



North American Actuarial Journal

ISSN: 1092-0277 (Print) 2325-0453 (Online) Journal homepage: http://www.tandfonline.com/loi/uaaj20

Efficient Greek Calculation of Variable Annuity Portfolios for Dynamic Hedging: A Two-Level **Metamodeling Approach**

Guojun Gan & X. Sheldon Lin

To cite this article: Guojun Gan & X. Sheldon Lin (2017) Efficient Greek Calculation of Variable Annuity Portfolios for Dynamic Hedging: A Two-Level Metamodeling Approach, North American Actuarial Journal, 21:2, 161-177, DOI: 10.1080/10920277.2016.1245623

To link to this article: http://dx.doi.org/10.1080/10920277.2016.1245623



Published online: 06 Jan 2017.



🖉 Submit your article to this journal 🗹





View related articles 🗹



🌔 🛛 View Crossmark data 🗹

Full Terms & Conditions of access and use can be found at http://www.tandfonline.com/action/journalInformation?journalCode=uaaj20



Efficient Greek Calculation of Variable Annuity Portfolios for Dynamic Hedging: A Two-Level Metamodeling Approach

Guojun Gan¹ and X. Sheldon Lin²

¹Department of Mathematics, University of Connecticut, Storrs, Connecticut ²Department of Statistical Sciences, University of Toronto, Toronto, Canada

The financial risk associated with the guarantees embedded in variable annuities cannot be addressed adequately by traditional actuarial techniques. Dynamical hedging is used in practice to mitigate the financial risk arising from variable annuities. However, a major challenge of dynamical hedging is to calculate the dollar Deltas of a portfolio of variable annuities within a short time interval so that rebalancing can be done on a timely basis. In this article, we propose a two-level metamodeling approach to efficiently estimating the partial dollar Deltas of a portfolio of variable annuities under a multiasset framework. The first-level metamodel is used to estimate the partial dollar Deltas at some well-chosen market levels, and the second-level metamodel is used to estimate the partial dollar Deltas at the current market level based on the precalculated partial dollar Deltas. Our numerical results show that the proposed approach performs well in terms of accuracy and speed.

1. INTRODUCTION

A variable annuity (VA) is an annuity policy issued by an insurance company, under which the policyholder agrees to make a lump-sum purchase payment or a series of purchase payments to the company and in return the company makes benefit payments to the policyholder, beginning either immediately or on a future date. Purchasing a VA is similar to investing in a portfolio of mutual funds because the policyholder may allocate his or her money to one or more investment funds provided by the issuer. However, a VA differs from mutual funds in the way that a VA provides certain benefit guarantees or riders. Common types of guarantees include guaranteed minimum death benefits (GMDBs) and guaranteed minimum living benefits (GMLBs). There are several types of GMLBs: guaranteed minimum accumulation benefits (GMABs), guaranteed minimum maturity benefits (GMMBs), guaranteed minimum income benefits (GMIBs), guaranteed minimum withdrawal benefits (GMWBs), and guaranteed life withdrawal benefits (GLWBs). For details about these guarantees, readers are referred to Milevsky and Posner (2001), Gerber and Shiu (2003), Bauer et al. (2008), Lin et al. (2009), Bélanger et al. (2009), Ng and Li (2013), Bacinello et al. (2014), and Huang et al. (2014).

Variable annuities have grown rapidly in popularity in the past decade. According to LIMRA, the sales of VAs in the United State in 2013 and 2014 were \$145 and \$140 billion, respectively.¹ As a result, almost every insurance company that has a VA business is managing a large VA portfolio. The exposure to the aforementioned guarantees embedded in VAs is posing a significant financial risk to these insurers, and the risk management is a critical issue to these VA providers.

Unlike the mortality risk associated with the traditional annuity policies, the financial risk associated with the guarantees embedded in VAs cannot be derisked by underwriting as many policies as possible. To hedge the financial risk, insurers use a dynamical hedging approach in which a hedge portfolio consisting of highly liquid equity index futures (such as S&P500 futures) is constructed and dynamically rebalanced such that the change of the hedge portfolio offsets the change of the guarantee value of the VA portfolio. For such a hedge program to be successful, one needs to first map each of the investment funds in the pool to a common set of tradable equity indices by analyzing the historical returns of the investment funds and the tradable indices, because the investment funds are normally not tradable. Examples of fund mapping are given in Table 2. One then needs to accurately and efficiently calculate the partial dollar Deltas of the VA portfolio, the sensitivities of the guarantee value of the VA portfolio with respect to tradable equity indices, at rebalancing times to determine the position of the futures of each tradable equity index in the hedge portfolio.

Address correspondence to X. Sheldon Lin, Department of Statistical Sciences, University of Toronto, Toronto, ON M5S 3G3, Canada. E-mail: sheldon@utstat.utoronto.ca

¹http://www.limra.com/uploadedFiles/limra.com/LIMRA_Root/Posts/PR/Data_Bank/_PDF/2005-2014-Annuities-Estimates-Alternative.pdf

G. GAN AND X. S. LIN

The calculation of partial dollar Deltas is challenging. The payoff function of the guarantees is often path-dependent so no closed-form formulas are available for calculating the fair market value of the guarantees. A VA portfolio is highly nonhomogeneous because each policy in the portfolio is different in terms of age, gender, time to maturity, guarantee type, etc. In practice, insurers rely heavily on Monte Carlo simulation to calculate the fair market value and the partial dollar Deltas of a VA portfolio by simulation. In a dynamical hedging program with intraday rebalancing, however, the calculation must not only be accurate but also be very fast to effectively hedge a VA portfolio. For example, an insurer plans to rebalance the hedge portfolio at 1:00 pm: the partial dollar Deltas must be calculated based on the levels of the equity indices at 1:00 pm, and the calculation must be completed within a very short time interval (e.g., a few minutes or less) so that the insurer can rebalance the hedge portfolio under the current market condition. Otherwise, the calculated partial dollar Deltas based on the market condition when the calculation started may be very different from the partial dollar Deltas of the VA portfolio at the time the calculation is completed. This is especially true when the market moves fast. For a large VA portfolio it is extremely challenging to calculate the partial dollar Deltas accurately and in a timely manner through complete simulation. See Gan and Lin (2015) in which we discuss computational time when performing a complete simulation on a large VA portfolio.

In this article, we propose a two-level metamodeling approach to deal with the aforementioned computational issue arising from dynamically hedging large VA portfolios. The idea is to precalculate the partial dollar Deltas of a VA portfolio at a number of well-chosen market levels at a time before the rebalancing time (e.g., at the market closing time of the previous day). We then estimate the partial dollar Deltas at the current market level at the rebalancing time using the precalculated partial dollar Deltas. The approach involves two metamodels: the first metamodel, called the level-one metamodel, is used to estimate the partial dollar Deltas of the portfolio at any market level based on the precalculated dollar Deltas obtained from the level-one metamodel. The changes of dollar Deltas of a VA portfolio within a short period of time (e.g., one night) are mainly drived by market changes. The changes in actuarial assumptions (e.g., mortality, lapse) can be ignored. As a result, it is reasonable to estimate the dollar Deltas of the portfolio based on market levels.

The rest of the article is organized as follows. In Section 2 we give a brief review of simulation metamodeling. In Section 3 two Latin hypercube sampling methods are introduced to select a number of well-chosen market levels of the tradable indices and representative VA policies. In Section 4 we present two kriging methods and the two-level metamodeling approach for partial dollar Delta calculation. In Section 5 we demonstrate the performance of the proposed approach using a portfolio of 100,000 synthetic VA policies. We conclude the article with some remarks in Section 6.

2. SIMULATION METAMODELING

In simulation metamodeling, a metamodel refers to a model of a simulation model as described in Friedman (1996). In many practical situations, an underlying simulation model in use is very complex and too computationally intensive. A metamodel of the simulation model relates the inputs to the outputs of the simulation model and is simple and computationally efficient. A metamodel is constructed by running a small number of expensive simulations and used in place of the simulation model for further analysis.

Since Kleijnen (1975) introduced the concept of metamodels for simulation models, many papers on metamodeling and its applications have been published. Kleijnen (2009) presented a review of the kriging metamodel. Among other applications, Ankenman et al. (2010) extended the basic theory of kriging in a stochastic simulation setting. There are also several books on metamodels. Friedman (1996) provides comprehensive coverage of simulation metamodeling and discusses methodology, usage, and applications of metamodels. Box and Draper (2007) cover many topics on response surface methodology, which is related to metamodeling. Das (2014) is devoted to robust response surface methodology and contains a review of the existing literature on response surface methodology.

Recently, the idea of metamodeling has been applied to address the computational problems related to the valuation and risk management of variable annuities. Gan (2013) proposed a clustering method (Gan 2011) and an ordinary kriging method (Isaaks and Srivastava 1990) to estimate the Greeks of a large VA portfolio. In Gan and Lin (2015), a clustering method and a universal kriging for the functional data method (Caballero et al. 2013) are proposed to estimate the partial dollar Deltas of a portfolio of VA policies under a nested simulation or stochastic-on-stochastic framework (Reynolds and Man 2008).

Building a metamodel of a simulation model involves two steps: first, one needs to select a small set of sample points from the input space of the simulation model by using some experimental design methods (Alam et al. 2004); second, a metamodel is built based on the input-output relationships generated by running the simulation model on the selected sample points. For example, Latin hypercube sampling is a popular experimental design method (Viana 2013) that can be used to select sample points in the first step. In the second step, we can choose a metamodel form and use the input-output information of the simulation model to estimate the parameters of the metamodel. There are several metamodel forms such as spline models, radial basis functions, kernel methods, and spatial correlation models (Barton 1994). In particular, kriging is a type of spatial correlation models. It has been

reported that Latin hypercube sampling works well for kriging (Viana 2013). In this article, we shall use Latin hypercube sampling and kriging as our experimental design method and metamodel form, respectively.

3. LATIN HYPERCUBE SAMPLING

To accurately and efficiently estimate the partial dollar Deltas of a VA portfolio, it is crucial to have a small number of wellchosen market levels of the tradable indices and representative VA policies (e.g., Gan and Lin 2015). In this section, we introduce two Latin hypercube sampling methods: an unconditional Latin hypercube sampling method and a conditional Latin hypercube sampling method. The first method is used for selecting market levels, and the second method is used for selecting representative VA policies. The difference between these two methods is that the former involves selecting sample points (i.e., possible market returns of tradable equity indices) from a space, while the later involves selecting sample points (i.e., the representative VA policies in the portfolio) from a large set of points.

3.1. Unconditional Latin Hypercube Sampling

Let *H* be the number of tradable equity indices of the hedge portfolio to which the investment funds are mapped. We use an unconditional Latin hypercube sampling method (Viana, 2013) to select *m* market levels on these *H* indices. Here, a market level **r** is a vector of *H* returns $(r_1, r_2, ..., r_H)$ with r_h being the intraday return (i.e., the return calculated from the current price and the close of the last business day) of the *h*th index. Regarding the number of sample points, we may choose m = 10H in practice (Loeppky et al. 2009).

A Latin hypercube design with *m* sample points in a *H*-dimensional space is created as follows. We first decide the ranges of movements of the indices. For example, we can use $R_h = [-\sigma_h, \sigma_h]$ as the range of the *h*th index for h = 1, 2, ..., H, where σ_h is the annualized volatility of the *h*th index. This range is wide enough to capture most of the daily movements of the index because the daily volatility is approximately $\sigma_h/\sqrt{252}$, which is much smaller than σ_h .

Second, for each h = 1, 2, ..., H, we divide the range R_h into m - 1 intervals of equal length. This generates m^H dividing points, including the points in corners. Then we obtain a Lain hypercube design by choosing m points from these dividing points such that no two points have the same coordinates. There are many such Latin hypercube designs (Liefvendahl and Stocki 2006). To obtain an optimal Latin hypercube design, we may use the maximin criterion (Carnell 2012; Moon et al. 2011), that is, choosing m points $\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_m$ from the dividing points such that the minimum distance

$$M = \min\{\|\mathbf{r}_i - \mathbf{r}_j\| : 1 \le i < j \le m\}$$

$$\tag{1}$$

is maximized, where $\|\cdot\|$ is the usual L^2 norm or the Euclidean distance.

3.2. Conditional Latin Hypercube Sampling

The unconditional Latin hypercube sampling method presented in the previous subsection provides a full coverage of the range of each variable. A conditional Latin hypercube sampling method is a little bit different in that it selects a subsample of size k from a set of n points with d ancillary variables $V = (v_1, v_2, \ldots, v_d)$ in such a way that the subsample forms a Latin hypercube (Minasny and McBratney 2006; Roudier 2011). Minasny and McBratney (2006) proposed a search algorithm based on heuristics rules to find conditional Latin hypercubes. The algorithm is also available in the R package clhs (Roudier 2011). In this subsection we apply the conditional Latin hypercube sampling method to select a set of representative policies from a VA portfolio.

Let $X = {\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n}$ be a portfolio of *n* VA policies. Suppose that each policy is characterized by *d* attributes/variables *V* such as gender, age, guarantee type, on time to maturity. The search algorithm used to select *k* representative VA policies is described as follows. First, the quantile distribution of each continuous variable in *V* is divided into *k* strata. Let the quantile distribution of a continuous variable be q_j^i for $j \in I_1$ and i = 1, 2, ..., k + 1, where I_1 is the index set of continuous variables in *V*. Second, *k* policies $Z = {\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_k}$ are selected from *X*, and the following objective function is calculated:

$$O = w_1 O_1 + w_2 O_2 + w_3 O_3.$$

In the above, w_1 , w_2 , and w_3 are the weights of the three components O_1 , O_2 , and O_3 , respectively. The first component, O_1 , is the objective function for continuous variables (e.g., age, time to maturity):

$$O_1 = \sum_{j \in I_1} \sum_{i=1}^n |\eta_j^i - 1|,$$

where I_1 is the index set of continuous variables and η_j^i is the number of z_{ij} , the value of the *j*th variable of \mathbf{z}_i , that fall in between quantiles q_i^i and q_j^{i+1} . The second component, O_2 , is the objective function for categorical variables (e.g., gender, guarantee type):

$$O_2 = \sum_{j \in I_2} \sum_{i=1}^{c_j} \left| \frac{\eta_j^i}{k} - \kappa_j^i \right|,$$

where I_2 is the index set of categorical variables and c_j is the number of categories of the *j*th variable, η_j^i is the number of z_{ij} that belongs to the *i*th category of the *j*th variable, and κ_j^i is the proportion of X in the *i*th category of the *j*th variable. The third component, O_3 , is used to ensure the correlation of the sampled continuous variables will replicate the original data and is defined as

$$O_3 = \sum_{j=1}^{|I_1|} \sum_{l=1}^{|I_1|} |c_{jl} - t_{jl}|.$$

where c_{jl} and t_{jl} are the elements of the sample correlation matrix of the continuous data of X and the sample correlation matrix of the continuous data of Z, respectively. An annealing schedule (i.e., the simulated annealing algorithm) is performed to obtain the final sample. As a result, the final sample obtained from the search algorithm preserves the distribution and multivariate correlation of the original data.

4. THE TWO-LEVEL METAMODELING APPROACH FOR PARTIAL DOLLAR DELTA CALCULATION

In this section we introduce the two-level metamodeling approach for calculating partial dollar Deltas used in dynamical hedging. This approach consists of two metamodels: the level-one metamodel is used to estimate the partial dollar Deltas of the portfolio at the prechosen market levels; the level-two metamodel is used to estimate the partial dollar Deltas of the portfolio at an arbitrary market level based on the precalculated dollar Deltas at the prechosen market levels. Figure 1 shows the hierarchical structure of the two-level metamodeling approach.

The level-one metamodel can be used to estimate the dollar Deltas of the VA portfolio at any market level. However, the levelone metamodel is still slow because it involves running the Monte Carlo simulation model on a set of representative VA contracts. This is the reason why we build the level-two metamodel on top of the level-one method model.



FIGURE 1. Two-Level Metamodeling Approach to Estimate the Partial Dollar Deltas at Arbitrary Market Level **r**. *Note:* The symbols $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_m$ denote the prechosen market levels. The symbols $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k$ denote the representative VA contracts.

4.1. The Level-One Metamodel

The level-one metamodel is used to estimate the dollar Deltas of a VA portfolio from the dollar Deltas of a set of representative VA contracts $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k$, which are selected by the so-called conditional Latin hypercube sampling method. The reason why we use the conditional Latin hypercube sampling method to select representative VA contracts is that we know all the VA contracts in the portfolio. The conditional Latin hypercube sampling method can be used to select a subset of VA contracts from the portfolio.

The dollar Deltas of a VA contract are usually proportional to some variables of the VA contract. For example, if a VA contract has large account values or large guarantee values, we expect the dollar Deltas of the VA contract are also large. Since the universal kriging method contains a regression part that can be used to capture such trend, we choose the universal kriging method to build the level-one metamodel.

Recall that $X = {\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n}$ represents a portfolio of *n* VA policies and $\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_k$ are *k* representative VA policies selected from *X* by the conditional Latin hypercube sampling method. Further, $\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_m$ are *m* market levels selected by the unconditional Latin hypercube method (see Section 3). Let $f(\mathbf{z}_i, \mathbf{r}_l, h)$ be the partial dollar Delta of the representative policy \mathbf{z}_i with respect to *h*th tradable index at market level \mathbf{r}_l , for i = 1, 2, ..., k, and l = 1, 2, ..., m. These quantities are obtained through Monte Carlo simulation. We now use a universal kriging model to estimate the dollar Deltas of an arbitrary policy from those of the *k* representative policies at the selected market levels $\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_m$. One of the advantages of the use of a universal kriging model is that it can capture the trends in the VA portfolio.

We assume that the partial dollar Delta of a VA policy \mathbf{x}_i on the *h*th tradable index when the market level on the next day is \mathbf{r}_l to be (Cressie 1993)

$$f(\mathbf{x}_i, \mathbf{r}_l, h) = \sum_{j=0}^d b_j(\mathbf{x}_i)\beta_j + \delta(\mathbf{x}_i, \mathbf{r}_l, h),$$
(2)

where β_0, \ldots, β_d are unknown parameters, $\delta(\cdot, \cdot, \cdot)$ is a zero-mean intrinsically stationary spatial random process, and $b_0(\cdot), \ldots, b_d(\cdot)$ are known functions of \mathbf{x}_i . In our implementation, we convert all categorical variables to binary dummy variables and choose $b_0(\mathbf{x}) = 1$ and $b_j(\mathbf{x}_i)$ to be the *j*th component of \mathbf{x}_i for $j = 1, 2, \ldots, d$. Under the universal kriging model (2), $f(\mathbf{x}_i, \mathbf{r}_i, h)$ can be predicted as (Cressie 1993)

$$\hat{f}(\mathbf{x}_i, \mathbf{r}_l, h) = \boldsymbol{\lambda}_i^T f(Z, \mathbf{r}_l, h) = \sum_{p=1}^k \lambda_{ip} f(\mathbf{z}_p, \mathbf{r}_l, h),$$
(3)

where $f(Z, \mathbf{r}_l, h)$ denotes a k-dimensional column vector given by

$$f(Z, \mathbf{r}_l, h) = (f(\mathbf{z}_1, \mathbf{r}_l, h), f(\mathbf{z}_2, \mathbf{r}_l, h), \dots, f(\mathbf{z}_k, \mathbf{r}_l, h))^T$$

and $\lambda_i = (\lambda_{i1}, \lambda_{i2}, \dots, \lambda_{ik})^T$ is a column vector of kriging weights. The kriging weights are determined by the following linear equation system:

$$\begin{pmatrix} \mathbf{A}(Z) & \mathbf{B}(Z) \\ \mathbf{B}(Z)^T & \mathbf{0} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\lambda}_i \\ \mathbf{v}_i \end{pmatrix} = \begin{pmatrix} \mathbf{A}(Z, \mathbf{x}_i) \\ \mathbf{B}(\mathbf{x}_i)^T \end{pmatrix}.$$
(4)

In Equation (4), A(Z) is a $k \times k$ matrix defined as

$$\mathbf{A}(Z) = \begin{pmatrix} \gamma(\mathbf{z}_1, \mathbf{z}_1) & \gamma(\mathbf{z}_1, \mathbf{z}_2) & \cdots & \gamma(\mathbf{z}_1, \mathbf{z}_k) \\ \gamma(\mathbf{z}_2, \mathbf{z}_1) & \gamma(\mathbf{z}_2, \mathbf{z}_2) & \cdots & \gamma(\mathbf{z}_2, \mathbf{z}_k) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(\mathbf{z}_k, \mathbf{z}_1) & \gamma(\mathbf{z}_k, \mathbf{z}_2) & \cdots & \gamma(\mathbf{z}_k, \mathbf{z}_k) \end{pmatrix},$$
(5)

where $\gamma(\cdot, \cdot)$ is a semivariogram function. Among several semivariogram functions, we found that the exponential semivariogram function works well in practice. It is defined as

$$\gamma(\mathbf{x}, \mathbf{y}) = 1 - \exp\left(-3\frac{\|\mathbf{x} - \mathbf{y}\|}{\beta}\right),$$

where $\|\cdot\|$ is the L^2 norm or Euclidean distance:

$$\|\mathbf{x} - \mathbf{y}\| = \sqrt{\sum_{s=1}^{d} (x_s - y_s)^2}.$$
(6)

In practice, we can set β to be the 95th percentile of all the distances between pairs of the *k* representative VA polices (Isaaks and Srivastava 1990). Again, in distance calculation we also assume that the categorical variables (e.g., gender and guarantee type) are converted to binary dummy variables, and the numerical variables are normalized to have a standard deviation of 1.

The column vector $\mathbf{A}(Z, \mathbf{x}_i)$ and the row vector $\mathbf{B}(\mathbf{x}_i)$ are defined as

$$\mathbf{A}(Z, \mathbf{x}_i) = \begin{pmatrix} \gamma(\mathbf{z}_1, \mathbf{x}_i) \\ \gamma(\mathbf{z}_2, \mathbf{x}_i) \\ \vdots \\ \gamma(\mathbf{z}_k, \mathbf{x}_i) \end{pmatrix}$$
(7)

and

$$\mathbf{B}(\mathbf{x}_i) = \begin{pmatrix} 1 & x_{i1} & \cdots & x_{id} \end{pmatrix},\tag{8}$$

respectively.

The additive property of the dollar Delta ensures that the partial dollar Deltas of the VA portfolio on the *h*th tradable index, when the market level is \mathbf{r}_l , is the aggregated partial dollar Deltas of the individual policies:

$$\hat{f}(X, \mathbf{r}_l, h) = \sum_{i=1}^n \boldsymbol{\lambda}_i^T f(Z, \mathbf{r}_l, h) = \sum_{j=1}^k \sum_{i=1}^n \lambda_{ip} f(\mathbf{z}_p, \mathbf{r}_l, h).$$
(9)

4.2. The Level-Two Metamodel

The level-two metamodel is used to estimate the dollar Deltas of a VA portfolio at an arbitrary market level **r** from the precalculated dollar Deltas at market levels $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_m$, which are selected by the unconditional Latin hypercube sampling method. The reason why the unconditional Latin hypercube sampling method is used to select the market levels is that we do not know the future levels of the market.

The dollar Deltas of an individual VA contract are proportional to some variables used to build the level-one metamodel. However, the dollar Deltas of the VA portfolio are not proportional to market levels, which are used to build the level-two metamodel. As a result, we choose the ordinary kriging method to build the level-two metamodel.

Under an ordinary kriging model, the partial dollar Delta of the portfolio on the *h*th tradable index when the market level is \mathbf{r} is assumed to be (Cressie 1993):

$$g(\mathbf{r}, h) = \mu + \delta(\mathbf{r}, h),$$

where μ is an unknown constant and again $\delta(\cdot, \cdot)$ is a zero-mean intrinsically stationary spatial process. In this model, $g(\mathbf{r}, h)$ can be predicted as (Cressie, 1993):

$$\hat{g}(\mathbf{r},h) = \sum_{l=1}^{m} w_l \cdot \hat{f}(X,\mathbf{r}_l,h), \tag{10}$$

where $\hat{f}(X, \mathbf{r}_l, h)$ is the partial dollar Delta of the portfolio X obtained in Equation (9) and w_1, w_2, \ldots, w_m are the kriging weights. The optimal weights are obtained by solving the following linear equation system:

$$\begin{pmatrix} V_{11} & \cdots & V_{1m} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ V_{m1} & \cdots & V_{mm} & 1 \\ 1 & \cdots & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} w_1 \\ \vdots \\ w_m \\ \theta \end{pmatrix} = \begin{pmatrix} D_1 \\ \vdots \\ D_m \\ 1 \end{pmatrix},$$
(11)

where θ is the Lagrange multiplier to ensure the sum of the kriging weights equal to one:

$$V_{ls} = 1 - \exp\left(-\frac{3}{\beta}\|\mathbf{r}_l - \mathbf{r}_s\|\right), \quad l, s = 1, 2, \dots, m,$$

and

$$D_l = 1 - \exp\left(-\frac{3}{\beta}\|\mathbf{r} - \mathbf{r}_l\|\right), \quad j = 1, 2, \dots, m.$$

Here $\beta > 0$ is a parameter and $\|\cdot\|$ is the L^2 norm (i.e., the Euclidean distance). In practice, we can set β to be the 95th percentile of all the distances between pairs of the *m* selected market levels (Isaaks and Srivastava 1990).

4.3. Procedure and Algorithm

In this subsection, we present the procedure and algorithm of the two-level metamodeling approach to calculate the partial dollar Deltas of a VA portfolio.

We first select a small set of possible market levels $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_m$ in order to precalculate the partial dollar Deltas of the portfolio at these market levels. We select these possible market levels using the maximin Latin hypercube sampling method (Section 3). We then use the precalculated partial dollar Deltas to estimate the partial dollar Deltas of the portfolio at an arbitrary market level.

A level-one metamodel is used to estimate the partial dollar Deltas of a VA portfolio at the selected *m* market levels. They are obtained using the universal kriging method as described in Section 4.1. Since estimating these partial dollar Deltas using a Monte Carlo simulation model at a market level can be done only on an individual policy basis and is time consuming, computation would be prohibitive if the simulation was applied to all possible market levels and to each of the policies in the portfolio.

A level-two metamodel is used to estimate the dollar Deltas of the VA portfolio at an arbitrary market level based on the precalculated dollar Deltas at the *m* market levels. Since there are no trends in the *m* market levels selected by the Latin hypercube sampling method, we can use the ordinary kriging method (see Section 4.2) to estimate the partial dollar Deltas of the portfolio at an arbitrary market level **r**. From Equation (10) we see that the kriging weights are independent of *h*. In other word, we can use the same kriging weights to estimate the partial dollar Deltas of different tradable indices. As a result, the level-two metamodel is extremely fast and can even be used to estimate partial dollar Deltas in real time.

The major steps of building metamodels for efficiently calculating Greeks of the portfolio are as follows:

- 1. Use a conditional Latin hypercube sampling method to select a small set $S_1 = \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k\}$ of VA policies from the portfolio *X*.
- 2. Use a (unconditional) Latin hypercube sampling method to create a small set $S_2 = \{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_m\}$ of possible market levels.
- 3. Use a Monte Carlo valuation system to calculate the partial dollar Delta $f(\mathbf{z}_i, \mathbf{r}_j, h)$ on the *h*th tradable index when the market level is \mathbf{r}_j , for i = 1, 2, ..., k, j = 1, 2, ..., m, and h = 1, 2, ..., H.
- 4. Build a metamodel \mathcal{M}_1 to calculate partial dollar Deltas of the portfolio at the select market levels.
 - a. For each \mathbf{r}_j and h, j = 1, 2, ..., m; h = 1, 2, ..., H, use the universal Kriging model to estimate the partial dollar Deltas $\hat{f}(\mathbf{x}_i, \mathbf{r}_j, h)$ for all the VA policies $\mathbf{x}_i, i = 1, 2, ..., n$.
 - b. Aggregate the partial dollar Deltas of individual policies to obtain the partial dollar Deltas of the portfolio:

$$\hat{f}(X, \mathbf{r}_j, h) = \sum_{i=1}^n \hat{f}(\mathbf{x}_i, \mathbf{r}_j, h)$$

G. GAN AND X. S. LIN

TABLE 1
Some Parameters Used to Generate Synthetic Variable Annuities

Attribute	Values
Guarantee type	{DBRP, DBRU, WB, WBSU, MB}
Gender	{Male, Female}
Birth date range	[Jan. 1 1950, Jan. 1 1980]
Issue date range	Jan. 1 2000, Jan. 1 2014]
Valuation date	Jan. 1 2014
Maturity range	[15, 30]
Account value range	[500,00, 500,000]
Maturity	$\{10, 11, 12, \ldots, 25\}$
Number of investment funds	10

Note: DBRP denotes a GMDB rider with return-of-principal guarantee. DBRU denotes a GMDB rider with an annual roll-up rate of 5%. WB denotes a GMWB rider with an annual withdrawal rate of 6% and a waiting period of 0 years. WBSU denotes a GMWB rider with an annual withdrawal rate of 7% and a waiting period of 10 years. MB denotes a GMMB rider with return-of-principal guarantee.

5. Build another metamodel M_2 from S_2 and the partial dollar Deltas of the portfolio using the ordinary kriging to estimate the partial dollar Deltas of the VA portfolio at an arbitrary market level.

5. NUMERICAL EXPERIMENTS

In this section, we demonstrate the performance of the two-level metamodeling approach using a portfolio of synthetic variable annuity policies. Unlike those used in our previous studies (Gan 2013; Gan and Lin 2015), the synthetic variable annuity policies in this study have multiple investment funds. In addition, the policies in this study have different issue dates.

5.1. A Synthetic Portfolio of Variable Annuities

To test the two-level metamodeling approach, we generate a portfolio of 10,000 synthetic variable annuity policies. Some features and attributes used to generate these policies are given in Table 1. The five guarantee types described in the table are generated uniformly. The gender of a policyholder is generated from a population of 40% females and 60% males. For convenience, all birth dates and issue dates are generated to be the first day of the years. The account values of the 10 investment funds are generated as follows. First an account value is generated uniformly from the given range. Then a subset of the 10 investment funds is selected and the total account value is uniformly divided to these selected funds.

The 10 investment funds are mapped to five tradable indices as shown in Table 2. For example, the sixth investment fund is mapped to 60% U.S. large cap and 40% U.S. small cap. That is, about 60% of the 6th investment fund is invested in U.S. large cap stocks, and the remainder is invested in U.S. small cap stocks.

	Fund Mappings of the 10 Investment Funds							
Fund	U.S. Large	U.S. Small	Intl. Equity	Fixed Income	Money Market			
1	1	0	0	0	0			
2	0	1	0	0	0			
3	0	0	1	0	0			
4	0	0	0	1	0			
5	0	0	0	0	1			
6	0.6	0.4	0	0	0			
7	0.5	0	0.5	0	0			
8	0.5	0	0	0.5	0			
9	0	0.3	0.7	0	0			
10	0.2	0.2	0.2	0.2	0.2			

TABLE 2 Fund Mappings of the 10 Investment Funds

Note: Each row is a mapping from a fund to a combination of five indices.

We implemented the risk-neutral scenario generator, the synthetic VA contract generator, and the Monte Carlo simulation model in Java as open-source software. Interested readers can download the code from a link in Gan (2015) to conduct the experiments.

5.2. Validation Measures

To validate the level-one metamodel, the level-two metamodel, and the combination, we use the following six measures: the (square) root mean squared error (RMSE), relative average absolute error (RAAE), R^2 , relative maximum absolute error (RMAE), average percentage error (APE), and average absolute percentage error (AAPE). These validation measures are defined as follows:

RMSE(h) =
$$\sqrt{\frac{1}{J} \sum_{l=1}^{J} (\hat{y}_{lh} - y_{lh})^2},$$
 (12a)

$$RAAE(h) = \frac{\sum_{l=1}^{J} |\hat{y}_{lh} - y_{lh}|}{J \times \sigma_h},$$
(12b)

$$R_h^2 = 1 - \frac{\sum_{l=1}^J (\hat{y}_{lh} - y_{lh})^2}{\sum_{l=1}^J (\mu_h - y_{lh})^2}$$
(12c)

$$\operatorname{RMAE}(h) = \frac{\max_{1 \le l \le J} |\hat{y}_{lh} - y_{lh}|}{\sigma_h},$$
(12d)

$$APE(h) = \frac{1}{J} \sum_{l=1}^{J} \frac{\hat{y}_{lh} - y_{lh}}{y_{lh}},$$
(12e)

$$AAPE(h) = \frac{1}{J} \sum_{l=1}^{J} \left| \frac{\hat{y}_{lh} - y_{lh}}{y_{lh}} \right|,$$
 (12f)

for h = 1, 2, ..., H, where μ_h and σ_h are the sample mean and sample standard deviation of $y_{1h}, y_{2h}, ..., y_{Jh}$, respectively.

~

The first four validation measures are commonly used to validate metamodels (Zhu et al. 2012). Since the partial dollar Deltas on different tradable indices may be different in magnitude, we calculate the validation measures separately for different tradable indices.

To validate the level-one metamodel, we set

$$\hat{y}_{lh} = f(X, \mathbf{r}_l, h), \quad y_{lh} = f(X, \mathbf{r}_l, h), \quad J = k,$$

where $\hat{f}(X, \mathbf{r}_l, h)$ is the partial dollar Delta estimated by the level-one metamodel, and $f(X, \mathbf{r}_l, h)$ is the partial dollar Delta calculated by the Monte Carlo simulation model. How partial dollar Deltas are calculated is shown in the Appendix.

To validate the level-two metamodel alone, we set

$$\hat{y}_{lh} = g(\mathbf{s}_l, h), \quad y_{lh} = f(X, \mathbf{s}_l, h), \quad J = m,$$

	TABLE 3			
Monthly Volatilities and Correlation	Coefficients between	the Five	Tradable	Indices

	Volatility			Correlation		
U.S. Large (Index 1)	0.033	1	0.7619	0.5571	0.2369	- 0.0383
U.S. Small (Index 2)	0.0448	0.7619	1	0.4433	0.1303	-0.0334
Intl. Equity (Index 3)	0.0372	0.5571	0.4433	1	0.1505	-0.0347
Fixed Income (Index 4)	0.0094	0.2369	0.1303	0.1505	1	0.036
Money Market (Index 5)	0.0011	-0.0383	-0.0334	-0.0347	0.36	1

where $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m$ are *m* possible market levels for testing, $g(\mathbf{s}_l, h)$ is the partial dollar Delta estimated by the level-two metamodel with the partial dollar Deltas calculated by the Monte Carlo simulation model, and $f(X, \mathbf{s}_l, h)$ is the partial dollar Delta calculated by the Monte Carlo simulation model. In particular, $g(\mathbf{s}_l, h)$ is defined as

$$g(\mathbf{s}_l, h) = \sum_{j=1}^m w_j \cdot f(X, \mathbf{r}_l, h),$$

where w_1, w_2, \ldots, w_m are the kriging weights from the level-two metamodel and $f(X, \mathbf{r}_l, h)$ is the partial dollar Deltas calculated by the Monte Carlo simulation model. We define the validation measures for the level-two metamodel in this way so that the errors caused by the level-one metamodel are excluded.

To validate the two-level metamodeling approach with the level-one and level-two metamodels together, we set

$$\hat{y}_{lh} = \hat{g}(\mathbf{s}_l, h), \quad y_{lh} = f(X, \mathbf{s}_l, h), \quad J = m,$$

where $\hat{g}(\mathbf{s}_l, h)$ is the partial dollar Delta estimated by the level-two metamodel with the partial dollar Deltas estimated by the level-one metamodel (see Eq. [10]).



FIGURE 2. The Scatter Plots of Partial Dollar Deltas. *Note:* The *x* axis (i.e., the horizontal axis) shows the partial dollar Deltas calculated by the Monte Carlo simulation model. The *y* axis (i.e., the vertical axis) shows the partial dollar Deltas estimated by the two-level metamodeling approach. The straight lines are the lines y = x. In this test, k = 220 and m = 50.

EFFICIENT GREEK CALCULATION OF VARIABLE ANNUITY PORTFOLIOS

5		0 11				
RMSE	RAAE	R^2	RMAE	APE	AAPE	
4,698,664	0.5811	0.6316	0.8675	-0.0419	0.0419	
2,302,026	0.4021	0.8188	0.7146	-0.0317	0.0317	
4,988,233	0.9079	0.1395	1.2809	-0.0609	0.0609	
3,746,782	0.7114	0.4688	0.9593	-0.047	0.047	
4,496,789	1.2115	-0.5247	1.6475	0.0717	0.0717	
	RMSE 4,698,664 2,302,026 4,988,233 3,746,782 4,496,789	RMSE RAAE 4,698,664 0.5811 2,302,026 0.4021 4,988,233 0.9079 3,746,782 0.7114 4,496,789 1.2115	RMSE RAAE R ² 4,698,664 0.5811 0.6316 2,302,026 0.4021 0.8188 4,988,233 0.9079 0.1395 3,746,782 0.7114 0.4688 4,496,789 1.2115 -0.5247	RMSE RAAE R ² RMAE 4,698,664 0.5811 0.6316 0.8675 2,302,026 0.4021 0.8188 0.7146 4,988,233 0.9079 0.1395 1.2809 3,746,782 0.7114 0.4688 0.9593 4,496,789 1.2115 -0.5247 1.6475	RMSE RAAE R ² RMAE APE 4,698,664 0.5811 0.6316 0.8675 - 0.0419 2,302,026 0.4021 0.8188 0.7146 - 0.0317 4,988,233 0.9079 0.1395 1.2809 - 0.0609 3,746,782 0.7114 0.4688 0.9593 - 0.047 4,496,789 1.2115 - 0.5247 1.6475 0.0717	

TABLE 4Accuracy of Two-Level Metamodeling Approach with k = 220 and m = 50

5.3. Results

To test the accuracy and speed of the two-level metamodeling approach, we randomly generate 60 market levels from a multivariate normal distribution with zero mean and a covariance matrix obtained from the volatilities and correlations given in Table 3. Note that these testing market levels are different from the prechosen market levels selected by the Latin hypercube sampling method. The testing market levels simulate how the market moves in future, while the prechosen market levels are created to fill the space of possible future market levels for metamodeling purposes.

We want to compare the partial dollar Deltas estimated from the two-level metamodeling approach and those calculated by a Monte Carlo simulation. In the Monte Carlo simulation model, we used 1000 risk-neutral scenarios and 30 annual time steps for cash flow projection.

We test the two-level metamodeling approach with different values of k and m. As suggested in Loeppky et al. (2009), the number of Latin hypercube designs should be 10 times the number of dimensions. After converting categorical variables (e.g., gender and guarantee type) to binary dummy variables, a policy has 22 variables, which include 10 investment funds, age, time to maturity, etc. For the level-two metamodel, a market level has five dimensions because we consider only five indices. In our first test, we use k = 220 (0.22% of the total number of VA policies) and m = 50; that is, we select 220 policies from the portfolio to build the level-one metamodel and select 50 possible market levels to build the level-two metamodel. To select the sample policies and the market levels, we use the clhs function and the lhs function in R (Roudier 2011; Carnell 2012), respectively. The R code for selecting the sample policies and the market levels is given in the Appendix.

For the first test, the scatter plots of the partial dollar Deltas estimated by the two-level metamodeling approach and those calculated by the Monte Carlo simulation model are shown in Figure 2. We see that there are biases in the estimated partial dollar Deltas. The accuracy of the two-level metamodeling approach is shown in Table 4. We see that the average absolute percentage errors calculated from the 60 test market levels range from 3.17% to 7.17%. We also see that the R^2 calculated from the predicted partial dollar Deltas on the 5th index is negative. The negative R^2 is caused by the fact that the sum of squared deviations between the partial dollar Deltas estimated by the two-level approach and those calculated by the Monte Carlo simulation model is larger than the variance of the partial dollar Deltas calculated by the Monte Carlo simulation model.

The values of the validation measures shown in Table 4 contains the errors from the level-one metamodel and the level-two metamodel. Tables 5 and 6 give the values of the validation measures that contain only errors from the level-one metamodel and the level-two metamodel, respectively. From these tables we see that the level-two metamodel is fairly accurate in that all the R^2 are larger than 99% and the average absolute percentage errors are less than 1%. However, Table 5 shows that the level-one metamodel has relatively large prediction errors. We can conclude that most of the prediction error of the two-level metamodeling approach comes from the level-one metamodel.

	Accuracy of Lever-one instantouch with $k = 220$ and $m = 50$							
	RMSE	RAAE	R^2	RMAE	APE	AAPE		
Index 1	4,901,431	0.3274	0.8703	0.6525	- 0.043	0.0431		
Index 2	3,085,133	0.2763	0.8884	0.6799	-0.0335	0.0378		
Index 3	5,076,421	0.4822	0.7307	0.9234	-0.0602	0.061		
Index 4	3,814,150	0.3984	0.8175	0.6768	-0.0466	0.0466		
Index 5	4,579,153	0.7101	0.4679	1.0208	0.0729	0.0729		

TABLE 5Accuracy of Level-One Metamodel with k = 220 and m = 50

Accuracy of Level-Two Metamodel with $k = 220$ and $m = 50$							
	RMSE	RAAE	R^2	RMAE	APE	AAPE	
Index 1	632,799	0.066	0.9933	0.1923	- 0.0012	0.0047	
Index 2	428,126	0.0655	0.9937	0.1836	-0.0011	0.0051	
Index 3	364,905	0.0515	0.9954	0.1664	-0.0015	0.0035	
Index 4	390,177	0.06	0.9942	0.1712	-0.0014	0.004	
Index 5	311,483	0.0694	0.9927	0.2025	- 9.00E-04	0.0042	

TABLE 6



FIGURE 3. Scatter Plots of Partial Dollar Deltas. Note: The x axis (i.e., the horizontal axis) shows the partial dollar Deltas calculated by the Monte Carlo simulation model. The y axis (i.e., the vertical axis) shows the partial dollar Deltas estimated by the two-level metamodeling approach. The straight lines are the lines y = x. In this test, k = 440 and m = 50.

EFFICIENT GREEK CALCULATION OF VARIABLE ANNUITY PORTFOLIOS

	5 II						
	RMSE	RAAE	R^2	RMAE	APE	AAPE	
Index 1	2,781,176	0.3327	0.8709	0.6853	0.0233	0.0233	
Index 2	2,124,997	0.3718	0.8456	0.6566	-0.0293	0.0293	
Index 3	1,808,466	0.2776	0.8869	0.7427	-0.0182	0.0193	
Index 4	2,142,709	0.3766	0.8263	0.7322	0.024	0.024	
Index 5	803,153	0.1776	0.9514	0.5359	0.004	0.0104	

TABLE 7Accuracy of Two-Level Metamodeling Approach when k = 440 and m = 50

TABLE 8Accuracy of Level-One Metamodel when k = 440 and m = 50

	RMSE	RAAE	R^2	RMAE	APE	AAPE
Index 1	3,733,350	0.226	0.9247	0.6062	0.0246	0.0283
Index 2	2,530,037	0.2294	0.925	0.5583	-0.0302	0.0316
Index 3	1,953,643	0.164	0.9601	0.3728	-0.0195	0.0218
Index 4	2,479,114	0.231	0.9229	0.6286	0.0246	0.0255
Index 5	803,141	0.1044	0.9836	0.3303	0.0036	0.0105

The first test shows that most of the prediction error of the two-level metamodeling approach comes from the level-one metamodel. To increase the accuracy of the level-one metamodel, we increase the number of policies used to build the level-one metamodel. In our second test, we use k = 440 (0.44% of the total number of VA policies) and m = 50; that is, we double the number of policies to build the level-one metamodel. Figure 3 shows the scatter plots of the partial dollar Deltas estimated by the two-level metamodeling approach and those calculated by the Monte Carlo simulation model. Comparing Figures 2 and 3 we see that the biases reduced. The accuracy of the two-level metamodeling approach is shown in Table 7. In this case, all R^2 are positive and higher than 80%. Increasing the number of sample policies actually increased the accuracy of the approach. Table 7 also shows that the average absolute percentage errors are less than 3% on all the indices.

The accuracy of the level-one metamodel and that of the level-two metamodel when considered separately are shown in Tables 8 and 9, respectively. The result shows that most of the error comes from the level-one metamodel.

Table 10 shows the run time used by the two-level metamodeling approach and the Monte Carlo simulation model. We see that it took the Monte Carlo simulation model about 42,090 seconds or 11.69 hours to calculate all the partial dollar Deltas at the 60 market levels. If we use the Monte Carlo simulation model to calculate the partial dollar Deltas of the portfolio at a single market level, it will take about 11.69 minutes. This is the run time for calculating the partial dollar Deltas of a portfolio of 10,000 policies. If a portfolio contains 300,000 policies, then it will take the Monte Carlo simulation model about 350.7 minutes or 5.85 hours to just calculate the partial dollar Deltas of the portfolio at a single market level.

If we use the two-level metamodeling approach, the run time of calculating the partial dollar Deltas at the 60 test market levels reduces to 1,465.55 seconds or 24.2 minutes if we use 440 sample policies. In practice, we can build the level-one metamodel and the level-two metamodel overnight and then use the level-two metamodel to estimate the partial dollar Deltas during the trading

Accuracy of Level-1 wo interantouch when $k = 440$ and $m = 50$							
	RMSE	RAAE	R^2	RMAE	APE	AAPE	
Index 1	632,799	0.066	0.9933	0.1923	-0.0012	0.0047	
Index 2	428,126	0.0655	0.9937	0.1836	-0.0011	0.0051	
Index 3	364,905	0.0515	0.9954	0.1664	-0.0015	0.0035	
Index 4	390,177	0.06	0.9942	0.1712	-0.0014	0.004	
Index 5	311,483	0.0694	0.9927	0.2025	-9.00E-04	0.0042	

TABLE 9Accuracy of Level-Two Metamodel when k = 440 and m = 50

Deltas at the of Test Market Levels					
k = 220, m = 50	k = 440, m = 50	Full, $m = 60$			
138.75	169.44	NA			
639.55	1279.10	42,090.08			
8.12	16.66	NA			
0.02	0.02	NA			
0.19	0.33	NA			
786.63	1465.55	42,090.08			
	k = 220, m = 50 138.75 639.55 8.12 0.02 0.19 786.63	behas at the 60 Test Market Levels $k = 220, m = 50$ $k = 440, m = 50$ 138.75 169.44 639.55 1279.10 8.12 16.66 0.02 0.02 0.19 0.33 786.63 1465.55			

TABLE 10 Runtime of Using Two-Level Metamodeling Approach and Monte Carlo Simulation Model to Calculate all the Partial Dollar Deltas at the 60 Test Market Levels

Note: The numbers are in seconds.

TABLE 11 Parameter Values for Generating Jumps under Merton's Jump-Diffusion Model Index λ μ' σ' U.S. Large 3 -0.100.30 3 U.S. Small -0.200.40 3 -0.15Intl. Equity 0.35

Note: λ represents the rate of the Poisson process, μ' is the mean of the jump, and σ' is the standard deviation of the jump.

TABLE 12

Accuracy of Two-Level Metamodeling Approach When Market Levels Have Jumps, k = 440, and m = 50

	RMSE	RAAE	R^2	RMAE	APE	AAPE
Index 1	4,495,497	0.3369	0.7842	1.6785	0.0206	0.0291
Index 2	4,356,134	0.3543	0.7023	2.7671	-0.0333	0.0385
Index 3	2,520,201	0.2683	0.8719	1.1444	-0.0166	0.0241
Index 4	3,249,667	0.3428	0.7909	1.8893	0.0206	0.03
Index 5	1,949,499	0.2089	0.8437	2.1182	9.00E-04	0.016

TABLE 13

Accuracy of Level-Two	Metamodel Whe	n Market Level	s Have Jumps,	k = 440, and $m = 5$	0
-----------------------	---------------	----------------	---------------	----------------------	---

	DMCE	DAAE	D ²	DMAE	A DE	
	RNISE	KAAE	Κ	KMAE	APE	AAPE
Index 1	3,661,974	0.1416	0.8568	1.9013	- 3.50E-03	0.0117
Index 2	3,730,780	0.1682	0.7817	2.6319	-0.0062	0.0173
Index 3	2,132,426	0.1303	0.9083	1.4386	4.00E-04	0.0117
Index 4	2,932,530	0.1559	0.8297	2.2952	-4.40E-03	0.0132
Index 5	2,030,721	0.1587	0.8304	2.3169	- 3.80E-03	0.012

hours of the next day. From Table 10 we see that it took the level-two metamodel about 0.33 seconds to estimate the partial dollar Deltas at the 60 test market levels. The result shows that the level-two metamodel is very fast.

The above numerical experiments show that the two-level metamodeling approach works well when the market levels follow a multivariate normal distribution. To test the accuracy of the two-level metamodeling approach when market levels have jumps, we generated 60 market levels by adding jumps to the three equity indices (i.e., U.S. Large Cap Equity, U.S. Small Cap Equity, and International Equity). We followed Merton's jump-diffusion model to generate the jumps. Under Merton's jump diffusion model,



FIGURE 4. Scatter Plots of Partial Dollar Deltas. *Note:* The x axis (i.e., the horizontal axis) shows the partial dollar Deltas calculated by the Monte Carlo simulation model. The y axis (i.e., the vertical axis) shows the partial dollar Deltas estimated by the two-level metamodeling approach. The straight lines are the lines y = x. In this test, the market levels have jumps, k = 440, and m = 50.

the stock price is modeled as (Merton 1976; Kou and Wang 2004):

$$\ln \frac{S_t}{S_0} = \mu t + \sigma W_t + \sum_{i=1}^{N_t} Y_i,$$
(13)

where $\{W_t : t \ge 0\}$ is a standard Brownian motion, $\{N_t : t \ge 0\}$ is a Poisson process with rate λ , $\{Y_n : n = 1, 2, ..., \}$ are independent and normally distributed variables with mean μ' and standard deviation σ' , μ is a drift parameter, and σ denotes the volatility of the stock return. The parameters we used are given in Table 11. The parameter $\lambda = 3$ was obtained from (Kou and Wang 2004), indicating that there are average three jumps per year. In this test, we use the same 440 representative VA contracts and the same 50 prechosen market levels used in the second test.

Table 12 shows the accuracy of the two-level metamodeling approach. Comparing Tables 7 and 12, we see that the accuracy of the two-level metamodeling approach decreases when the market levels have jumps. R^2 decreases and the RMSE increases. Since we used the same prechosen market levels in this test as those used in the second test, the accuracy of the level-one metamodel is the same as shown in Table 8. The accuracy of the level-two metamodel is shown Table 13. Comparing Tables 9 and 13, we

see that the accuracy of the level-two metamodel is affected by the jumps. The RMSE increases significantly. Figure 4 shows the scatter plots of the partial dollar Deltas calculated by the Monte Carlo simulation model and those estimated by the two-level metamodeling approach. We see that the partial dollar Deltas estimated by the two-level metamodeling approach do not match those calculated by the Monte Carlo simulation model at the market levels with jumps.

In summary, the test results show that the two-level metamodeling approach is much faster than the Monte Carlos simulation model and produces acceptable estimates of the partial dollar Deltas used in dynamical hedging. In particular, the proposed approach performs well in terms of accuracy when the market levels follow multivariate normal distribution. The accuracy of the proposed approach decreases when the market levels have jumps. However, the results are still acceptable because the absolute average percentage errors are less than 5%.

6. CONCLUDING REMARKS

Dynamic hedging is one of the most efficient risk management methods to mitigate the financial risk arising from VA. In dynamic hedging, the dollar Deltas of the liability need to be calculated so that the hedge positions can be adjusted. To rebalance the hedge portfolio against the liability of a VA portfolio in a timely manner, we need to estimate the dollar Deltas of the VA portfolio within a short time period.

In this article, we proposed a two-level metamodeling approach to estimate the partial dollar Deltas of a large VA portfolio under a multiasset framework. The idea behind the two-level metamodeling approach is to precalculate the partial dollar Deltas at a small set of well-designed market levels and then estimate the partial dollar Deltas at a new market level based on the precalculated partial dollar Deltas. In the proposed two-level metamodeling approach, we used universal kriging to estimate the partial dollar Deltas at new market levels. The sample policies and market level used to construct the metamodels were selected by a conditional Latin hypercube sampling method, respectively. Note that our approach is different from the least square Monte Carlo method (Bauer and Ha 2013) in that the latter focuses on one contract.

To test the performance of the two-level metamodeling approach, we used a Monte Carlo simulation model with 1000 riskneutral scenarios and 30 annual time steps for cash flow projection. The partial dollar Deltas calculated by the Monte Carlo simulation model are used as benchmarks to validate the proposed approach. The test results have shown that the proposed twolevel metamodeling approach performs well in terms of accuracy and speed. In our test, we assumed that the interest rate does not change. However, the two-level metamodeling approach can be extended to handle nonconstant interest rates straightforwardly.

Our experiments have shown that most of the prediction error of the two-level metamodeling approach comes from the levelone metamodel. In the future, we would like to improve the level-one metamodel by considering the errors of the Monte Carlo simulation model.

FUNDING

Natural Sciences and Engineering Council of Canada (NSERC): 458146.

REFERENCES

Alam, F. M., K. R. McNaught, and T. J. Ringrose. 2004. A comparison of experimental designs in the development of a neural network simulation metamodel. Simulation Modelling Practice and Theory 12(7): 559–578.

Ankenman, B., B. L. Nelson, and J. Staum. 2010. Stochastic kriging for simulation metamodeling. Operations Research 58(2): 371–382.

Bacinello, A. R., P. Millossovich, and A. Montealegre. 2014. The valuation of GMWB variable annuities under alternative fund distributions and policyholder behaviours. *Scandinavian Actuarial Journal* 2014: 1–20.

Barton, R. 1994. Metamodeling: a state of the art review. In Winter Simulation Conference Proceedings, edited by J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila, pp. 237–244. Piscataway, NJ: IEEE.

Bauer, D., and H. Ha. 2013. A least-squares Monte Carlo approach to the calculation of capital requirements. Department of Risk Management and Insurance, Georgia State University.

Bauer, D., A. Kling, and J. Russ. 2008. A universal pricing framework for guaranteed minimum benefits in variable annuities. ASTIN Bulletin 38(2): 621-651.

Bélanger, A., P. Forsyth, and G. Labahn. 2009. Valuing the guaranteed minimum death benefit clause with partial withdrawals. *Applied Matehmatical Finance* 16(6): 451–496.

Box, G. E. P., and N. R. Draper. 2007. Response Surfaces, Mixtures, and Ridge Analyses. 2nd ed. Hoboken, NJ: Wiley.

Caballero, W., R. Giraldo, and J. Mateu. 2013. A universal kriging approach for spatial functional data. *Stochastic Environmental Research and Risk Assessment* 27(7): 1553–1563.

Carnell, R. 2012. Ihs: Latin Hypercube Samples. R package version 0.10. Vienna: R Foundation.

Cressie, N. 1993. Statistics for Spatial Data. Rev. ed. Hoboken, NJ: Wiley.

Das, R. N. 2014. Robust Response Surfaces, Regression, and Positive Data Analyses. Boca Raton, FL: CRC Press.

Friedman, L. W. 1996. The Simulation Metamodel. Norwell, MA: Kluwer.

Gan, G. 2011. Data Clustering in C++: An Object-Oriented Approach. Data Mining and Knowledge Discovery Series. Boca Raton, FL: Chapman & Hall/CRC Press.

Gan, G. 2013. Application of data clustering and machine learning in variable annuity valuation. Insurance: Mathematics and Economics 53(3): 795-801.

- Gan, G. 2015. A multi-asset Monte Carlo simulation model for the valuation of variable annuities. In *Proceedings of the Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, pp. 3162–3163. Piscataway, NJ: IEEE.
- Gan, G., and X. S. Lin. 2015. Valuation of large variable annuity portfolios under nested simulation: A functional data approach. *Insurance: Mathematics and Economics* 62(1): 138–150.

Gerber, H., and E. Shiu. 2003. Pricing lookback options and dynamic guarantees. North American Actuarial Journal 7(1): 48-67.

Huang, H., M. Milevsky, and T. Salisbury. 2014. Optimal initiation of a GLWB in a variable annuity: No arbitrage approach. *Insurance: Mathematics and Economics* 56: 102–111.

Isaaks, E., and R. Srivastava. 1990. An Introduction to Applied Geostatistics. Oxford: Oxford University Press.

Kleijnen, J. P. 2009. Kriging metamodeling in simulation: A review. European Journal of Operational Research 192(3): 707–716.

Kleijnen, J. P. C. 1975. A comment on Blanning's "metamodel for sensitivity analysis: The regression metamodel in simulation." Interfaces 5(3): 21–23.

Kou, S. G., and H. Wang. 2004. Option pricing under a double exponential jump diffusion model. Management Science 50(9): 1178–1192.

Liefvendahl, M., and R. Stocki. 2006. A study on algorithms for optimization of Latin hypercubes. Journal of Statistical Planning and Inference 136(9): 3231–3247.

Lin, X. S., K. S. Tan, and H. L. Yang. 2009. Pricing annuity guarantees under a regime-switching model. North American Actuarial Journal 13: 316–338.

- Loeppky, J. L., J. Sacks, and W. J. Welch. 2009. Choosing the sample size of a computer experiment: A practical guide. Technometrics 51(4): 366–376.
- Merton, R. C. 1976. Option pricing when underlying stock returns are discontinuous. Journal of Financial Economics 3(1): 125–144.
- Milevsky, M., and S. Posner. 2001. The titanic option: Valuation of the guaranteed minimum death benefit in variable annuities and mutual funds. *Journal of Risk and Insurance* 68(1): 93–128.
- Minasny, B., and A. B. McBratney. 2006. A conditioned latin hypercube method for sampling in the presence of ancillary information. *Computers & Geosciences* 32(9): 1378–1388.
- Moon, H., A. Dean, and T. Santner. 2011. Algorithms for generating maximin latin hypercube and orthogonal designs. *Journal of Statistical Theory and Practice* 5(1): 81–98.
- Ng, A. C.-Y., and J. S.-H. Li. 2013. Pricing and hedging variable annuity guarantees with multiasset stochastic investment models. North American Actuarial Journal 17(1): 41–62.

Reynolds, C., and S. Man. 2008. Nested stochastic pricing: The time has come. Product Matters! - Society of Actuaries 71: 16-20.

Roudier, P. 2011. clhs: a R Package for Conditioned Latin Hypercube Sampling. Vienna: R Foundation.

- Viana, F. 2013. Things you wanted to know about the Latin hypercube design and were afraid to ask. In *10th World Congress on Structural and Multidisciplinary Optimization*, pp. 1–9. Orlando, FL.
- Zhu, H., L. Liu, T. Long, and L. Peng. 2012. A novel algorithm of maximin latin hypercube design using successive local enumeration. *Engineering Optimization* 44(5): 551–564.

Discussions on this article can be submitted until January 1, 2018. The authors reserve the right to reply to any discussion. Please see the Instructions for Authors found online at http://www.tandfonline.com/uaaj for submission instructions.

APPENDIX

A.1. Calculation of Partial Dollar Deltas

In Monte Carlo simulation, we calculate the partial dollar Deltas of a variable annuity policy on the *h*th tradable index as follows:

Dollar Delta(**r**, h) =
$$\frac{FMV(AV_1^*, \dots, AV_{h-1}^*, 1.01AV_h^*, AV_{h+1}^*, \dots, AV_H^*)}{0.02} - \frac{FMV(AV_1^*, \dots, AV_{h-1}^*, 0.99AV_h^*, AV_{h+1}^*, \dots, AV_H^*)}{0.02},$$

where $\mathbf{r} = (r_1, r_2, ..., r_H)$ is the market level of the *H* tradable indices, $AV_h^* = (1 + r_h)AV_h$ is the adjusted partial account value based on the market level, and AV_h is the base partial account value calculated from the investment funds based on the fund mappings. Here we used 1% shock to calculate the dollar Deltas. The Monte Carlo simulation model for calculating the partial dollar Deltas is implemented in Java as open-source software (Gan 2015).

A.2. R Code for Experimental Designs

The R code used to select sample policies and market levels is given below:

```
set.seed(1)
S <- clhs(inforce, size=220)
set.seed(1)
X <- maximinLHS(n, k)
for(i in c(1:n)) {
X[i,] <- round(X[i,] * 2 * sigma - sigma, 4)
}</pre>
```

The sample policies are selected by the clhs function from the clhs R package. The market levels are created by the maximinLHS function from the lhs R package. We fixed the seed so that we can repeat the same experiment.