A Least-Squares Monte Carlo Approach to the Pricing of GMWBs

Hongjun Ha* & Daniel Bauer
Department of Risk Management and Insurance. Georgia State University
35 Broad Street. Atlanta, GA 30303. USA

July 2015

Abstract

We consider a least-squares Monte Carlo (LSMC) simulation method to price a guaranteed minimum withdrawal benefit(GMWB). We implement LSMC and compare it to the usual grid method under Black-Scholes framework. In particular, we apply the regression-later algorithm to speed up convergence and get more reliable results. Then we extend the structure of GMWB into more complicated setting with stochastic interest rate, stochastic mortality and stochastic volatility and apply two LSMC algorithms.

Keywords: GMWB, Least-square Monte Carlo, Regression-now algorithm, Regression-later algorithm, Variable annuity.

*Corresponding author. Phone: +1-(404)-413-7490. Fax: +1-(405)-413-7499. E-mail addresses: hha6@gsu.edu (H. Ha); dbauer@gsu.edu (D. Bauer).

1 Introduction

Variable annuities (VAs) are insurance vehicles for policyholders to provide investment features and insurance features simultaneously. One of famous VAs is Guaranteed Minimum Withdrawal Benefits (GMWBs). Under the GMWB contract, the policyholder can make periodical withdrawals from his account formed by the initial payment, even though the account value reaches at zero. And surrender is also possible whenever the policyholder wants to withdraw the contract. The policyholder is paid the predetermined amount if he survives at the maturity. Death benefit rider is also possible.

To provide these benefits to policyholders, the insurer invests initial premium into risky asset, forms an account and deducts some fees ,so called option fee, from the account. The one of the most important obligations of insurer is that the option fee should be determined so as to provide agreed benefits to policyholders sufficiently. Therefore, the insurer is exposed to various risk such as financial risk, mortality risk and behavioral risk of policyholders, and the insurer should have a good system and process to determine an appropriate price of GMWB not to be forced into insolvency.

When we consider the pricing of GMWBs, the policyholder's behavior is important factor. Since policyholder's withdrawal behavior directly affects the value of GMWB, it is essential to assume a reasonable policyholder's behavior to get a proper price.

The simplest assumption on policyholder's behavior is *static* withdrawal behavior. Under this assumption, policyholder necessarily makes static withdrawal regardless of financial performance and mortality. In this case, it is easy to price GMWBs through simulation even though if we assume multiple stochastic factors. Moreover, it is possible to use option pricing theory (see **?**).

The second possible assumption is *mixed* behavior. Under the mixed behavior, policyholder chooses between static withdrawal and surrender based on which choice maximizes value of his contract. Therefore, pricing problem boils down to find optimal stopping time. By comparing surrender value to continuation value at time of withdrawal, he decides to surrender or not. Therefore, it is important to compute the continuation value to determine the time of surrender. To solve this problem, it is possible to use binomial framework or the *Least Squares Monte Carlo*(LSMC) algorithm (see **?**). In the LSMC algorithm, the continuation value is replaced by linear combination of basis function and associated coefficients are estimated least- squares method. As seen in **?**, it is equivalent to price an American style derivatives.

The most complicated assumption from a perspective of the insurer is *fully dynamic* behavior. Under this assumption, policyholder determines his amount of withdrawal or surrender time to maximize value of his contract. To find optimal amount of withdrawal, policyholder solves *dynamic programming* at each withdrawal time. Intuitively, the dynamic strategy assumption yields the largest value of contract and so produces the largest value of liability involved to GMWBs. In the risk management context, it is important to consider future cash flows which maximizes the value of liability of contract. Therefore, it becomes more required to assume the dynamic strategy since it gives insurer the worst scenario. The biggest difference between mixed strategy and fully dynamic strategy is that admissible solutions for the policyholder are not binary under fully dynamic behavior. For instance, he is able to choose among "not withdrawal","guaranteed withdrawal" or "surrender" if he acts based on fully dynamic behavior. Under these admissible solutions, we should compute value of value function at each solution. On the other hand, we care about the value of value function at “not withdrawal” only under mixed behavior. This facts makes one to have some numerical methods to calculate expectation of value function which are not easy to obtain.

To price GMWBs under the dynamic behavior assumption, most researchers and practitioners exploit grid algorithm (See e.g. **?**). One of problems when using grid algorithm, we are usually trapped in *curse of dimension*. So it is common to assume that there are two state variable to make a problem doable. For instance, in **?**, the authors assume the Black-Scholes framework and implement their model under two state variables. In **?**, they implement grid algorithm to find optimal withdrawal strategy under Lévy processes framework. However, the dimension of state variables is also two dimension. In the theory, it is possible to find a solution of dynamic programming under $m > 2$ where m is the dimension of state variables. In practice, however, it is computationally expensive to solve it numerically. If one wants to assume stochastic volatility of reference assets, stochastic mortality , stochastic interest rate, etc to have market consistent value of GMWB contracts, the grid algorithm is not a sensible choice since computation time is too long.

Recently, some authors apply LSMC to overcome the dimension problem and price GMWBs. If one implements LSMC to solve the dynamic programming, the problem becomes dimensional free and is closely related to how many sample paths and how many basis functions should be used. In **?**, they approximates the expectation of value function by replacing it with a linear combination of basis functions under stochastic volatility model. After having approximated expectation, they solve the dynamic programming where number of admissible solutions is three.

In most applications, the LSMC approximates the expectation of value function. This technique is general but sill requires many sample paths to have a reliable results. That is, the speed of convergence is usually slow. Moreover, the algorithm can be inefficient when stochastic interest rate and stochastic mortality are considered since the correlation between value function and them becomes low if time interval for simulation is large.

In **?**, the authors propose an alternative method to the usual LSMC. In the paper, they call the usual LSMC the regression-now method. Rather than approximating the expectation, they develop the algorithm which approximates the value function itself by LSMC, called regression-later method. Then taking expectation yields an approximation of the expectation if some conditions hold. In **?**, they show that estimates from regression-later converges faster than them of regression-now by applying LSMC to energy derivatives. In **?**, they provide proofs why regression-later is superior to regression-now under some conditions.

In this paper, we apply regression-now and regression-later to price GMWBs under Black-Scholes framework with deterministic mortality to verify if LSMC works properly by comparing them to grid method. Then we extend the model including stochastic interest rate, stochastic volatility and stochastic mortality since the improvement of mortality trend and nature of long-term contract are important factors for prices of insurance contracts. To apply LSMC methods, one should assume finite possible withdrawals from uncountable continuous admissible set and generate sample paths of underlying state variables. In the case of Guaranteed Minimum Life Benefits (GMLBs), it is known that there are only three optimal *bang-bang* solutions, not withdrawal, guaranteed amount of withdrawal, and surrender. However, it is not necessarily true in GMWBs contexts. So we assume finite set of possible withdrawal which is not trivial.

In most applications, it is enough to generate sample paths of market risk factors to make variations for relevant quantities to apply least-squares algorithm. Therefore, most researchers assume that generated withdrawals are guaranteed contractual amount or zero withdrawal. However, in our setting, the static withdrawals or zero withdrawal can not make variations on the guarantee account since our setting dose not allow to normalize the price system with respect to value of guarantee account. So we develop an alternative way to generate sample paths to make enough variation to apply the least-squares algorithm.

The remainder of this paper is organized as follows: Section **??** includes set up and price formulation of GMWBs, and related dynamic programming. Section **??** introduces the LSMC algorithm in the sense of regression-now and regression-later separately. Section **??** provides examples of LSMC algorithm. Section **??** concludes.

2 Set up, Price and Dynamic Programming of GMWB

To price GMWBs under fully dynamic behavior, the dynamic programming should be derived. To find specific form of dynamic programming, stochastic differential equations are specified for state variables in this section. In subsection **??**, we provide the underlying assumptions to set up and price GMWBs. The following subsection **??** introduces the dynamic programming to find optimal amount of withdrawal.

2.1 Structure of GMWB and Price

We assume that the policyholder whose age is x enters into a GMWB contract. He pays initial lump sum premium, P_0, for future benefits. The initial premium is invested into risky asset whose price at time t is defined S_t and forms a *personal account* for policyholder. The policyholder has the right to withdrawal money from his personal account after some periods. The withdrawal is assumed to be only possible at t_i, $i = 1, 2, ..., n-1$ with

$$0 = t_0 < t_1 < \cdots < t_{n-1} < t_n = T,$$

where T denotes the maturity of GMWB.

On the other hand, the insurer provides a *guarantee account* to the policyholder which quantifies the possible *guaranteed* total amount of withdrawals for the policyholder for remaining life of the contract regardless of the performance of risky asset. Therefore, if the personal account reaches zero and the guarantee account is larger than zero, the policyholder can still withdraw.

We introduce the law of motions of market risk factors over time. The dynamics of market risk factors are assumed to satisfy the following stochastic differential equations under the $\mathbb{Q}$ measure (See **?**),

$$dS_t = r_t S_t dt + \sigma_t S_t d\mathbf{LW}_t^{(1)} \quad (1)$$

$$d\sigma_t^2 = \kappa(\theta - \sigma_t^2)dt + \nu\sigma_t^2 d\mathbf{LW}_t^{(2)} \quad (2)$$

$$dr_t = a(b - r_t)dt + \gamma\sqrt{r_t} d\mathbf{LW}_t^{(3)}, \quad (3)$$

where, r_t is the time dependent risk-free interest rate, σ_t is the volatility of risky asset, κ is the speed of mean reversion for variance, θ is the long-term mean of variance, ν is the volatility of variance, a is the speed of mean reversion for risk-free interest rate, b is the long-term mean of risk-free interest rate, γ is the volatility of risk-free interest rate, $\mathbf{L}$ is the low triangular matrix from the Cholesky decomposition of 3×3 correlation matrix for three dimensional Brownian motion, $\mathbf{LW}_t^{(1)}$, $\mathbf{LW}_t^{(2)}$ and $\mathbf{LW}_t^{(3)}$ are the first, second and third component of $\mathbf{LW}_t$, and $\mathbf{W}_t$ is the three dimensional independent standard Brownian motion.

We specify the law of motion for the personal account and guaranteed account of GMWB. We follow **?** for notations and structure of GMWB. Let $X_{t_i}^-$ be the price of personal account at time t_i before making withdrawal. After withdrawal, the value of personal account is defined $X_{t_i}^+$. w_{t_i} denotes amount of withdrawal at time t_i. Also insurer deduct continuously guarantee fee, ϕ, to cover cost for providing option feature embedded in the contract to policyholder. Then we assume the following laws of motions of quantities,

$$X_{t_i}^- = X_{t_{i-1}}^+ \frac{S_{t_i}}{S_{t_{i-1}}} e^{-\phi(t_i - t_{i-1})}, \qquad i = 1, 2, ..., n \tag{4}$$

$$X_{t_i}^+ = \max\left(0, X_{t_i}^- - w_{t_i}\right), \qquad i = 1, 2, ..., n-1 \tag{5}$$

with $X_{t_0}^+ = P_0$.
We provide more explanation about the amount of withdrawal possible at time t_i. The policyholder can make withdrawal from his personal account even though the personal account is depreciated as long as the guarantee account dose not reach zero before making withdrawal. We assume that possible amount of withdrawal at time t_i is,

$$0 \le w_{t_i} \le \max\left(X_{t_i}^-, \min\left(g_{t_i}, G_{t_i}\right)\right), \tag{6}$$

where G_{t_i} is the guarantee account value at time t_i and g_{t_i} is the guaranteed contractual amount of withdrawal. Constraints (**??**) is general form of possible withdrawal amount. In other research, the constraint for withdrawal is relatively simple compared to our setting (See e.g. **?** and **?**). Note that we allow the policyholder to surrender contract before the maturity by permitting $w_{t_i} = X_{t_i}^-$ or $w_{t_i} = G_{t_i}$ in inequality (**??**). Hence, the policyholder should determine the optimal stopping time, τ, to maximize value of his contract. After withdrawal, the guarantee accounts is updated also based on the amount of withdrawal. We assume that the law of motion for guarantee account is given by,

$$G_{t_{i+1}} = \begin{cases} \max\left(0, G_{t_i} - w_{t_i}\right), & w_{t_i} \le g_t \\ \min\left(\max\left(0, G_{t_i} - w_{t_i}\right), \frac{X_{t_i}^+}{X_{t_i}^-} G_{t_i}\right), & w_{t_i} > g_t \end{cases} \tag{7}$$

with $i = 1, 2, ..., n-1$ and $G_{t_1} = P_0$. From now on, denote $\mathcal{W} = (w_{t_1}, ..., w_{t_{n-1}})$ an arbitrary withdrawal strategy and $\mathcal{A}$ denotes set of possible $\mathcal{W}$. If $\tau \le t_{n-1}$, we assume that $w_{t_i} = 0$ for $t_i > \tau$.

Nevertheless the amount and timing of withdrawal are determined by the policyholder, it is common to put penalties in the forms of fee, if amount of withdrawal exceed some threshold or withdrawals happen too early. Therefore, the cash amount belonging to the policyholder at time t_i may be different from w_{t_i}. Let $C(t_i, w_{t_i})$ be the cash amount paid to the policyholder after determining the amount of withdrawal at time t_i. We assume

$$C(t_i, w_{t_i}) = w_{t_i} - \mathrm{fee}_{t_i}^I - \mathrm{fee}_{t_i}^R \tag{8}$$

$$\mathrm{fee}_{t_i}^I = ep_{t_i} \times \max(0, w_{t_i} - \min(g_{t_i}, G_{t_i})) \tag{9}$$

$$\mathrm{fee}_{t_i}^R = pg_{t_i} \times (w_{t_i} - \mathrm{fee}_{t_i}^I) \mathbb{1}_{\{x + t_i < 59.5\}}, \tag{10}$$

where $\mathbb{1}_A$ is an indicator function which is one if A is true, and zero otherwise, ep_{t_i} and pg_{t_i} are percentage of penalty. In (**??**), the fee is applied to the withdrawal if it is greater than threshold, $\min(g_{t_i}, G_{t_i})$, and the another fee (**??**) is also applied when withdrawal is made too early, before 59.5 years old.

If we consider a mortality of the policyholder, it is natural to provide a death benefit rider for the policyholder. We define death benefit, D_{t_i}, which is paid at t_i, $i = 1, 2, ..., n$ if death of policyholder occurs during $(t_{i-1}, t_i]$. D_{t_i} may be constant, $X_{t_i}^-$ or guaranteed. Since it is common to load Guaranteed Minimum Death Benefit (GMDB) riders on GMWBs, we assume that

$$D_{t_i} = \max(X_{t_i}^-, G_{t_i}) \tag{11}$$

If the policyholder survives until to the maturity, the policyholder may have, $V(T)$, which is given by

$$V(T) = \max\left(X_T^-,\ \min(g, G_T)\right). \tag{12}$$

The stochastic mortality is assumed to be independent of financial market factors. However, the assumption is easily solved in LSMC framework. ${}_tp_x$ denotes the probability that an x-year old policyholder survives until to $x+t$. ${}_tq_x$ denotes the probability that an x-year old policyholder dies before $x+t$. In actuarial application, it is important to reflect the current improvement of mortality trend (See e.g. **?** or **?**). We follow stochastic Makehem law of mortality in **?**. $\mu_t(x)$ denotes the x-year policyholder's force of mortality at $x+t$ defined by $\mu_t(x) = -\left(\frac{d}{dt}{}_tp_x\right)/{}_tp_x$. We transform $\mu_x(t)$ in **?** into simpler form,

$$\mu_t(x) = (A + c\exp(dx)) \times Z_t^{(2)} \tag{13}$$

where A, c, and d are some positive constants, $Z_t^{(2)}$ is the second component of $\mathbf{Z}_t$. And the diffusion process of $\mathbf{Z}_t$ is given under $\mathbb{P}$ measure by

$$d\mathbf{Z}_t = \left\{\begin{bmatrix}2\xi_1 e + \xi_2 e^2\\ -\xi_1\end{bmatrix} + \begin{bmatrix}-2e & -e^2\\ 1 & 0\end{bmatrix}\right\}dt + \begin{bmatrix}(1-2ef)\times\sqrt{Z_t^{(2)}}\\ e\times\sqrt{Z_t^{(2)}}\end{bmatrix}dB_t$$

where e and f are a positive constant, B_t is an independent standard Brownian motion from market risk factors. Then, the survival probability, ${}_tp_x$ is given by

$${}_tp_x = e^{-\int_0^t \mu_s(x)ds} \tag{14}$$

Finally, we are ready to formulate the price of GMWB. Let $V(0)$ be the price of future cash flow of GMWB determined by the fully dynamic strategy at t_0. Then, the price of contract, $V(0)$, is given by

$$V(0) = \sup_{\mathcal{W}\in\mathcal{A}} \mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\Bigg[\underbrace{\sum_{i=1}^{n-1\wedge\tau} e^{-\int_0^{t_i} r_s ds}{}_{t_i}p_x C(t_i, w_{t_i}) + {}_Tp_x e^{-\int_0^T r_s ds} V(T)\mathbb{1}_{\{T\geq\tau\}}}_{\text{Survival Benefit}}$$
$$+ \underbrace{\sum_{j=1}^{n\wedge\tau} e^{-\int_0^{t_j} r_t dt}{}_{j-1}p_x\, q_{x+j-1} D_{t_j}}_{\text{Death Benefir}}\Bigg], \tag{15}$$

where $\mathbb{P} \times \mathbb{Q}$ is a product measure.

To find $V(0)$ in (??), $\mathcal{W}^*$, called optimal policy, which maximizes the contract should be determined. Usually, it is not possible to find $V(0)$ directly and analytically. The methodology to obtain optimal policy is well known in economic or mathematical literature. In the following section ??, we shortly introduce a *dynamic programming* to find the optimal policy.

2.2 Dynamic Programming

To find an optimal withdrawal strategy and (??) is equivalent to solve the following *dynamic programming* (??) recursively (See e.g. ?). At time $t_i, (i = 1, 2, ..., n-1)$, the policyholder's problem is

$$\begin{aligned}
&V_{t_i}(X_{t_i}^-, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)) = \max_{w_{t_i}} \quad c(t_i, w_{t_i}) + \\
&\quad \mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[e^{-\int_{t_i}^{t_{i+1}} r_s ds} \left\{ p_{x+t_i} V_{t_{i+1}}(X_{t_{i+1}}^-, G_{t_{i+1}}) + q_{x+t_i} D_{t_{i+1}} \right\} \middle| X_{t_i}^-, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x) \right], \\
&\text{subjec to} \quad 0 \le w_{t_i} \le max(X_{t_i}^-, \min(g, G_{t_i})), \\
&V_{t_N}(X_{t_N}^-, G_{t_N}) = \max(X_{t_N}^-, \min(g, G_{t_N})) \qquad (16)
\end{aligned}$$

A challenging part to solve (??) is to calculate the expectation. Usually, it is not possible to get the expectation in the closed form due to unknown $V_t(\cdot)$ and multidimensional integration. Instead, we rely on numerical techniques. To get the value of expectation numerically, we can take advantage of the gird method by discretizing state space. However, this method is doable and effective when the dimension of problem is low (See e.g. ?). In the case that one wants to include multiple financial risks and insurance risks such as mortality risk and operation risk, however, the previous method is usually trapped by the *curse of dimension*. In particular, we need to solve *five* dimensional dynamic programming problem under the previous setting in Section ??. For instance, if we discretize the space of five states by five hundred grid points evenly, the number of functional equations to be solved at time t_i is 3.125×10^{13}. Moreover, since it the maturity of contract is not short, the problem becomes worse. To overcome this shortcoming of the methods we adapt the *Least-Squares Monte Carlo* (LSMC) method which is dimension-free algorithm to approximate the expectation. All details about LSMC are explained in Section ??.

3 LSMC Algorithm

From now on, the considered space is assumed to be the separable Hilbert space, $\mathcal{H}$. We also assume that the expectation in (??) is an element of $\mathcal{H}$. Therefore, there is at least one complete set of basis function for the space and the expectation can be approximated by linear combination of some complete basis functions. This is a main intuition of LSMC.

In the LSMC algorithm, the conditional expectation is approximated (*regression-now*) by the linear combination of basis functions. In the LSMC with *regression-later*, the inside part in the expectation is approximated by the linear combination of basis functions and then the expectation is replaced with expectation of approximated inside part. If one say the LSMC algorithm, this is usually the regression-now approach. The second approach is first introduced in ?. Applications can be found in ? and ?. Regardless of methods, two LSMC methods are free from dimension of problem. They only matter number of simulations, number of basis functions, kind of basis function and other some conditions which are not too restrictive. Of course, there are the pros and cons of each method. The first approach is always possible to implement if minimum requirements are satisfied although second approach requires more compared to the first. On the other hand, the second approach converges faster than the first approach since it uses $\mathcal{F}_t$ information directly to approximate $\mathcal{F}_t$-measurable function. Rigorous analysis about regression-now can be found in **?**. For regression-later, see **?** and **?**. In the next two subsection, we introduce two approach to approximate the expectation in (**??**).

3.1 Regression-now algorithm

At time t_i, available information on state variables are $X^-_{t_i}$, G_{t_i}, r_{t_i}, σ_{t_i} and $\mu_{t_i}(x)$. The expectation in (**??**) is function, f, of these information. So, in the regression-now approach, the expectation is approximated by the following form,

$$\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[e^{-\int_{t_i}^{t_{i+1}} r_s ds}\left\{p_{x+t_i}V_{t_{i+1}}\left(X^-_{t_{i+1}}, G_{t_{i+1}}, r_{t_{i+1}}, \sigma_{t_{i+1}}, \mu_{t_{i+1}}(x)\right) + q_{x+t_i}D_{t_{i+1}}\right\}\middle| X^-_{t_i}, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right]$$

$$:= f\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{x+t_i}\right) = \sum_{j=1}^{\infty} \alpha_j^{t_i} \times e_j\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right) \quad (17)$$

$$\approx \sum_{j=1}^{M} \alpha_j^{t_i} \times e_j\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right) \quad \text{(First Approximation)} \quad (18)$$

$$\approx \sum_{j=1}^{M} \hat{\alpha}_j^{t_i} \times e_j\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right) \quad \text{(Second Approximation)} \quad (19)$$

where $(e_j)_{j=1}^M$ is set of basis functions, M is the degree of approximation, and $\alpha_j^{t_i}$, $j = 1, 2, ..., M$ is the associated coefficient to the jth basis function. Note that the function in (**??**) is the function of $\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{x+t_i}\right)$, not $\left(X^-_{t_i}, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{x+t_i}\right)$. In (**??**), the equality is true since we assume that there is a set of complete basis functions. Famous basis functions are simple polynomials, Hermite polynomials, Legendre polynomials and Chebyshev polynomials etc. Also it is possible to make them orthonormalized to speed up the convergence. This functional form is approximated in the first approximation (**??**). After setting up basis functions and M, the next step is to estimate $(\alpha_j^{t_i})_{j=1}^M$ which is unknown in (**??**). As the name of LSMC suggests, the coefficients are approximated via the least-squares method which yields $(\hat{\alpha}_j^{t_i})_{j=1}^M$. Then the final estimator for the expectation is given in (**??**). The convergence of (**??**) as $N \to \infty$ and $M \to \infty$ is found in **?**. More precisely, the regression-now algorithm to solve the (**??**) is the following,

- Initiate S_{t_0}, r_{t_0}, σ_{t_0}, $\mu_{t_0}(x)$, $X^+_{t_0}$, and G_{t_1}.
- For t_i, $i = 1, 2, ..., n$
  1. Generate $S^k_{t_i}$, $r^k_{t_i}$, $\sigma^k_{t_i}$, and $\mu^k_{t_i}(x)$, $k = 1, 2, ..., N$, where the superscript denotes kth simulation and N is the total number of simulation.
  2. Calculate $X^{-,k}_{t_i}$, $k = 1, 2, ..., N$.

3. Generate $w_{t_i}^k$, $k = 1, 2, ..., N$
4. Update $X_{t_i}^{+,k}$ and $G_{t_{i+1}}^k$, $k = 1, 2, ..., N$.
5. If $i \neq n$, go back to the **??**. Otherwise, go to the next step.
6. Drop $w_{t_n}^k$, $k = 1, 2, ..., N$

- Calculate $V_{t_n}(X_{t_n}^{-,k}, G_{t_n}^k) := \hat{V}_n^{k,now}(X_{t_n}^{-,k}, G_{t_n}^k, r_{t_n}^k, \sigma_{t_n}^k, \mu_{t_n}^k(x))$, $k = 1, 2, ..., N$
- At t_i, $i = n-1, ..., 1$

1. Regress
$e^{-r_s^k(t_{i+1}-t_i)} \left\{ e^{\mu_s^k(x)(t_{i+1}-t_i)} \hat{V}_{t_{i+1}}^{k,now}(X_{t_{i+1}}^{-,k}, G_{t_{i+1}}^k, r_{t_{i+1}}^k, \sigma_{t_{i+1}}^k, \mu_{t_{i+1}}^k(x)) + \left(1 - e^{\mu_s^k(x)(t_{i+1}-t_i)}\right) D_{t_{i+1}}^k \right\}$
on $\left(e_j\left(X_{t_i}^{+,k}, G_{t_{i+1}}^k, r_{t_i}^k, \sigma_{t_i}^k, \mu_{t_i}^k(x)\right)\right)_{j=1,...,M,\ k=1,...,N}$ and obtain $\hat{\alpha}_j^{t_i}$, $j = 1, 2, ..., M$.
2. Solve

$$\hat{V}_{t_i}^{k,now}(X_{t_i}^{-,k}, G_{t_i}^k, r_{t_i}^k, \sigma_{t_i}^k, \mu_{t_i}^k(x)) = \max_{w_{t_i}} w_{t_i} - \text{fee}_{t_i}^I - \text{fee}_{t_i}^R$$
$$+ \sum_{j=1}^{M} \hat{\alpha}_j^t \times e_j\left(X_{t_i}^{+,k}, G_{t_{i+1}}^k, \sigma_{t_i}^k, r_{t_i}^k, \mu_{t_i}^k(x)\right) \tag{20}$$

subject to

$$0 \leq w_{t_i} \leq \max(X_{t_i}^{-,k}, \min(g, G_{t_i}^k))$$

for $k = 1, 2, ..., N$.
3. If $i \neq 1$, go back to **??**. Otherwise, go to the next step.

- The price of GMWB at $t = 0$

$$\hat{V}_{now}(0) = \frac{1}{N}\sum_{k=1}^{N} e^{-r_{t_0}^k(t_1-t_0)} \left\{ e^{\mu_{t_0}^k(x)(t_1-t_0)} \hat{V}_{t_1}^{k,now}(X_{t_1}^{-,k}, G_{t_1}^k, \sigma_{t_1}^k, r_{t_1}^k, \mu_{t_1}^k(x)) + \left(1 - e^{-\mu_{t_0}^k(x)(t_1-t_0)}\right) D_{t_1}^k \right\}$$

As seen the above algorithm, the explanatory variables for regression are current information. This is why the algorithm is called regress-now method. There are some cautions when implementing the algorithm. The first one is that we need to discretize the continuous admissible set in (**??**) into the discrete set to make algorithm doable. In Guaranteed Minimum Life Benefits (GMLBs), it is known that there are only three possible *bang-bang* solution, "do not withdraw", "withdraw g" and "withdraw $X_{t_i}^{-}$" (see e.g. **?** or **?**). But it is difficult to generalize the optimal bang-bang solution in GMWBs. So we assume that there is finite optimal set of solution in Subsection **??**.

The second one is how to generate $w_{t_i}^k$. In **?**, the exercise does not occur until to the maturity when pricing American style derivatives. Similarly, arbitrary withdrawal in **?** is g since it is possible to give variation on G_{t_i} thanks to its relatively simple structure. However, it is not possible to generate variation on G_{t_i} if we assume that arbitrary amount is g due to the contract update rule in (**??**). Hence, we should develop an alternative sample path generating scheme which is discussed also in Section **??**. After considering above two topics, results are also applied to regression-later algorithm.

3.2 Bang-bang solution and Generating sample withdrawal

We denote A_{t_i} by available amount of withdrawal at time t_i. When the policyholder makes decision at time t_i, he encounters following six cases:

- If $X_t^- \leq g \leq G_t$, the original possible amount of withdrawal is $[0, g]$. In this case, we assume that

  $$A_{t_i} = \{0, X_t^-, g\}$$

- $X_t^- \leq G_t \leq g$, the original possible amount of withdrawal is $[0, G_{t_i}]$. In this case, we assume that

  $$A_{t_i} = \{0, X_t^-, G_t\}$$

- $G_t \leq X_t^- \leq g$, the original possible amount of withdrawal is $[0, X_t^-]$. In this case, we assume that,

  $$A_{t_i} = \{0, G_t, X_t^-\}$$

- $G_t \leq g \leq X_t^-$, the possible original amount is $[0, X_t^-]$. In this case, we assume that,

  $$A_{t_i} = \{0, G_t, X_t^-\}$$

- $g \leq G_t \leq X_t^-$, the original possible amount is $[0, X_t^-]$
  In this case, we consider two sub-cases :

  - if $g(T-t) < G_t$
    In this sub-case, the policyholder has the motivation to rebalance his guarantee account value since he may not be able to fully enjoy his minimum guarantee fearure if he only withdraws g in the future. So, we assume that the possible

    $$A_{t_i} = \left\{0, g, G_t, G - g(T-t-1), X_t^-\right\}$$

  - if $g(T-t) \geq G$

    $$A_{t_i} = \{0, g, G_t, X_t^-\}$$

- $g \leq X_t^- \leq G_t$, the original possible amount is $[0, X_t^-]$
  In this case, we have two sub-cases :

  - $g(T-t) < G_t$
    Similar to the previous case, the policyholder has the motivation to rebalance his gaurantee account. So, we assume that

    $$A_{t_i} = \left\{0, g, X_t^- - \frac{X_t^-}{G_t} g(T-t-1), X_t^-\right\}$$

- $g(T-t) \geq G_t$

$$A_{t_i} = \{0, g, G_t, X_t^-\}$$

After specifying A_{t_i}, we generate arbitrary w_{t_i} which gives variation on personal account and guarantee account simultaneously. m denotes number of elements of A_{t_i}. To do that, we rely on the following strategy:

- Put uniform point masses, $\mathbf{p} = (p_1, ..., p_{m-1})$ on assumed amount of withdrawals except for the maximum amount of withdrawal to rule out early surrender where $\sum_{h=1}^{m-1} p_h = 1$.
- Generate sample w_{t_i} from A_{t_i} whose distribution is given by $(\mathbf{p}, 0)$.

Like Monte Carlo simulation for American-style derivatives, the above random withdrawal scheme rules out lapse event by putting zero probability mass on possible maximum amount withdrawal.

3.3 Regression-later algorithm

In some cases, there is an alternative way to speed up convergence of LSMC. The regression-later algorithm approximates the inside part of the expectation in (**??**) by linear combination of basis functions. Then we compute the expectation of linear combination in the closed form and this expectation becomes an approximated value for the true expectation. That is,

$$e^{-\int_{t_i}^{t_{i+1}} r_s ds} \left\{ p_{x+t_i} V_{t_{i+1}} \left(X_{t_{i+1}}^-, G_{t_{i+1}}, r_{t_{i+1}}, \sigma_{t_{i+1}}, \mu_{t_{i+1}}(x) \right)) + q_{x+t_i} D_{t_{i+1}} \right\}$$

$$:= g\left(X_{t_{i+1}}^-, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{x+t_i} \right) = \sum_{j=1}^{\infty} \beta_j^{t_{i+1}} \times \varphi_j \left(X_{t_{i+1}}^-, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x) \right) \quad (21)$$

$$\approx \sum_{j=1}^{M} \beta_j^{t_{i+1}} \times \varphi_j \left(X_{t_{i+1}}^-, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x) \right) \quad \text{(First Approximation)} \quad (22)$$

$$\approx \sum_{j=1}^{M} \hat{\beta}_j^{t_{i+1}} \times \varphi_j \left(X_{t_i}^+, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x) \right) \quad \text{(Second Approximation)} \quad (23)$$

where $(\varphi_j)_{j=1}^M$ is set of basis functions, M is the degree of approximation, and $\beta_j^{t_{i+1}}$, $j = 1, 2, ..., M$ is the associated coefficient to the jth basis function. Even though the regression-later through (**??**) to (**??**) seems to be similar to the regression-now, it is fundamentally different form the regression-now. In the regression-later algorithm, the expectation is not approximated but the inside part of expectation is using $\mathcal{F}_{t_{i+1}}$ information. After estimating $\left(\beta_j^{t_{i+1}}\right)_{j=1}^M$, the true expectation is replaced by the expectation of (**??**). As mentioned in Section **??**, there are the pros and cons to the regression-later algorithm. The first benefit of regression-later is that the algorithm approximates the discounted benefit precisely since the value function, death benefit are highly correlated with later $\mathcal{F}_{t_{i+1}}$ information. However, we need to impose the following stronger conditions than the regression-now to compute (**??**). See **?** and **?**). In this paper, we impose some conditions on φ_j:

- Separability: the basis function $\varphi_j(x_1,x_2,x_3,x_4,x_5)$ is separable function, that is, it can be expressed as

$$\varphi_j(x_1,x_2,x_3,x_4,x_5) = \varphi_{j,1}(x_1,x_3,x_4,x_5)\varphi_{j,2}(x_2) \quad j=1,2,...,M$$

where x_2 is endogenous variable. Then,

$$\begin{aligned} &\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[\varphi_j\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right)|X^-_{t_i}, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right] \\ =&\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[\varphi_{j,1}\left(X^+_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right)|X^-_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right] \times \varphi_{j,2}(G_{t_{i+1}}) \end{aligned}$$

- Solvability : $\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[\varphi_{j,1}\left(X^+_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right)|X^-_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right]$ is known in the closed form of $v_j(X^+_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x))$.

Under above restrictions, we have

$$\begin{aligned} &\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[e^{-\int_{t_i}^{t_{i+1}} r_s ds}\left\{p_{x+t_i}V_{t_{i+1}}(X^-_{t_{i+1}}, G_{t_{i+1}}) + q_{x+t_i}D_{t_{i+1}}\right\}\middle| X^-_{t_i}, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right] \\ \approx\ &\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[\sum_{j=1}^{M}\hat{\beta}_j^{t_{i+1}} \times \varphi_j\left(X^+_{t_i}, G_{t_{i+1}}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right)\middle| X^-_{t_i}, G_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right] \\ \approx\ &\sum_{j=1}^{M}\hat{\beta}_j^{t_{i+1}} v_j(X^+_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)) \times \varphi_{j,2}(G_{t_{i+1}}) \end{aligned} \tag{24}$$

Then we solve the dynamic programming. More precisely,

- Initiate S_{t_0}, r_{t_0}, σ_{t_0}, $\mu_{t_0}(x)$, $X^+_{t_0}$, and G_{t_1}.
- For t_i, $i=1,2,...,n$
  1. Generate $S^k_{t_i}$, $r^k_{t_i}$, $\sigma^k_{t_i}$, and $\mu^k_{t_i}(x)$, $k=1,2,...,N$.
  2. Calculate $X^{-,k}_{t_i}$, $k=1,2,...,N$.
  3. Generate $w^k_{t_i}$, $k=1,2,...,N$
  4. Update $X^{+,k}_{t_i}$ and $G^k_{t_{i+1}}$, $k=1,2,...,N$.
  5. If $i \neq n$, go back to the **??**. Otherwise, go to the next step.
  6. Drop $w^k_{t_n}$, $k=1,2,...,N$
- Calculate $V_{t_n}(X^{-,k}_{t_n}, G^k_{t_n}) := \hat{V}_n^{k,later}(X^{-,k}_{t_n}, G^k_{t_n}, r^k_{t_n}, \sigma^k_{t_n}, \mu^k_{t_n}(x))$, $k=1,2,...,N$
- At t_i, $i=n-1,...,1$
  1. Regress $e^{-r^k_s(t_{i+1}-t_i)}\left\{e^{\mu^k_s(x)(t_{i+1}-t_i)}\hat{V}^{k,later}_{t_{i+1}}(X^{-,k}_{t_{i+1}}, G^k_{t_{i+1}}, r^k_{t_{i+1}}, \sigma^k_{t_{i+1}}, \mu^k_{t_{i+1}}(x)) + \left(1-e^{\mu^k_s(x)(t_{i+1}-t_i)}\right)D^k_{t_{i+1}}\right\}$ on $\left(\varphi_j\left(X^{-,k}_{t_{i+1}}, G^k_{t_{i+1}}, r^k_{t_i}, \sigma^k_{t_i}, \mu^k_{t_i}(x)\right)\right)_{j=1,...,M,\ k=1,...,N}$ and obtain $\hat{\beta}_j^{t_{i+1}}$, $j=1,2,...,M$.
  2. Compute

$$\mathbb{E}^{\mathbb{P}\times\mathbb{Q}}\left[\varphi_{j,1}\left(X^+_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right)|X^-_{t_i}, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)\right]$$

3. Solve

$$\hat{V}_{t_i}^{k,later}(X_{t_i}^{-,k}, G_{t_i}^k, r_{t_i}^k, \sigma_{t_i}^k, \mu_{t_i}^k(x)) = \max_{w_{t_i}} w_{t_i} - \text{fee}_{t_i}^I - \text{fee}_{t_i}^R$$
$$+ \sum_{j=1}^{M} \hat{\beta}_j^{t_{i+1}} v_j(X_{t_i}^+, r_{t_i}, \sigma_{t_i}, \mu_{t_i}(x)) \times \varphi_{j,2}(G_{t_{i+1}}) \tag{25}$$

subject to

$$w_{t_i} \in \mathcal{A}_{t_i}$$

for $k = 1, 2, ..., N$.

4. If $i \neq 1$, go back to ??. Otherwise, go to the next step.

- The price of GMWB at $t = 0$

$$\hat{V}_{later}(0) = \frac{1}{N}\sum_{k=1}^{N} e^{-r_{t_0}^k(t_1-t_0)}\left\{e^{\mu_{t_0}^k(x)(t_1-t_0)}\hat{V}_{t_1}^{k,later}(X_{t_1}^{-,k}, G_{t_1}^k) + \left(1 - e^{-\mu_{t_0}^k(x)(t_1-t_0)}\right) D_{t_1}^k\right\}$$

In ?, authors show why regression-later converges faster than regression-now. And we provide another advantage of regression-later over regression-now in Section ?? when we consider stochastic interest rate, stochastic volatility and stochastic mortality.

4 Applications

In this section, we implement two LSMC algorithms to price GMWB. First, we follow the usual Black-Scholes framework with deterministic mortality. Under these setting, the dimension of the dynamic programming is two, and so we can find optimal withdrawal and price of GMWB using an usual grid method. Therefore, we can check if the LSMC works properly for solving dynamic programming and finding a price. Then we compare regression-now algorithm and regression-later algorithm.

And then we consider a model including stochastic mortality, stochastic interest rate and stochastic volatility where the grid method is computationally expensive in Section ??. In particular, we discuss the closed form of special basis function for regression-later.

4.1 GMWB under Black-Scholes framework

Under the Black-Scholes framework with deterministic mortality, the dynamic programming at t_i becomes

$$V_{t_i}(X_{t_i}^-, G_{t_i}) = \max_{w_{t_i}} w_{t_i} - \text{fee}_{t_i}^I - \text{fee}_{t_i}^R$$
$$+ e^{-r(t_{i+1}-t_i)}\mathbb{E}^{\mathbb{Q}}\left[{}_{t_{i+1}-t_i}p_{x+t_i} V_{t_{i+1}}(X_{t_{i+1}}^-, G_{t_{i+1}}) + {}_{t_{i+1}-t_i}q_{x+ti} D_{t_{i+1}} \middle| X_{t_i}^-, G_{t_i}\right],$$
$$\text{subject to} \quad w_{t_i} \in A_{t_i}$$
$$V_{t_N}(X_{t_N}^-, G_{t_N}) = \max(X_{t_N}^-, \min(g, G_{t_N})) \tag{26}$$

First, we solve (**??**) via grid method. Under the Black-Scholes framework, the expectation in (**??**) is given by

$$\mathbb{E}^{\mathbb{Q}}\left[{}_{t_{i+1}-t_i}p_{x+t_i}V_{t_{i+1}}(X^-_{t_{i+1}}, G_{t_{i+1}}) + {}_{t_{i+1}-t_i}q_{x+t_i}D_{t_{i+1}}\middle| X^-_{t_i}, G_{t_i}\right]$$
$$= {}_{t_{i+1}-t_i}p_{x+t_i}\int_0^\infty V_{t_{i+1}}(X^+_{t_i}e^x, G_{t_{i+1}})f(x)dx + {}_{t_{i+1}-t_i}q_{x+t_i}\int_0^\infty \max(X^+_{t_i}e^x, G_{t_{i+1}})f(x)dx$$

where

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2(t_{i+1}-t_i)}}e^{-\frac{(x-(r-\phi-\frac{1}{2}\sigma^2(t_{i+1}-t_i))}{2\sigma^2(t_{i+1}-t_i)}}$$

Then this integral can be evaluated using usual numerical integration method with linear interpolation. We assume that the mortality law is given by exponential law, that is, the constant force of mortality. Under this assumption, ${}_tp_x = e^{-\mu t}$ where $\mu > 0$. For LSMC, we consider M basis functions of simple polynomials to approximate the expectation. Under the regression-now, the second form of approximation for the expectation becomes

$$\mathbb{E}^{\mathbb{Q}}\left[{}_{t_{i+1}-t_i}p_{x+t_i}V_{t_{i+1}}(X^-_{t_{i+1}}, G_{t_{i+1}}) + {}_{t_{i+1}-t_i}q_{x+t_i}X^-_{t_{i+1}}\middle| X^+_{t_i}, G_{t_{i+1}}\right]$$
$$\approx \sum_{j=1}^{M}\hat{\alpha}^{t_i}_j\left(X^+_{t_i}\right)^{j_1}\left(G_{t_{i+1}}\right)^{j_2}$$

where $j_1 + j_2 = j$ and $j_1,\ j_2 \in \mathbb{N}$.

Under the regression-later algorithm with simple polynomials functions, we have the second form of approximation by

$$\begin{aligned}&{}_{t_{i+1}-t_i}p_{x+t_i}V_{t_{i+1}}(X^-_{t_{i+1}}, G_{t_{i+1}}) + {}_{t_{i+1}-t_i}q_{x+t_i}D_{t_{i+1}}\\&\approx \sum_{k=1}^{M}\hat{\beta}^{t_{i+1}}_k\left(X^-_{t_{i+1}}\right)^{k_1}\left(G_{t_{i+1}}\right)^{k_2}\end{aligned} \tag{27}$$

where $k_1 + k_2 = k$ and $k_1,\ k_2 \in \mathbb{N}$. Then the expectation of (**??**) is

$$\begin{aligned}&\mathbb{E}^{\mathbb{Q}}\left[\sum_{k=1}^{M}\hat{\beta}^{t_{i+1}}_k\left(X^-_{t_{i+1}}\right)^{k_1}\left(G_{t_{i+1}}\right)^{k_2}\right]\\&= \sum_{k=1}^{M}\hat{\beta}^{t_{i+1}}_k\mathbb{E}^{\mathbb{Q}}\left[(X^-_{t_{i+1}})^{k_1}(G_{t_{i+1}})^{k_2}\right]\\&= \sum_{k=1}^{M}\hat{\beta}^{t_{i+1}}_k(X^+_{t_i})^{k_1}e^{(r-\phi-\frac{1}{2}\sigma^2)(t_{i+1}-t_i)k_1+\frac{1}{2}\sigma^2(t_{i+1}-t_i){k_1}^2}(G_{t_{i+1}})^{k_2}\end{aligned} \tag{28}$$

To improve efficiency of algorithm, we only consider samples with $X^+_{t_i} > 0$ or $G_{t_{i+1}} > 0$ for regression-now algorithm and $X^-_{t_{i+1}} > 0$ or $G_{t_{i+1}} > 0$ for regression-later algorithm when estimating coefficients.

GMWB contract	
Maturity	15
Number of withdrawal per year	1
Initial Premium (P_0)	15
Option fee(ϕ)	0.25%
ep_{t_i}	$(8\%, 7\%, 6\%, 5\%, 4\%, 3\%, 2\%, 1\%, 0\%, \cdots, 0\%)$
pg_{t_i}	10%
g	1
Policyholder	
Age	55
μ	0.01
Financial Market	
Risk free rate	5%
Volatility	18%

Table 1: Description of GMWB contract under Black-Scholes assumption

The description of GMWB contract, financial market and policyholder is given in Table **??**. We first implement LSMC algorithm with second degree and compare results from regression-now and regression-later to results from grid method.

In the grid method, the maximum value of $X_{t_i}^-$ is thirty and the space for the personal account is desrcretized by sixty points. For G_{t_i}, the maximum value is P_0 and the space for the guarantee account is desrcretized by thirty points. The estimated price from grid method is 15.06. We take it as a true price even though the original option fee 0.25% should be reduced to achieve the equivalence principle. As we see in Figure **??**, the estimated price is closed to the true price even though we use simple polynomials with degree two. So we expect the estimated price also converge to the true price if polynomials with three or four degrees are used. In particular, we see that the speed of convergence from regression-later is faster than regression-now and the estimates are stable. High degree of polynomial does not play much role to improve accuracy of estimator by analyzing R^2 in regression-later algorithm. So, it seems to be enough to set highest degree of polynomial two in regression-later algorithm.

We analyze optimal amount of withdrawal with 700,000 simulation. In Figure **??**, we provide the optimal policy at $t_i = 14, 13, 12, 11$. We see that two LSMC algorithms provide good estimated optimal policy at $t_i = 14$. As moving backward, the estimated optimal policies deviate true optimal policies. The more moving backward, the more error are generated from estimated value functions. However, from all estimated optimal policies in Figure **??**, the regression-later performs better than the regression-now.

Now we increase the number of basis functions. But increasing number of basis functions deserves attention. In this example, we use simple polynomials as basis functions. One of disadvantages of simple polynomial function is that high-degree polynomials oscillate much between exact-fit values. This property of high-degree polynomials produce large error of estimates for value function by generating over-fitting problem. For instance, the estimated price of regression-now shows poor convergence as the number of simulation is increased. We verify this unfavorable property of simple polynomial in Table **??**. Therefore, we should use another orthonormal basis functions to avoid this problem. In the next application, we

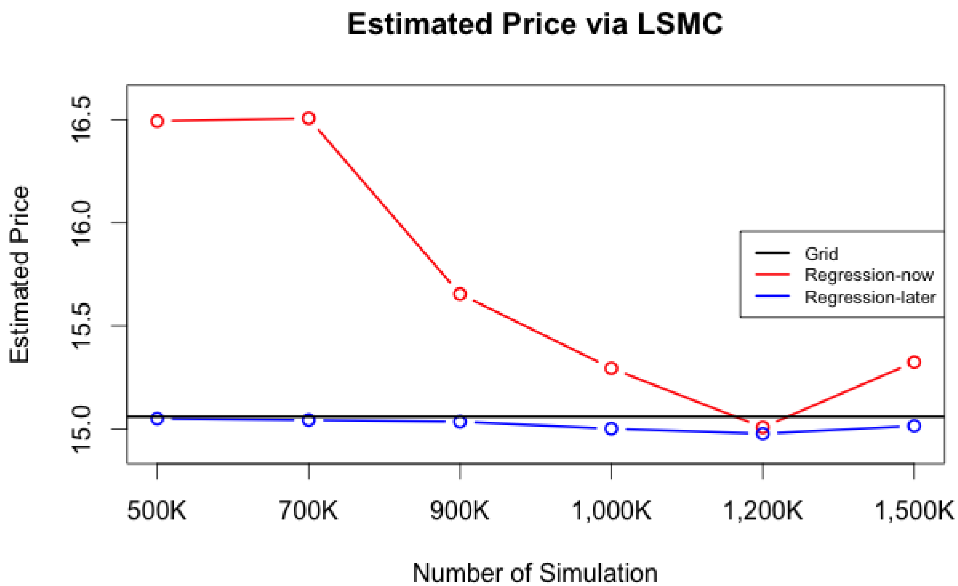

Figure 1: Convergence of LSMC algorithm

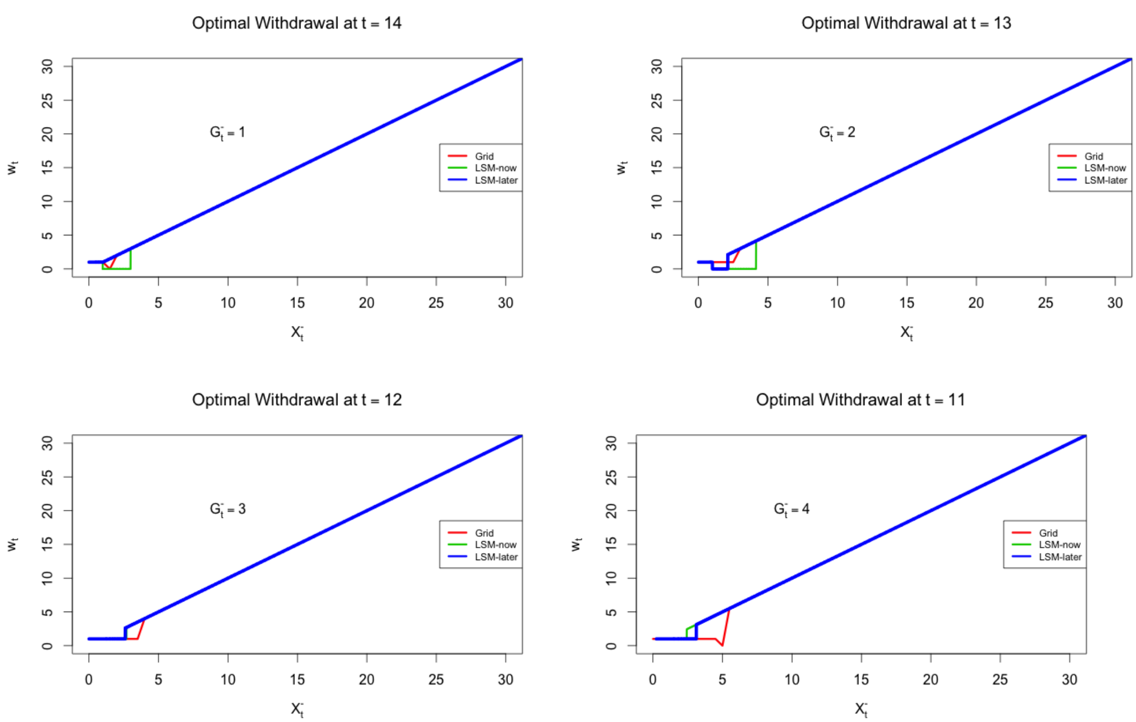

Figure 2: Comparison between LSMC algorithms and grid method

# of Simulation	Regression-now	Regression-later
500,000	19.3290	15.2815
700,000	18.5032	15.2018
900,000	17.0200	14.9841
1,000,000	15.8018	15.0707
1,200,000	16.5261	15.0410
2,000,000	16.6768	15.0755

Table 2: Convergence of LSMC with four degree

apply orthonormal polynomials such as Hermite polynomials or Chebyshev polynomials. Or we can find optimal basis functions for specific problem. See **?**. As we see in Table **??**, the regression-later algorithm converges to the true price fast. On the other hand, polynomial of four degree for regression-now does not look good choice to approximate value functions. Also, the convergence speed of regression-later is faster than regression-now.

4.2 stochastic mortality, stochastic interest rate and stochastic volatility

5 Conclusion

We provide two LSMC algorithms to price GMWB contracts. Two methods performs well in general. As shown in previous sections, the regression-later estimators converge faster than the regression-now estimators. This is because the regression-later algorithm directly regresses $\mathcal{F}_{t_{i+1}}$-measurable function on $\mathcal{F}_{t_{i+1}}$-measurable information. On the other hand, the regression-now algorithm uses information at t_i to approximate $\mathcal{F}_{t_{i+1}}$-measurable function. The error associated to regression-now algorithm can be serious if $t_{i+1} - t_i$ is large. However, the regression-now algorithm is more applicable than the regression-later algorithm since it is not model dependent. On the other hand, it is essential to have closed forms of basis functions, which is model dependent if one wants to apply regression-later algorithm.

In this paper, we assume that simple polynomials and classical orthogonal polynomials are regarded as basis functions. But we do not address which polynomials are best basis functions. For further study, it is important how to find the best basis functions and their expectations for regression-later algorithm.

References

Amemiya, T., 1985. Advanced econometrics. Harvard university press, Cambridge.

Andrew, J.G., David, B., Kevin, D., 2008. Modeling and management of mortality risk: a review. Scandinavian Actuarial Journal, 2009.

Azimzadeh, P., Forsyth, P., 2008. The existence of optimal bang-bang controls of gmxb contracts. Working paper, 2014.

Bauer, D., Ha, H., 2013. A Least-Squares Monte Carlo Approach to the Calculation of Capital requirements. Working paper.

Bauer, D., Bergmann, D., Kiesel, R., 2010a. On the Risk-Neutral Valuation of Life Insurance Contracts with Numerical Methods in View. ASTIN Bulletin, 40: 31-61.

Bauer, D., Kling, A., Russ, A., 2008. A Universal Pricing Framework for Guaranteed Minimum Benefits in Variable Annuities. ASTIN Bulletin, 38: 621-651.

Bauer, D., Bergmann, D., Reuss, A., 2009. Solvency II and Nested Simulations - a Least-Squares Monte Carlo Approach. Working paper, Georgia State University and Ulm University.

Bacinello, A.R., Milossovich, P., Montealegre A., 2013. A Dynamic Programming Algorithm for the Valuation of Guaranteed Minimum Withdrawal Benefits in Variable Annuities. Research Paper Series, N. 4, 2013, Bruno de Finetti.

Bacinello, A.R., Milossovich, P., Olivieri, A., Pitacco, E., 2011. Variable Annuities: A Unifying Approach. Available at SSRN: http://ssrn.com/abstract=1810522 or http://dx.doi.org/10.2139/ssrn.1810522.

Beutner, E., Pelsser, A., Schweizer, J., 2013. Fast Convergence of Regress-Later Estimates in Least Squares Monte Carlo. Available at SSRN: http://ssrn.com/abstract=2328709 or http://dx.doi.org/10.2139/ssrn.2328709.

Björk, T., 2004. Arbitrage Theory in Continuous Time, Second Edition. Oxford University Press, New York.

Glassermann, P., 2004. Monte Carlo Methods in Financial Engineering. Springer, New York.

Glassermann, P., Yu, B., 2002. Simulation for American Options: Regression now or Regression later?. Monte Carlo and Quasi-Monte Carlo Methods.

Huang Y.T., Kwok, Y.K., 2014. Regression-based Monte Carlo methods for stochastic control models: Variable annuities with lifelong guarantees. Submitted to Mathematical Finance.

Judd, K., 1998. Numerical Methods in Economics. The MIT press, Cambridge.

Longstaff, F.A., Schwartz, E.S., 2001. Valuing American Options by Simulation: A Simple Least-Squares Approach. Review of Financial Studies, 14: 113-147.

Yan L., 2010. Guaranteed Minimum Withdrawal Benefits in Variable Annuities. Ph.D. Thesis, University of Waterloo, 14: 113-147.

Moenig T., Bauer, D., 2011. Policyholder Exercise Behavior for Variable Annuities including Guaranteed Minimum Withdrawal Benefits. Working Paper.

Nadarajah S., Margot, F., Secomandi, N., 2012. Valuation of Multiple Exercise Options with Energy Applications. Tepper Working Paper.

Zhu, N., Bauer, D., 2013. Modeling and Forecasting Mortality Projections. Working Paper.