic Dynamics and Control*, 46(Sept):219–236, 2014.

[30] Robert S. Pindyck. Irreversible investment, capacity choice and the value of the firm. *American Economic Review*, 78:969–985, 1988.

[31] Paul A. Samuelson. Rational theory of warrant pricing. *Industrial Management Review*, 6(Spring):13–31, 1965.

[32] Sudipto Sarkar and Fernando Zapatero. The trade-off model with mean reverting earnings: Theory and empirical tests. *The Economic Journal*, 113(490):834–860, 2003.

[33] Han T.J. Smit and L.A. Ankum. A real options and game-theoretic approach to corporate investment strategy under competition. *Financial Management*, 22(3):241–250, 1993.

[34] Sigbjørn Sødal. Entry and exit decisions based on a discount factor approach. *Journal of Economic Dynamics and Control*, 30(11):1963–1986, 2006.

[35] Lenos Trigeorgis. *Real options: Managerial flexibility and strategy in resource allocation.* MIT Press, 1996.

Appendix

In equation (37), the two discount factors in equation (29) are evaluated at threshold rules and inserted into the discount matrix that links seed and harvest vectors (from equation (27), also two simple discounts are used).

$$\begin{array}{ccc} \mathbf{S} & = & \mathbf{D} & \mathbf{H} \\ \begin{bmatrix} P(R_4) \\ K(R_3) \\ K(R_2) \\ C(R_1) \end{bmatrix} & = & \begin{bmatrix} 0 & D_P\left(R_4, R_3\right) & 0 & 0 \\ D_C\left(R_3, R_4, R_1\right) & 0 & 0 & D_P\left(R_3, R_1, R_4\right) \\ D_C\left(R_2, R_4, R_1\right) & 0 & 0 & D_P\left(R_2, R_1, R_4\right) \\ 0 & 0 & D_C\left(R_1, R_2\right) & 0 \end{bmatrix} & \begin{bmatrix} K(R_4) \\ P(R_3) \\ C(R_2) \\ K(R_1) \end{bmatrix} \end{array} \quad (37)$$

As well as the four rule value matching equation (27), we require definitions for smooth pasting matrices $\beta_{\mathbf{H}}, \beta_{\mathbf{S}}$ from equation (37) before the system can be solved. This is worked through in a numerical example with $\mathbf{R} = (4, 3, 2, 1)^\top$ and $\beta_C = 2, \beta_P = -1$.

Other arbitrarily large and more complex or nested structures are possible, but first we turn to the practical problem of finding $\mathbf{R}$ given a special or target value of $\mathbf{X}^\dagger$.

6.1 Two way beta matrices

Following the definition in equation (37) of the discount matrix with two way factors, in the first line of the next equation we break out a secondary matrix that carries the normalization constants l, m, n leaving one way factors only in the first matrix.

$$\mathbf{D} = \begin{bmatrix} 0 & D_P(R_4, R_3) & 0 & 0 \\ D_C(R_3, R_4) & 0 & 0 & D_P(R_3, R_1) \\ D_C(R_2, R_4) & 0 & 0 & D_P(R_2, R_1) \\ 0 & 0 & D_C(R_1, R_2) & 0 \end{bmatrix} \begin{bmatrix} n & 0 & 0 & -m \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -l & 0 & 0 & n \end{bmatrix} \quad (38)$$

For the values $\beta_C = 2, \beta_P = -1$ and rules $\mathbf{R} = (4, 3, 2, 1)^\top$ these evaluate to:-

$$\begin{bmatrix} 0.000 & 0.750 & 0.000 & 0.000 \\ 0.550 & 0.000 & 0.000 & 0.196 \\ 0.222 & 0.000 & 0.000 & 0.444 \\ 0.000 & 0.000 & 0.250 & 0.000 \end{bmatrix} = \begin{bmatrix} 0.000 & 0.750 & 0.000 & 0.000 \\ 0.563 & 0.000 & 0.000 & 0.333 \\ 0.250 & 0.000 & 0.000 & 0.500 \\ 0.000 & 0.000 & 0.250 & 0.000 \end{bmatrix} \begin{bmatrix} 1.016 & 0.000 & 0.000 & - \\ 0.000 & 1.000 & 0.000 & \\ 0.000 & 0.000 & 1.000 & \\ -0.063 & 0.000 & 0.000 & \end{bmatrix}$$

For the inverse, $\mathbf{G}$ is given by:-

$$\mathbf{G} = \begin{bmatrix} 0 & D_C(R_4, R_3) & D_P(R_4, R_2) & 0 \\ D_P(R_3, R_4) & 0 & 0 & 0 \\ 0 & 0 & 0 & D_C(R_2, R_1) \\ 0 & D_C(R_1, R_3) & D_P(R_1, R_2) & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{n} & -\widehat{l} & 0 \\ 0 & -\widehat{m} & \widehat{n} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and its numerical version

$$\begin{bmatrix} 0.000 & 2.211 & -0.974 & 0.000 \\ 1.333 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 4.000 \\ 0.000 & -1.105 & 2.737 & 0.000 \end{bmatrix} = \begin{bmatrix} 0.000 & 1.778 & 0.500 & 0.000 \\ 1.333 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 4.000 \\ 0.000 & 0.111 & 2.000 & 0.000 \end{bmatrix} \begin{bmatrix} 1.000 & 0.000 & 0.00 \\ 0.000 & 1.421 & -0.9 \\ 0.000 & -0.632 & 1.42 \\ 0.000 & 0.000 & 0.00 \end{bmatrix}$$

We presented the inverse[15] discount or *growth* matrix $\mathbf{G} = \mathbf{D}^{-1}$ that relates harvest to seed via $\mathbf{H} = \mathbf{GS}$. This depends on other normalization constants from R_3, R_2 i.e. the inner separation; $\widehat{l} = \widehat{n} \times D_P\left(R_3, R_2\right)$, $\widehat{m} = \widehat{n} \times D_C\left(R_2, R_3\right)$ and $\widehat{n} = \left(1 - D_C(R_2, R_3) D_P(R_3, R_2)\right)^{-1}$.

The dynamic discount factors in equation (29) that depend on the current state Π can also be treated this way in equation (39).

$$\mathbf{D}\left(\Pi\right) = \begin{bmatrix} 0 & D_P\left(\Pi, R_3\right) & 0 & 0 \\ D_C\left(\Pi, R_4\right) & 0 & 0 & D_P\left(\Pi, R_1\right) \\ D_C\left(\Pi, R_4\right) & 0 & 0 & D_P\left(\Pi, R_1\right) \\ 0 & 0 & D_C\left(\Pi, R_2\right) & 0 \end{bmatrix} \begin{bmatrix} n & 0 & 0 & -m \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -l & 0 & 0 & n \end{bmatrix} \tag{39}$$

Based on $\mathbf{V}(\Pi) = [P(\Pi), K(\Pi), K(\Pi), C(\Pi)]^\top$ a dynamic set of (time t option) values , equation (39) shows how $\mathbf{D}\left(\Pi\right)$ can carry dependency on Π to portray $\mathbf{V}(\Pi)$ dynamically so $\mathbf{V}\left(\Pi\right) = \mathbf{D}\left(\Pi\right)\mathbf{H}$. This collapses to $\mathbf{S} = \mathbf{DH}$ if the times in $\mathbf{V}\left(\Pi\right)$ are matched against seed rules.

The times (t) at which $\Pi_{(t)}$ encounters each row is different and these options do not exist at the same time but it is convenient to present them this way to calculate the required betas.

6.2 Betas from scaled differentials

We wish to evaluate the beta of a vector of values $\mathbf{V}\left(\Pi\right)$ each at an arbitrary time. This is done by applying the definition of the beta $V'(\Pi)\Pi = \Pi\partial V(\Pi)/\partial\Pi = \beta_V(\Pi)V(\Pi)$ to both sides of all vector elements in $\mathbf{V}\left(\Pi\right) = \mathbf{D}\left(\Pi\right)\mathbf{H}$. This is done in equation (40), a vector analogue of equation (2). Since the elements in $\mathbf{H}$ are fixed, they are treated as constant; the sensitivity of a vector of values depends only on the sensitivity of its discount

[15] The inverse discount or growth matrix $\mathbf{G}$ contains factors greater than one such as $D_C\left(R_2, R_1\right) = \left(R_2/R_1\right)^{\beta_C} > 1$ (because $R_2 > R_1$). The three argument, two direction growth factors also have magnitude greater than unity. Multiplying $\mathbf{GD}$ gives $\mathbf{I}$, some elements give unity straight away (one way options e.g. for the put $D_P\left(P_3, P_2\right) D_P\left(P_2, P_3\right) = 1$) but the two way options take more algebraic expansion before reducing to 1 or cancelling to 0.

matrix not its harvest values – this is a multivariate equivalent of previous specifications.

$$\mathbf{V}'(\Pi)\,\mathbf{\Pi} = \beta_{\mathbf{V}}(\Pi)\mathbf{V}(\Pi) = \mathbf{D}'(\Pi)\,\mathbf{H} \tag{40}$$

In the first row of the next equation, to determine $\mathbf{D}'(\Pi)$ we apply the beta operation to equation (39) (normalization l, m, n weights remain unchanged). Then to generate static $\mathbf{D}'$ in the second row of the next equation we specialize the choices of Π to seeds; this results in the GBM betas β_P, β_C appearing before their discounts in the second row of the next equation.

$$\begin{aligned}
\mathbf{D}'(\mathbf{\Pi})\mathbf{\Pi} &= \begin{bmatrix} 0 & \beta_P D_P(\Pi, R_3) & 0 & 0 \\ \beta_C D_C(\Pi, R_4) & 0 & 0 & \beta_P D_P(\Pi, R_1) \\ \beta_C D_C(\Pi, R_4) & 0 & 0 & \beta_P D_P(\Pi, R_1) \\ 0 & 0 & \beta_C D_C(\Pi, R_2) & 0 \end{bmatrix} \begin{bmatrix} n & 0 & 0 & -m \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -l & 0 & 0 & n \end{bmatrix} \\
\mathbf{D}'(\mathbf{R})\mathbf{R} &= \begin{bmatrix} 0 & \beta_P D_P(R_4, R_3) & 0 & 0 \\ \beta_C D_C(R_3, R_4) & 0 & 0 & \beta_P D_P(R_3, R_1) \\ \beta_C D_C(R_2, R_4) & 0 & 0 & \beta_P D_P(R_2, R_1) \\ 0 & 0 & \beta_C D_C(R_1, R_2) & 0 \end{bmatrix} \begin{bmatrix} n & 0 & 0 & -m \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -l & 0 & 0 & n \end{bmatrix}
\end{aligned}$$

and its numerical values:-

$$\begin{bmatrix} 0.000 & -0.750 & 0.000 & 0.000 \\ 1.164 & 0.000 & 0.000 & -0.624 \\ 0.540 & 0.000 & 0.000 & -0.635 \\ 0.000 & 0.000 & 0.500 & 0.000 \end{bmatrix} = \begin{bmatrix} 0.000 & -0.750 & 0.000 & 0.000 \\ 1.164 & 0.000 & 0.000 & -0.624 \\ 0.540 & 0.000 & 0.000 & -0.635 \\ 0.000 & 0.000 & 0.500 & 0.000 \end{bmatrix} \begin{bmatrix} 1.016 & 0.000 & 0 \\ 0.000 & 1.000 & 0 \\ 0.000 & 0.000 & 1 \\ -0.063 & 0.000 & 0 \end{bmatrix}$$

Specializing value $\mathbf{V}$ to seed rules in equation (40) takes the vector of values $\mathbf{V}(\Pi)$ to $\mathbf{S}$ and $\mathbf{V}'(\Pi)$ to $\mathbf{S}'$ which are the beta weighted values at the seed rules. This implies equation (41) from which a definition of the beta matrix $\beta_{\mathbf{S}}$ for options at their seed points is drawn as a product of $\mathbf{D}'$ (in the second row of the prior equation and $\mathbf{G}$.

$$\beta_{\mathbf{S}}\mathbf{S} = \mathbf{S}' = \mathbf{D}'\mathbf{H} = \mathbf{D}'\mathbf{G}\mathbf{S} \Rightarrow \beta_{\mathbf{S}} = \mathbf{D}'\mathbf{G} = \begin{bmatrix} -1.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 3.263 & -2.842 & 0.000 \\ 0.000 & 1.895 & -2.263 & 0.000 \\ 0.000 & 0.000 & 0.000 & 2.000 \end{bmatrix} \quad (41)$$

Similarly the beta operation applied to the growth matrix gives $\mathbf{G}'(\Pi) =$

$$\begin{bmatrix} 0 & \beta_C D_C(\Pi, R_3) & \beta_P D_P(\Pi_t, R_2) & 0 \\ \beta_P D_P(\Pi, R_4) & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta_C D_C(\Pi, R_1) \\ 0 & \beta_C D_C(\Pi, R_3) & \beta_P D_P(\Pi_t, R_2) & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{n} & -\widehat{l} & 0 \\ 0 & -\widehat{m} & \widehat{n} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0.000 & 5.368 & -4.079 & 0.000 \\ -1.333 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 8.000 \\ 0.000 & 1.579 & -3.053 & 0.000 \end{bmatrix} = \begin{bmatrix} 0.000 & 5.368 & -4.079 & 0.000 \\ -1.333 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 8.000 \\ 0.000 & 1.579 & -3.053 & 0.000 \end{bmatrix} \begin{bmatrix} 1.000 & 0.000 \\ 0.000 & 1.421 \\ 0.000 & -0.632 \\ 0.000 & 0.000 \end{bmatrix}$$

from which $\mathbf{G}'$ at harvest can be specialized and then derive for $\beta_{\mathbf{H}}$.

$$\beta_{\mathbf{H}}\mathbf{H} = \mathbf{H}' = \mathbf{G}'\mathbf{S} = \mathbf{G}'\mathbf{D}\mathbf{H} \Rightarrow \beta_{\mathbf{H}} = \mathbf{G}'\mathbf{D} = \begin{bmatrix} 0.000 & 0.750 & 0.000 & 0.000 \\ 0.550 & 0.000 & 0.000 & 0.196 \\ 0.222 & 0.000 & 0.000 & 0.444 \\ 0.000 & 0.000 & 0.250 & 0.000 \end{bmatrix} \quad (42)$$

Finally, the matrix whose inverse must be applied to the payoff vector $\mathbf{\Delta R}$

is given by:-

$$\mathbf{H} = [\beta_{\mathbf{H}} - \beta_{\mathbf{S}}\mathbf{D}]^{-1}\Delta\mathbf{R} = \begin{bmatrix} 2.048 & 0.000 & 0.000 & -0.762 \\ 0.000 & -1.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 2.000 & 0.000 \\ 0.190 & 0.000 & 0.000 & -1.048 \end{bmatrix}^{-1} \begin{bmatrix} -4 \\ 3 \\ 2 \\ -1 \end{bmatrix} = \begin{bmatrix} -1.371 \\ -1.107 \\ 0.479 \\ 0.477 \end{bmatrix}$$

$$\mathbf{S} = \left[\beta_{\mathbf{H}}\mathbf{D}^{-1} - \beta_{\mathbf{S}}\right]^{-1}\Delta\mathbf{R} = \begin{bmatrix} 2.048 & 0.000 & 0.000 & -0.762 \\ 0.000 & -1.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 2.000 & 0.000 \\ 0.190 & 0.000 & 0.000 & -1.048 \end{bmatrix}^{-1} \begin{bmatrix} -4 \\ 3 \\ 2 \\ -1 \end{bmatrix} = \begin{bmatrix} -0.830 \\ -0.661 \\ -0.093 \\ 0.120 \end{bmatrix}$$

$$\mathbf{\Delta X} = \Delta\mathbf{R} + \mathbf{S} - \mathbf{H} = \begin{bmatrix} -4 \\ 3 \\ 2 \\ -1 \end{bmatrix} + \begin{bmatrix} -0.830 \\ -0.661 \\ -0.093 \\ 0.120 \end{bmatrix} - \begin{bmatrix} -1.371 \\ -1.107 \\ 0.479 \\ 0.477 \end{bmatrix} = \begin{bmatrix} -3.459 \\ 3.446 \\ 1.429 \\ -1.357 \end{bmatrix}$$

6.3 Example path

For the target values $\mathbf{X}^{\dagger} = [4.0, -2.5, 2.5, -1.0]^{\top})$, the starting values in $\mathbf{R}^0$ were $[16, 2.5, 10, 1]^{\top}$ were used. These are policy rules consistent with $\mathbf{X}(\mathbf{R}^0) = [4.2481,\ -2.2131,\ 2.5676,\ -0.9257]^{\top}$. From this, the error to target $\mathbf{X}^{\dagger}$ is $\mathbf{dX} = [-0.2481,\ -0.2869,\ -0.0676,\ -0.0743]^{\top}$ which had the inverse Jacobian applied to generate $\mathbf{R}^1$ etc.

In Table (1), the sequence from $\mathbf{X}^{\dagger}$ and $\mathbf{R}^0(\mathbf{X}^{\dagger})$ is shown with $\mathbf{R}^1$, $\mathbf{X}(\mathbf{R}^1)$ etc. proceeding toward $\mathbf{X}(\mathbf{R}^3)$. Generating four decimal place accuracy for $\mathbf{X}^{\dagger}$, $\mathbf{R}^3 = [14.5803,\ 2.9261,\ 9.6718,\ 1.0913]^{\top}$ is taken as a close approximation of $\mathbf{R}^{\dagger}$ where $\mathbf{X}(\mathbf{R}^{\dagger}) = \mathbf{X}^{\dagger}$. The last line of Table (1) quantifies the sum of slopes for the fixed point stability criteria (all less than one).

	R_4 or X_4	R_3 or X_3	R_2 or X_2	R_1 or X_1
$\mathbf{X}^\dagger$	4.0000	−2.5000	2.5000	−1.0000
$\mathbf{\Delta}^{-1}\beta_{\mathbf{H}}^{\mathbf{0}}(\beta_{\mathbf{H}}^{\mathbf{0}}-1)^{-1}$	4	−1	4	−1
$\mathbf{R}^0\left(\mathbf{X}^\dagger\right)$	16.0000	2.5000	10.0000	1.0000
$\mathbf{X}(\mathbf{R}^0)$	4.2481	−2.2131	2.5676	−0.9257
$\mathbf{dX}$	−0.2481	−0.2869	−0.0676	−0.0743
$\mathbf{R}^1$	14.6479	2.9003	9.6759	1.0899
$\mathbf{X}(\mathbf{R}^1)$	4.0101	−2.4833	2.5008	−0.9989
$\mathbf{dX}$	−0.0101	−0.0167	−0.0008	−0.0011
$\mathbf{R}^2$	14.5806	2.9260	9.6718	1.0913
$\mathbf{X}(\mathbf{R}^2)$	4.0000	−2.4999	2.5000	−1.0000
$\mathbf{dX}$	0.0000	−0.0001	0.0000	0.0000
$\mathbf{R}^3$	14.5803	2.9261	9.6718	1.0913
$\mathbf{X}(\mathbf{R}^3)$	4.0000	−2.5000	2.5000	−1.0000
$\sum_{i=1}^4 \left\lvert \frac{\partial X_i}{\partial R_j} \right\rvert$	0.4463	0.7449	0.3837	0.8437

Table 1: A sample path iterating from R_j^0 to $R_j^\dagger$, i.e. from $-X_3, X_2 = (-2.2131, 2.5676)$ to $(-2.500, 2.500)$