PV and XVA Greeks for Callable Exotics by Algorithmic Differentiation

A. Antonov, S. Issakov, M. Konikov, A. McClelland, and S. Mechkov
Numerix R&D

December 9, 2016

Abstract

We generalize the algorithmic differentiation method proposed by Antonov (2016) from price Greeks to XVA Greeks. This method, named backward differentiation (BD), was developed in the context of computing price or PV Greeks for individual callable exotic trades.

We start by treating cases where cashflow derivatives are sufficient for computing PV/XVA Greeks, i.e., where the differentiation of conditional expectations (or regression functions) is not necessary. For example, PV Greeks for Bermudan swaptions can be computed without having to perform the complicated step of regression function differentiation. We modify the BD algorithm to calculate Greeks for such instruments: the method is applied during the backward pricing procedure and has almost no overhead with respect to a pure backward pricing (without the Greeks).

A general XVA calculation cannot be done using only the cashflow derivatives—some exceptions are listed in this article—instead, the differentiation of future instrument values that are results of the regression may be required. We leverage the algorithmic calculation of future values (algorithmic exposures) proposed in [2] and describe adjoint differentiation (AD) and the new BD for XVA Greeks. The latter algorithm is much simpler than the former, in particular, it does not require the use of the instrument tape, i.e., it does not require the storage of certain payoff derivatives during the pricing procedure as is the case for AD. At the same time, both AD and BD enjoy a similar level of performance.

1 Introduction

Greeks, or sensitivity calculations, are complicated and time-consuming operations performed by financial libraries. Traditionally, Greeks have been used by traders who needed them to hedge their risk. After the financial crisis, an extra layer of complexity was added as banks began systematically hedging different valuation adjustments, starting with the Credit Valuation Adjustment (CVA). These valuation adjustments, often referred to as XVAs (with FVA, KVA, and MVA being other major constituents), depend on the future exposure of a portfolio of instruments (at the netting set level and above in the hierarchy). As a result, we require a distribution of future prices for even a simple swap position; this can be handled using Monte Carlo (MC) methods.

Aside from the hedging perspective, Greeks rapidly became more important for operational and regulatory needs. Recent initial margin rules, including the ISDA standard initial margin model (SIMM), require sensitivity calculations for non-cleared deals, such as callable exotics or structured products, for a large set of risk factors on a daily basis. Note that while the SIMM model requires the calculation of price sensitivities to determine today's initial margin, the calculation of MVA, which reflects margin requirements over the life of a trade or a portfolio, requires sensitivities of future values or "future Greeks" to determine the future margin requirements. This requirement represents a significant quantitative challenge. In a similar vein, the regulatory Fundamental Review of the Trading Book initiaitives, FRTB-MR and FRTB-CVA, require a massive computational effort in the calculation of multiple sensitivities for both PVs and CVAs. The most common method of computing a sensitivity is the bump-and-reprice method, which is widely used due to the simplicity and universality of its implementation. The main drawback of the bump-and-reprice method is its slow speed: to compute an extra Greek, one needs to rerun the pricing calculation of sensitive instruments. Modern methods for computing Greeks, such as payoff differentiation or likelihood methods, are much faster than the bump-and-reprice method but are

Electronic copy available at: https://ssrn.com/abstract=2881992

more complicated and less universal. They deliver a Greek as a mathematical derivative of a PV or XVA over a given parameter and must be tailored to the specific model or payoff.

The FRTB requirements are not entirely clear as to whether the more advanced methods for computing Greeks are allowed under its given definitions of sensitivities. In the current documentation (see, e.g., paragraph 67a of BCBS352) an interest rate delta is defined as a change in the instrument's market value with respect to a 1 basis point (bp) bump of a given rate divided by this 1 bp bump. In other words, under the FRTB, the Greek is not a mathematical derivative corresponding to an infinitesimal "bump" but rather is its finite difference analog. On the other hand, according to [11], such a restriction was not intentional. Thus, it is possible that modern powerful techniques will be authorized for the calculation of FRTB sensitivities. Such approaches were discussed at a recent ISDA Workshop [10]. We stress the importance of the fast and careful treatment of exotic and semi-exotic products for both PV and XVA sensitivities used for managing the entire portfolio. When referring to exotic instruments we mean callable structured products (e.g., Bermudan swaptions), while by semi-exotics we mean products with non-vanilla coupons (with barriers, path-dependency, etc.) but without early exercise features. Clearly, the proportion of such instruments in an average portfolio is usually not very high. However, their presence still complicates the calculation of prices and sensitivities at the portfolio level. Consider first the PV Greeks, e.g., those required by the FRTB: simple linear instruments and European options are likely to be valued via a formula or approximated analytically. This means that their Greeks can be easily and rapidly computed via the bump-and-reprice method. Exotics and semi-exotics are in general treated by Monte Carlo or finite difference methods, which makes their treatment much more complicated.

Let us turn now to the issue of XVA sensitivities (e.g., FRTB-CVA and CVA hedging). Even for the portion of the portfolio that consists of linear instruments, the CVA sensitivity calculation is not trivial and eventually requires the use of modern, rapid methods given the number of sensitivities required and the time taken to compute the CVA. If simple instruments' exposures are available analytically, one may be able to proceed with the bump-and-reprice approach. This exercise, however, becomes suddenly more complicated if we add just a single callable product to the portfolio. Thus, it is very important to consider exotics' Greeks, which we will focus on in the details below. A traditional payoff differentiation is applied path-by-path to forward instruments. It can either be direct (i.e., forward in time) or adjoint (i.e., backward in time). Direct differentiation is more efficient when the number of results (outputs) is greater than the number of parameters. Adjoint differentiation (AD) is more efficient in the opposite case and is more natural in finance, where we seek the sensitivity of a single output (the CVA) against the many risk factors influencing the portfolio.

Following the seminal paper by Giles and Glasserman in 2006 [9], AD was introduced to mathematical finance by Capriotti and Giles [4, 5] as a general method for the efficient calculation of sensitivities and has gained wide popularity. Andreasen [1] was the first to apply AD to the complicated world of XVA.

AD coding is technically difficult and requires a substantial effort: it can sometimes lead to the rewriting of entire libraries. There are two main ways to implement AD:

- The first method is a manual arrangement of differentiation capabilities, i.e., the hand-coding of the algorithm.
- The second way is an automatic overload of the double type in the code to add differentiation capabilities to all of the operations containing double variables. To be efficient, this approach requires a significant amount of work, primarily in terms of deciding which variables to differentiate and which variables are intermediate (i.e., are not to be differentiated). It is also complicated to handle non-differentiable operations such as comparison, which may be very common in the code.

In general, the manual and automatic methods deliver the same performance if they are properly implemented. The technique discussed in this article falls under the category of the manual modification of an existing pricing library.

Another important aspect of AD is that it is an adjoint action, meaning that it takes place after the main (pricing) calculation. Therefore, it is necessary to record information that is generated during pricing that will be utilized later on for the AD. All of the pricing operations are recorded in a data structure commonly referred to as the tape. During the adjoint pass, the tape is "played back" to perform the differentiation using the recorded information. The tape adds a lot of complexity to the AD implementation in the form of potential memory issues due to large storage demands, coding, debugging, and maintenance difficulties, etc.

Callable instrument prices and XVA measures are generally calculated using Monte Carlo regression (also known as least-squares Monte Carlo or American Monte Carlo). In general, present values of callable trades are obtained by the backward repeated application of regression and arithmetic operations encoded in the so-called pricing (payoff) script. The main component of the backward pricing procedure

Electronic copy available at: https://ssrn.com/abstract=2881992

is a continuation value (CV). The CV is a function of the factors driving the model. Financially, the CV is a hold value: it is the price of holding an option at some time with all possible payments and exercises taking place thereafter. A simple CV example is a leg with a single unit payment at some time. More complicated examples include a swap value or a swaption holding value. Performing a regression is one of the most complex and time-consuming numerical methods used in pricing. It introduces path interdependencies that compromise the ability to perform path-by-path valuation and differentiation. The value of a given path at a particular point in time can depend on the values of all paths (including itself) at previous time steps. This property makes a direct AD application difficult.

Despite these complications, the AD of the regression-based pricing functions is appealing because, unlike the bump-and-reprice approach, it avoids the repeated regression encountered during the instruments repricing. The first detailed description of AD for exotic instruments with full regressions appeared in [3]. In that paper a technical recipe was given for the traditional application of the AD approach to general instruments. As the main result of that paper, a new differentiation technique, named "backward differentiation" (BD), was proposed. It is applied at the time of pricing and completely avoids the instrument tape and its related complications. The BD method is applied to the CVs: it keeps track of the mathematical operations associated to the CVS and updates their derivatives over the model parameters accordingly.

It is important to note that price Greeks do not always require regression differentiation: analytically-known derivatives of cashflows can be sufficient. This is true, for example, for Bermudan swaptions. Moreover, for certain cases, XVA Greeks do not require regression differentiation either; the related methods were explained in [1]. In this paper, we systematically study such cases and propose a list of conditions under which regression differentiation is not necessary. As a by-product of our analysis we come up with a simple modification of the pricing BD that ignores the regression derivatives and makes the algorithm extremely simple and efficient. Contrary to AD, BD does not require any instrument tape. The BD method also delivers excellent Greeks performance (e.g., a 50 times acceleration compared to the bump-and-reprice method for $\sim$100 parameters).

The majority of XVAs are computed on portfolios with general collateral arrangements and consisting of bespoke instruments requiring the application of differentiation or regression functions. This is covered in the present article, where we generalize the algorithmic differentiation methods for exotic instruments [3] from pricing to XVA calculations. Note that a similar theme was recently treated in Capriotti et al. [6], in which the authors applied the standard AD for XVA to a Bermudan instrument. Our approach generalizes these results to the case of general collateral with an arbitrary portfolio leveraging the algorithmic exposure method [2]. Along with AD, we also elaborate upon the BD method and how to apply it to XVA.

We operate with a portfolio exposure that is calculated using a sum of underlying instrument future values (FVs). We consider calculating the portfolio exposure (or the instruments FVs) on a rather dense set of future ("observation") dates, e.g., monthly dates for the next 10–20 years. This information is then aggregated into different XVAs.

When considering the calculations, it is important to remember the main difference between FVs and CVs: the FV accounts for exercises before the date on which it is observed, while the CV, being an option hold value, does not. For example, the Bermudan swaption FV is equal to the swaption CV if the exercise did not take place before the observation date and is equal to the swap CV if it *was* exercised before the observation date. Note that for non-callable instruments the FV naturally coincides with the CV because there are no exercises. For general instruments with possible exercises, it appears that in order to calculate the FVs on all observation dates, one should store all the CVs on these dates during the backward pricing. Then, in the forward pass, one would find the FV by analysing the exercise and choosing the swaption CV or the swap CV. This procedure, however, becomes instrument dependent.

To keep the algorithm uniform with respect to instrument type, we take advantage of the "algorithmic exposure" procedure introduced in [2] that allows the calculation of the FVs during the backward pricing step. By operating on the CVs used in pricing, the algorithmic exposure procedure induces a set of rules for the FVs. The most important feature is that regression is not applied to the FVs, which makes its algorithmic differentiation intuitive and simple. Thus, we can combine both AD and BD with the algorithmic exposure calculations in order to obtain the XVA derivatives.

This article is organized as follows. In Section 2, we use a simple forward instrument example to provide an overview of the properties of AD and demonstrate why it is much faster than the bump-and-reprice approach. In Section 3, we consider a general instrument that requires backward Monte Carlo pricing by regression and provide an overview of how to compute its present and future values (as well as XVA), using the algorithmic exposure method for the latter. In Section 4 we analyze, when computing Greeks, whether it is necessary to differentiate conditional expectations. We then classify

the corresponding cases. In Section 5, we collect key formulas for conditional expectation derivatives. In Section 6, we review the standard AD method for XVA with a focus on estimating memory usage for the various tapes involved, leaving the detailed analysis to Appendix A. In Section 7, we introduce backward differentiation for XVA. In Section 8, we compare backward bifferentiation with the standard adjoint differentiation for XVA. In Section 9, we provide the results of our numerical experiments for backward differentiation applied to the computation of XVA Greeks. This includes speed benchmarks. We conclude in Section 10. Appendix A gives a detailed overview of the standard AD method for XVA Greeks. Appendix B lists formulas for the smoothed delta function used in this paper.

2 Adjoint differentiation for forward instruments

Let us introduce our notations and derive some preliminary formulas. We will study a general instrument's price dependence on the *model* parameters $\theta = \{\theta_1, \dots, \theta_{N_\theta}\}$, an N_θ-dimensional vector. Denote by $X(t) = \{X_1(t), \dots, X_{N_X}(t)\}$ an N_X-dimensional state vector of the model, which is considered to be a hybrid model with a potentially large number of components. When appropriate we will highlight the dependence of the state on the model parameters by writing $X(t;\theta)$.

We define a general Markov discretization scheme on a set of N_t simulation dates $\{t_i\}_{i=1}^{N_t}$ by

$$X(t_{i+1}) = G(t_i, X(t_i), Z_i, \theta), \tag{1}$$

where Z_i is a Gaussian random variable whose origin is at t_0.

Using a simple example, we will explain the high speed of the AD method. Let us consider a general path-dependent discounted payoff $Q(X) = Q(X(t_1), \dots, X(t_{N_t}))$ and the corresponding value $V(\theta) = \mathbb{E}\left[Q(X(\theta))\right]$ for a given model expectation $\mathbb{E}$. When computing Greeks, the goal is to calculate the price derivative over all model parameters θ_n:

$$\frac{\partial V(\theta)}{\partial \theta} = \mathbb{E}\left[\frac{\partial Q(X)}{\partial \theta}\right],$$

where the payoff derivative can be expressed via the chain rule:

$$\frac{\partial Q(X)}{\partial \theta} = \sum_{k=1}^{N_t} \frac{\partial Q(X)}{\partial X(t_k)} \frac{\partial X(t_k)}{\partial \theta}.$$

In practice, we calculate the value V using MC simulations, i.e.,

$$V(\theta) \simeq \frac{1}{N_p} \sum_{p=1}^{N_p} Q(X(\theta)[p]),$$

where N_p is the number of MC paths and $X(t)[p]$ is the state realization of the pth path. Below, we calculate the underlying derivatives $\partial X(t_k)[p]/\partial\theta$ path-by-path. For notational brevity, we will omit the path index.

The AD calculation is based on the *backward* substitution of the simulation step in (1) into the payoff:

$$\begin{aligned} Q(X(t_1), \dots, X(t_{N_t})) &= Q(X(t_1), \dots, X(t_{N_{t-1}}), X(t_{N_t}, G(t_{N_t-1}, X(t_{N_t-1}), Z_{N_t-1}, \theta)) \\ &= \cdots \\ &= Q_i(X(t_1), \dots, X(t_i); Z_i, \dots, Z_{N_t-1}; \theta), \end{aligned} \tag{2}$$

where we have denoted by Q_i the payoff value for which we have backwardly substituted the states until its latest direct dependence on $X(t_i)$. In other words, we have calculated the payoff *conditional* on the time t_i, substituting all $X(t_j)$ from the very end ($j = t_{N_t}$) back to $j = i + 1$. Thus, we have obtained explicit dependence on the parameters thanks to the substitution, while some dependence upon θ can still be hidden in $X(t_i)$. We will use the notation

$$Q_i(X(t_1), \dots, X(t_i); Z_i, \dots, Z_{N_t-1}; \theta) = Q(X(t_1), \dots, X(t_{N_t}))\big|_{t_i} \tag{3}$$

to highlight this. Our goal is to find the payoff dependence on the states for which we have substituted all of the θ dependencies, that is,

$$Q_0(Z_1, \dots, Z_{N_t-1}; \theta) = Q(X(t_1), \dots, X(t_{N_t}))\big|_{t_0}. \tag{4}$$

The recursive relationship between Q_i and Q_{i-1},

$$\begin{aligned} &Q_i(X(t_1),\dots,X(t_i);Z_i,\dots,Z_{N_t-1};\theta) \\ &\qquad = Q_i(X(t_1),\dots,X(t_{i-1}),G(t_{i-1},X(t_{i-1}),Z_{i-1},\theta);Z_i,\dots,Z_{N_t-1};\theta) \\ &\qquad = Q_{i-1}(X(t_1),\dots,X(t_{i-1});Z_{i-1},\dots,Z_{N_t-1};\theta), \end{aligned}$$

is the key to expressing the payoff derivative over the parameters in a *backward* manner. Keeping in mind that our goal is to find $\partial Q_0/\partial\theta$, we note that

$$\frac{\partial Q_{i-1}}{\partial\theta} = \frac{\partial Q_i}{\partial\theta} + \frac{\partial Q_i}{\partial X(t_i)}\frac{\partial G(t_{i-1},X(t_{i-1}),Z_{i-1},\theta)}{\partial\theta}, \tag{5}$$

and we prepare at the same time the payoff derivative over $X(t_{i-1})$:

$$\frac{\partial Q_{i-1}}{\partial X(t_{i-1})} = \frac{\partial Q_i}{\partial X(t_{i-1})} + \frac{\partial Q_i}{\partial X(t_i)}\frac{\partial G(t_{i-1},X(t_{i-1}),Z_{i-1},\theta)}{\partial X(t_{i-1})}. \tag{6}$$

Note that the first derivative on the right-hand side of (6) is a partial derivative of the payoff over its explicit dependence on $X(t_{i-1})$, while the second term is responsible for the *substituted* derivative over $X(t_{i-1})$ coming from the later time steps.

For the AD we need the following derivatives of the discretization function:

$$\frac{\partial G(t_i,X(t_i),Z_i,\theta)}{\partial\theta}, \tag{7}$$

which is an $N_\theta \times N_X$ matrix per path per time step, and

$$\frac{\partial G(t_i,X(t_i),Z_i,\theta)}{\partial X(t_i)}, \tag{8}$$

which is an $N_X \times N_X$ matrix per path per time step. Here, N_θ is the number of parameters and N_X is the number of model states. In addition, the scheme defined in (5) and (6) requires the partial derivatives of the payoff,

$$\frac{\partial Q_i}{\partial X(t_{i-1})}, \tag{9}$$

which is a vector of length N_X per path per time step.

For any fixed payoff Q, AD requires backward propagation. Having calculated and stored the information in (7)–(9), we go backward in time, calculating the payoff's derivatives over the states and over the parameters. The derivative calculations $\partial Q_i/\partial X(t_i)$ from (6) cost N_X^2 multiplications per path and per timestep.

In practice, the parameter vector θ can be very large, particularly if there are time-varying parameters, as there almost certainly will be in the case of interest rate products since the initial yield curve is often heavily parameterized. However, in the case of time-dependent parameters, the discretization function $G(t_i,X(t_i),Z_i,\theta)$ often depends only on a limited number of parameters (corresponding to the timestep $[t_i,\, t_{i+1}]$) rather than on the whole set of parameters. Consequently, the number of multiplications in (5) will be $O(1)N_X$ per path and per timestep due to the explained sparseness of the matrix $\partial G/\partial\theta$. Thus, the overall cost is dominated by the calculation of derivatives over the states in (6) and is reduced to

$$N_X^2 \times N_p \times N_t \tag{10}$$

multiplications. Note that this timing is for our fixed payoff. A new payoff Greeks treatment will require the whole procedure to be repeated; thus, the time is proportional to the number of payoffs (outputs).

Let us stress again that the cost of the adjoint procedure is independent of the number of parameters due to the sparse structure of $\partial G/\partial\theta$ for time-dependent parameters. If this were not the case, i.e., if all the parameters were global, the total derivatives calculation time could be as big as $\sim N_\theta \times N_X \times N_p \times N_t$, i.e., proportional to the number of parameters.

Below, we assume that all of our parameters are time dependent.

We also note that the AD procedure should be repeated for each new payoff.

To conclude this simple analysis on the differentiation speed, we will mention that other factors can also influence the performance of AD with respect to the bump-and-reprice method. The most important of these factors is the speed of access to the memory, which directly influences the speed of writing and reading the tape. Another important factor is the speed of calculating the discretization function G

and its derivatives. If the function G is "heavy",[1] its derivatives are often fast to calculate using the information about the function itself. In this case, the pathwise derivatives methods will be preferable over the bump-and-reprice method. However, heavy derivatives can sometimes still be an important slowdown factor in pathwise methods.

2.1 Adjoint differentiation structure for general forward instruments

We have seen in the simple example above that AD has the following steps:

- *Simulation tape preparation*
  Simulate the model states and write the required information from (7) and (8). We will refer to this step as "preparing" a *simulation tape*. It can be done either path-by-path or for all paths.
- *Instrument tape preparation*
  In our example the instrument tape was simple: it contained only derivatives of the payoff over the states as per (9). However, for general forward instruments one should store the derivatives of the payments over the states as well as the derivatives over the parameters (if present) during the forward pricing, i.e., $\partial Q/\partial X(t_i)$ per path per time step and $\partial Q/\partial\theta$ per path per time step. An example of the direct dependence of a payoff on a model parameter is given by a Libor payment in a short-rate model: a Libor is a function of the states and the parameters.
- *Adjoint calculation of the instrument derivatives*
  In our example, the calculation was backward (i.e., in the opposite direction of the forward pricing) and path-by-path, see (5) and (6). The explicit derivatives of the payoff over the parameters results in the "initial" condition
  $$\frac{\partial Q_{N_t}}{\partial\theta} = \frac{\partial Q}{\partial\theta}$$
  for (5) with $i = N_t$.
  Going forward, we will refer to this iterative backward procedure
  $$\left\{\frac{\partial Q}{\partial X(t_i)}, \frac{\partial Q}{\partial\theta_n}\right\} \underset{\text{AD}}{\longrightarrow} \frac{\partial Q}{\partial\theta_n} \tag{11}$$
  as the playback of the simulation tape or as classic AD. Its cost is dominated by the number of multiplications in (10).

This concludes our discussion of forward instruments. However, as we discuss the application of AD to backward deals, keep in mind that we will try to obtain the same advantages for backward instruments that we have seen above for forward instruments, in spite of the regression path-mixing that arises for backward deals.

3 General instruments and backward algorithms

In this section we briefly explain backward instrument pricing; see [2] for more details on general scripting instruments. The main component of the backward pricing procedure is a *continuation value* (CV) at time t expressed in some currency units and denoted $V(t)$. The CV is a certain function of the model state variables at time t. Financially, $V(t)$ is a *hold* value: it is the price of holding an option at time t with all possible payments and exercises taking place thereafter. For coherence, we express all of the CVs in domestic currency units. A simple CV example is a leg with a single unit payment at time T. More complicated examples include a swap value or a swaption holding value.

Consider an example of a *Bermudan swaption* giving the right to enter into a swap on exercise dates T_j. Suppose the underlying swap pays a *generalized cashflow*

$$c_j \quad \text{at date} \quad \tau_j \quad \text{for} \quad j = 1, \dots, M. \tag{12}$$

For example, c_j may be a fixed payment $c_j =$ `notional*dcf*`K with a fixed rate K, or it may be a floating payment. We consider that such cashflows are known as analytical functions of the model states and parameters. Indeed, the majority of models permits it.

For simplicity, we will suppose for the remainder of this paper that the payment dates coincide with the exercise dates; that is, $\tau_j = T_j$.

[1] I.e., the function G is not given in closed form and instead must be found using a numerical procedure.

Let $V(t)$ and $S(t)$ denote the CVs of the swaption and the swap, respectively. We will refer to the union of exercise and payment dates as instrument dates. The swaption's backward pricing is performed using two repeated steps.

The first step is to calculate the following **transformations** at the specified instrument dates:

$$V(T_j^-) = \max(V(T_j), S(T_j)), \tag{13}$$

$$S(T_j^-) = S(T_j) + c_j. \tag{14}$$

If the cashflow is set before the payment date (like a Libor payment), it should be added to the swap leg on a fixing date being multiplied by a zero coupon bond maturing at the payment date. Here, for simplicity, we suppose that the fixing and payment dates coincide.

Here and below we write T^- to denote the time just before time T. By avoiding special notations for the CVs on the left-hand side, this convention serves to simplify the writing of the exercise and payment conditions. Otherwise, we can write such updates in the *coding style* using the operator "+=", so that (14) is equivalent to

$$S(T_j) \mathrel{+}= c_j.$$

In general, the transformations above (payments and exercises) reduce to a linear combination

$$V_n(T^-) = \sum_m \gamma_{m,n} V_m(T), \tag{15}$$

with *dimensionless* coefficients $\gamma_{m,k}$ and with the CVs bearing the currency dimension. For example, for the maximum we have

$$V(T^-) = 1_{V(T)>S(T)} V(T) + 1_{V(T)\le S(T)} S(T), \tag{16}$$

where the exercise indicators are dimensionless coefficients. These linear combinations are used further in the context of the XVA construction.

The second step is a **propagation**, i.e., the calculation of the discounted conditional expectation in the model measure between the instrument dates of the CVs at hand:

$$S(t) = N(t)\mathbb{E}\left[\frac{S(T^-)}{N(T)}\middle|\mathcal{F}_t\right] \quad \text{and} \quad V(t) = N(t)\mathbb{E}\left[\frac{V(T^-)}{N(T)}\middle|\mathcal{F}_t\right], \tag{17}$$

where $N(t)$ is the model numeraire. The numeraire can be a deterministic function of the states at the same time (e.g., in the case of a zero bond corresponding to a forward measure, $N(t) = N(t, X(t))$) or a path-dependent function (e.g., in the case of a savings account under the risk-neutral measure).

The swaption price V coincides with the corresponding CV at the origin, i.e., $V \equiv V(0)$. A general instrument contains a set of CVs V_j that are transformed using arithmetical operations and other functions (like max, step, etc.) of the instrument dates as well as the conditional expectations between the instrument dates.

Let us now move to the XVA calculation, which is more complicated than the PV. We denote a *future price* at time t with small letters $v_m(t)$ as oppossed to the capital letter notation of the continuation values $V_m(t)$. The main difference between these two quantities is that the CVs do not take into account the eventual exercises before the time t, while the FVs do. Below we will examine these values using several examples.

The sum of future values across a portfolio

$$u(t) = \sum_m v_m(t)$$

is the main value of interest in the XVAs. For example, a CVA for a single-side CSA is

$$\text{CVA} = \int_0^T ds \mathbb{E}\left[\frac{u(s)^+}{N(s)}\right] \lambda(s) e^{-\int_0^s dt \lambda(t)},$$

where $\lambda(t)$ is the credit spread of a counterparty and T is the maturity of the portfolio. To calculate the integral above numerically, we discretize it on a set of *observation dates*

$$\{s_i\} \quad \text{for} \quad i = 1, \ldots, N_{obs}$$

such that a general collateral XVA is a sum over the dates and MC paths

$$\text{XVA} = \sum_i \mathbb{E}\left[\frac{F_i(u(s_1), \ldots, u(s_i))}{N(s_i)}\right] w_i, \tag{18}$$

where the deterministic function $F_i(u_1, \cdots, u_i)$ is a generalized exposure and the w_i are certain deterministic weights. Note that the exposure F_i potentially depends on all the future values up to time s_i, which reflects a general collateral structure.

Let us illustrate now the main difference between the FVs and CVs. The continuation value $V(t)$, being an option *hold* value, does not take into account the eventual exercises before the time t, while the FV does. For example, the Bermudan swaption future value is equal to the swaption CV if the exercise did not take place before t and is equal to the swap CV if the exercise *did* take place before t:

$$v(t) = V(t)(1 - \mathcal{I}(t)) + S(t)\mathcal{I}(t). \tag{19}$$

Here the global exercise indicator $\mathcal{I}_j$ is a path-dependent quantity composed from local exercise indicators

$$\mathcal{C}_j = 1_{S(T_j)>V(T_j)}, \tag{20}$$

which take into account the CVs only for the given exercise date. Its relation

$$\mathcal{I}_j = 1 - \prod_{i=1}^{j} (1 - \mathcal{C}_i) \tag{21}$$

means that the first in time local exercise sets the global exercise to one. Note that the swap FV coincides with its CV because the swap does not have exercises.

Another view on the FVs is related with the cashflows in (12). Using our swaption example it is easy to see that its PV is built with discounted cashflows subjected to the exercise conditions in (21):

$$V = \mathbb{E}\left[\sum_{j=1}^{M} \frac{\mathcal{I}(T_j)c_j}{N(T_j)}\right], \tag{22}$$

where the global indicator $\mathcal{I}(t) = \mathcal{I}_j$ for $t \in [T_j,\, T_{j+1})$. The swaption FV for a given date t is a *conditional* expectation of all payments on and after t:

$$v(t) = N(t)\mathbb{E}\left[\sum_{j=1, T_j \geq t}^{M} \frac{\mathcal{I}(T_j)c_j}{N(T_j)} \middle| \mathcal{F}_t\right]. \tag{23}$$

One can show (see, e.g., [2]) that this expression is equivalent to (19).

Equation (19) indicates that to calculate the FVs on all observation dates one should store all the CVs on these dates during the backward pricing and then, in the forward pass, analyse the exercise and choose either the swaption CV or the swap CV. However, as we have shown in [2], it is actually possible to calculate the FVs at the same time as the backward pricing. We refer to this procedure as the algorithmic exposure calculation. Its simple rules algorithmically modify the so-called observation value (OV) that is denoted by $v(t, s)$, where t is the current time and s is the observation date.

To begin, we add extra entities (the OVs corresponding to the observation dates) to each CV:

$$V_m(t) \to \{V_m(t); v_m(t; s_1), \ldots, v_m(t; s_{N_{obs}})\}.$$

We proceed backward by following the pricing script instructions. Once at the origin, the OV gives the FV. Note the difference between the product PV (a number) and the stochastic variable FV (a path-wise number).

Below, we describe these algorithmic rules in more detail. Once the backward script reaches an observation date s_i, the OV is initialized by the corresponding CV:

$$v_m(s_i, s_i) = V_m(s_i). \tag{24}$$

Before reaching that date in the procedure, the OV is zero ($v_m(t, s) = 0$ for $t > s$) and does not participate in the algorithmic rules.

Now, the backward-propagation rules (15) and (17) algorithmically induce the following concise rules for the OVs for $T < s_i$:

- The linear update rules for OVs are the same as for CVs:

$$V_k(T^-) = \sum_m \gamma_{m,k} V_m(T) \quad \Rightarrow \quad v_k(T^-, s_i) = \sum_m \gamma_{m,k} v_m(T, s_i), \tag{25}$$

  where $\gamma_{m,k}$ are generalized market rates (for example, numbers or barrier exercise indicators).

- There are no changes to the OVs during the backward propagation of the CVs:

$$V_n(t) = N(t)\mathbb{E}\left[\left.\frac{V_n(T^-)}{N(T)}\right|\mathcal{F}_t\right] \quad \Rightarrow \quad v_n(t) = v_n(T^-). \tag{26}$$

This way, the future value is finalized once the backward script reaches the origin; in particular, we have $v_m(s_i) = v_m(0, s_i)$.

The procedure of the future value calculation is purely backward and automatic: it avoids the complications of a manual method that are caused by cumbersome exercise aggregation.

As a simple illustration, consider the CV of a unit payment in some currency at time T that we add to a leg at date t. When the backward pricing running time reaches the date t, we create a CV corresponding to the zero bond $V_T(t) = P(t, T)$. This algorithmically creates the OVs attached to the zero bond CVs:

$$v_T(t, s_i) = P(s_i, T),$$

for the observation dates $s_i \in [t, T]$. Adding this payment to a leg V at time t will induce the following OV transformation:

$$V(t^-) = V(t) + P(t, T) \quad \Rightarrow \quad v(t^-, s_i) = v(t, s_i) + P(s_i, T).$$

The other OVs (i.e., those corresponding to observation dates later than T) are unchanged. Thus, adding a payment to a leg has a *local* influence on its OVs.

An example of a *global* influence to the OVs is provided by the maximum (or minimum) operation. Namely, the maximum operation (16) will update *all* the OVs that have an observation date after time T:

$$V(T^-) = 1_{V(T)>S(T)}V(T) + 1_{V(T)\leq S(T)}S(T)$$
$$\Downarrow$$
$$v(T^-, s_i) = 1_{V(T)>S(T)}v(T, s_i) + 1_{V(T)\leq S(T)}v_S(T, s_i)$$

for $s_i \geq T$. Here we have denoted the swap OVs by v_S. We note that the indicators will depend on the CVs at time T.

We conclude this section with a comment on the dates. In general, the observation dates are quite numerous; it is common to have around 100–200 dates. Quite often, these are close to the simulation discretization dates.

4 PV and XVA Greeks: should we differentiate conditional expectations?

We move now to Greeks calculations for general instruments. From the first sight, we should always compute derivatives of continuation values inside the backward pricing procedure. As we mentioned in the introduction, such a differentiation is a time-consuming and complicated operation; people often avoid it by instead calculating derivatives of cashflows [1].

In this section we will systematically address the cases when CV differentiation is necessary and when it can be avoided. Here we do not appeal to numerical methods but treat the problem for general conditional expectations and differentiations.

The model states $X(t)$ depend on parameters θ through the iteration (1). Following the substitution rules and notations (2)–(4) we can write

$$X(T)|_t = X(T; X(t), \theta, \{Z_i\}_{t<t_i<T}),$$

where we have expressed $X(T)$ through the driving factors Z_i between times t and T, the value of the state at time t, and the induced parameter dependence. Sometimes we will omit the driving factors argument, writing simply $X(T)|_t = X(T; X(t), \theta)$.

For a certain CV function $V(T) = V(T, X(T), \theta)$ explicitly dependent on the states and parameters, we can do the same as above:

$$V(T)|_t = V\left(T; X(T)|_t, \theta\right) = V\left(T; X(T; X(t), \theta), \theta\right).$$

A *full* dependence on the parameters is attained when we substitute the state back to the origin:

$$\left.\frac{\partial V(T)}{\partial \theta}\right|_{t_0} = \frac{\partial V(T)}{\partial \theta} + \frac{\partial V(T)}{\partial X(T)}\frac{\partial X(T; X(0), \theta, \{Z_i\}_{t_i<T})}{\partial \theta}.$$

Below, we will need the following identities

$$\begin{aligned}\frac{\partial}{\partial\theta}\mathbb{E}\left[V(T)|_{t_0}\right] &= \mathbb{E}\left[\left.\frac{\partial V(T)}{\partial\theta}\right|_{t_0}\right],\\ \frac{\partial}{\partial\theta}\mathbb{E}\left[V(T)|_{t_0}\,\middle|\,\mathcal{F}_t\right] &= \mathbb{E}\left[\left.\frac{\partial V(T)}{\partial\theta}\right|_{t_0}\,\middle|\,\mathcal{F}_t\right].\end{aligned} \tag{27}$$

The derivatives over the parameters are *full* ones, i.e., we have substituted all the states and their parameter dependence to obtain $V(T)|_{t_0} = V(T; X(0), \theta, \{Z_i\}_{t_i<T})$. Thus, the unconditional (t-conditional) average is done over $\{Z_i\}_{t_i<T}$ ($\{Z_i\}_{t<t_i<T}$, respectively) and can be interchanged with the derivatives due to the independence of Z_i from the parameters.

Now we return to our instrument cashflows c_k. Suppose that the cashflow dependence on the parameters and the states is known analytically, that is,

$$c_k = c_k(X(T_k), \theta),$$

and that we can calculate their full derivatives

$$\left.\frac{\partial c(T_k)}{\partial\theta}\right|_{t_0}$$

over the parameters. Below we will slightly abuse notations and write the full derivative of the cashflows as

$$\frac{\partial c(T_k)}{\partial\theta} \equiv \left.\frac{\partial c(T_k)}{\partial\theta}\right|_{t_0}.$$

4.1 PV analysis

Let us consider the pricing (PV) case starting with the swaption example. The exercise indicators form a surface in the state space for each exercise date. This surface is *optimal* for the Bermudan swaption price (22):

$$V = \mathbb{E}\left[\sum_{j=1}^{M}\mathcal{I}_{j-1}\frac{c_j}{N(T_j)}\right], \tag{28}$$

i.e., "bumping" the indicators as functions of parameters leads to zero:

$$\mathbb{E}\left[\sum_{j=1}^{M}\frac{\partial\mathcal{I}_{j-1}}{\partial\theta}\frac{c_j}{N(T_j)}\right] = 0.$$

This immediately gives us the PV Greek

$$\frac{\partial}{\partial\theta}V = \mathbb{E}\left[\sum_{j=1}^{M}\mathcal{I}(T_j)\frac{\partial}{\partial\theta}\left(\frac{c_j}{N(T_j)}\right)\right], \tag{29}$$

which requires only the differentiation of the cashflows and the numeraire.

How can we understand this optionality on the pricing script level? Actually, it is directly related to the "max" function in the transformations (13)–(14). To see this from the backward algorithm perspective, one should first replace the maximum update (13) with the equivalent one (16) on the instrument dates T_k:

$$\begin{aligned}V(T_k^-) &= (1-\mathcal{C}_k)V(T_k) + \mathcal{C}_k S(T_k),\\ S(T_k^-) &= S(T_k) + c_k.\end{aligned}$$

Denote the swaption without the swap as $U(t) \equiv V(t) - S(t)$. Then, restoring the discounting from the above, we have

$$\frac{U(T_k^-)}{N(T_k)} = (1-\mathcal{C}_k)\frac{U(T_k)}{N(T_k)} - \frac{c_k}{N(T_k)}$$

with a discounting conditional expectation between the instrument dates:

$$\frac{U(T_{k-1})}{N(T_{k-1})} = \mathbb{E}\left[\left.\frac{U(T_k^-)}{N(T_k)}\right|\mathcal{V}_{T_{k-1}}\right].$$

Denote the difference between two discounted CVs as

$$\Delta(T_{k-1}) = \frac{U(T_k^-)}{N(T_k)} - \frac{U(T_{k-1})}{N(T_{k-1})},$$

which satisfies

$$\mathbb{E}\left[\Delta(T_{k-1}) \,\middle|\, \mathcal{F}_{T_{k-1}}\right] = 0. \tag{30}$$

Then, we can form the induction

$$\frac{U(T_{k-1})}{N(T_{k-1})} = (1-\mathcal{C}_k)\frac{U(T_k)}{N(T_k)} - \frac{c_k}{N(T_k)} - \Delta(T_{k-1}). \tag{31}$$

Initializing it at the swap maturity $k = K$ by zero we readily obtain

$$\sum_{j=1}^{M}(\mathcal{I}_{j-1}-1)\left(\frac{c_j}{N(T_j)} + \Delta(T_{j-1})\right)$$

at the origin. Its average gives the PV

$$U = \mathbb{E}\left[\sum_{j=1}^{M}(\mathcal{I}_{j-1}-1)\left(\frac{c_j}{N(T_j)} + \Delta(T_{j-1})\right)\right],$$

which can be obviously presented without the conditional average defect Δ in the spirit of (28) thanks to (30):

$$U = \mathbb{E}\left[\sum_{j=1}^{M}(\mathcal{I}_{j-1}-1)\frac{c_j}{N(T_j)}\right].$$

This derivation shows how the present conditional expectation seemingly disappears from the final price. Another intuitive explanation is based on the fact that given the exercise indicators, the instrument transforms to the forward one, which can be solved by calculating unconditional averages.

Let us pass now to the Greeks. Differentiating the induction (31) over a parameter, we obtain

$$\frac{\partial}{\partial\theta}\frac{U(T_{k-1})}{N(T_{k-1})} = (1-\mathcal{C}_k)\frac{\partial}{\partial\theta}\frac{U(T_k)}{N(T_k)} - \frac{\partial}{\partial\theta}\frac{c_k}{N(T_k)} - \frac{\partial}{\partial\theta}\Delta(T_{k-1}). \tag{32}$$

The derivative over the indicators proportional to the delta-function $\delta(U(T_k))$ cancels out with $U(T_k)$.[2] This is the reason why one should not differentiate over the optimal indicators. Thus, the induction (32) gives us

$$\frac{\partial}{\partial\theta}U = \mathbb{E}\left[\sum_{j=1}^{M}(\mathcal{I}_{j-1}-1)\frac{\partial}{\partial\theta}\left(\frac{c_j}{N(T_j)} + \Delta(T_{j-1})\right)\right] = \mathbb{E}\left[\sum_{j=1}^{M}(\mathcal{I}_{j-1}-1)\frac{\partial}{\partial\theta}\left(\frac{c_j}{N(T_j}\right)\right],$$

which is equivalent to (29). The second equation is a consequence of $\mathbb{E}\left[\partial\Delta(T_{k-1})/\partial\theta \,\middle|\, \mathcal{F}_{T_{k-1}}\right] = 0$ due to (27).

Note that if the exercise condition is not optimal, e.g., if

$$V(T_j^-) = (1-\mathcal{B}_j)V(T_j) + \mathcal{B}_j S(T_j)$$

with an indicator $\mathcal{B}_j$ related with some barrier condition on the model states, its derivatives $\partial\mathcal{B}/\partial\theta$ will not disappear; instead, they will modify the derivatives $\partial V/\partial\theta$ updated with respect to the values and will require differentiation of the CVs.

To conclude with th PV Greeks, we can state that the optimal exercise instruments will not depend on the exercise derivatives but general exercise instruments (e.g., barriers) will. Having said that, the only non-trivial instruments that will require the differentiation of the exercises in the backward pricing are complicated options with two exercises: optimal and barrier (e.g., TARNs); simple barrier instruments do not require backward pricing.

If our pricing script does not contain "step" functions leading to non-trivial dependence on the parameters, one can suggest the following algorithm that avoids the differentiation of conditional expectations:

[2] One can obtain this directly by differentiating the maximum function from the very beginning:

$$\frac{\partial\max(V,S)}{\partial\theta} = (1-\mathcal{C}\frac{\partial V}{\partial\theta}) + \mathcal{C}\frac{\partial S}{\partial\theta}.$$

- Associate with any CV $V(t)$ its "derivative partner" $\partial\tilde{V}(t)/\partial\theta$.[3]
- When the CV is modified, the derivative partners will be modified according to the differentiation rules, e.g.,

$$V(T_k^-) = \max(V(T_k), S(T_k)) \qquad \Rightarrow \qquad \frac{\partial\tilde{V}(T_j^-)}{\partial\theta} = (1-\mathcal{C}_j)\frac{\partial\tilde{V}(T_j)}{\partial\theta} + \mathcal{C}_j\frac{\partial\tilde{S}(T_j)}{\partial\theta},$$
$$S(T_k^-) = S(T_k) + c_k \qquad \Rightarrow \qquad \frac{\partial\tilde{S}(T_k^-)}{\partial\theta} = \frac{\partial\tilde{S}(T_k)}{\partial\theta} + \frac{\partial c_k}{\partial\theta},$$

  where $\partial c_k/\partial\theta$ is a real derivative of the analytically available cashflows and the rest, i.e., $\partial\tilde{V}(T_j^-)/\partial\theta$, are just notations.
- Conditional expectations for the CVs (17) induce a simple discounting for derivative partners, i.e.,

$$\frac{\partial\tilde{V}(t^-)}{\partial\theta} = \frac{\partial\tilde{V}(T^-)}{\partial\theta}\frac{N(t)}{N(T)} + V(T^-)\frac{\partial}{\partial\theta}\left(\frac{N(t)}{N(T)}\right). \tag{33}$$

This algorithm is closely related to the BD algorithm, see Section 7.

4.2 XVA analysis

Consider a simple valuation adjustment related to our counterparty default with a deterministic hazard rate $\lambda(t)$ and a deterministic collateral $C(t)$. We suppose that the only element of our portfolio is a Bermudan swaption. Thus, if we have exercised into a swap, a loss on the interval $[t, t+dt]$ related with the counterparty default will be

$$1_{\tau\in[t,t+dt]}(S(t)-C(t))^+,$$

while if we did not exercise before t, the loss will be

$$1_{\tau\in[t,t+dt]}(V(t)-C(t))^+,$$

where τ is a default time that we assume is uncorrelated with the model driving factors and where

$$\mathbb{E}\left[1_{\tau>t}\right] = e^{-\int_0^t ds\lambda(s)}.$$

Also, we have ignored nonlinearities and consider the close-out convention based on default-free instruments.

We denote the new exercise indicator as $\hat{\mathcal{I}}(t)$. This is formally different than the default-free indicator $\mathcal{I}(t)$. The total instrument price will be

$$\hat{V} = \mathbb{E}\left[\sum_{j=1}^{M}\hat{\mathcal{I}}_{j-1}\frac{c_j}{N(T_j)}\right] - \mathbb{E}\left[\int_0^T d\tilde{\lambda}(t)\frac{\hat{\mathcal{I}}(t)(S(t)-C(t))^+ + (1-\hat{\mathcal{I}}(t))(V(t)-C(t))^+}{N(t)}\right], \tag{34}$$

where we used $\mathbb{E}\left[1_{\tau\in[t,t+dt]}\right] = dt\tilde{\lambda}(t)$ for $\tilde{\lambda}(t) = \lambda(t)e^{-\int_0^t ds\lambda(s)}$. The last term in (34) is traditionally called a Credit Valuation Adjustment (CVA).

Alternatively, the total price can be obtained as a solution of the following transformation of new, default-adjusted CVs that are denoted with hats:

$$\hat{V}(T_k^-) = \max(\hat{V}(T_k), \hat{S}(T_k)), \tag{35}$$
$$\hat{S}(T_k^-) = \hat{S}(T_k) + c_k,$$

with the backward propagation taking into account the default losses that occurred on the interval $[T_{k-1}, T_k]$:

$$\hat{V}(T_{k-1}) = N(T_{k-1})\mathbb{E}\left[\frac{\hat{V}(T_k^-)}{N(T_k)}\middle|\mathcal{F}_{T_{k-1}}\right] - N(T_{k-1})\mathbb{E}\left[\int_{T_{k-1}}^{T_k} d\tilde{\lambda}(t)\frac{(V(t)-C(t))^+}{N(t)}\middle|\mathcal{F}_{T_{k-1}}\right],$$
$$\hat{S}(T_{k-1}) = N(T_{k-1})\mathbb{E}\left[\frac{\hat{S}(T_k^-)}{N(T_k)}\middle|\mathcal{F}_{T_{k-1}}\right] - N(T_{k-1})\mathbb{E}\left[\int_{T_{k-1}}^{T_k} d\tilde{\lambda}(t)\frac{(S(t)-C(t))^+}{N(t)}\middle|\mathcal{F}_{T_{k-1}}\right].$$

[3] We put a tilde over the CV because, strictly speaking, it is not a derivative over the CV.

Again, thanks to the presence of the maximum in the total CV logic (35) or, in other words, the total price optimality, the total price Greek should ignore the indicator derivatives:

$$\frac{\partial}{\partial\theta}\hat{V} = \mathbb{E}\left[\sum_{j=1}^{M}\hat{\mathcal{I}}_{j-1}\frac{\partial}{\partial\theta}\frac{c_j}{N(T_j)}\right] \\ - \mathbb{E}\left[\int_0^T dt\tilde{\lambda}(t)\left(\hat{\mathcal{I}}(t)\frac{\partial}{\partial\theta}\frac{(S(t)-C(t))^+}{N(t)} + (1-\hat{\mathcal{I}}(t))\frac{\partial}{\partial\theta}\frac{(V(t)-C(t))^+}{N(t)}\right)\right].$$

So, is it possible to calculate $\partial\hat{V}/\partial\theta$ using only the information about the derivatives of cashflows and all CV values? Take the most complicated part

$$\mathbb{E}\left[(1-\hat{\mathcal{I}}(t))\frac{\partial}{\partial\theta}\frac{(V(t)-C(t))^+}{N(t)}\right] = \mathbb{E}\left[(1-\hat{\mathcal{I}}(t))\frac{1_{V(t)>C(t)}}{N(t)}\frac{\partial V(t)}{\partial\theta}\right] + \cdots,$$

where $\cdots$ indicates a term containing the derivative of the numeraire, which can be found easily. From the previous section we know that the swaption price derivative does not depend on the derivatives of the indicators (29). This is also true for conditional averages

$$\frac{\partial}{\partial\theta}V(t) = \mathbb{E}\left[\sum_{j=1,t<T_j}^{M}\mathcal{I}(t,T_j)\frac{\partial}{\partial\theta}\left(\frac{c_j}{N(T_j)}\right)\middle|\,\mathcal{F}_t\right],$$

where $\mathcal{I}(t,T_j)$ is an exercise indicator conditional on whether there were no exercises before time t. Consequently, the complicated derivative part reads

$$\mathbb{E}\left[(1-\hat{\mathcal{I}}(t))\frac{1_{V(t)>C(t)}}{N(t)}\frac{\partial V(t)}{\partial\theta}\right] \\ = \mathbb{E}\left[(1-\hat{\mathcal{I}}(t))\frac{1_{V(t)>C(t)}}{N(t)}\mathbb{E}\left[\sum_{j=1,t<T_j}^{M}\mathcal{I}(t,T_j)\frac{\partial}{\partial\theta}\left(\frac{c_j}{N(T_j)}\right)\middle|\,\mathcal{F}_t\right]\right] \\ = \mathbb{E}\left[(1-\hat{\mathcal{I}}(t))\frac{1_{V(t)>C(t)}}{N(t)}\sum_{j=1,t<T_j}^{M}\mathcal{I}(t,T_j)\frac{\partial}{\partial\theta}\left(\frac{c_j}{N(T_j)}\right)\right].$$

Thus, we can state that for a deterministic collateral the total price (PV minus CVA) calculated using the optimal exercise can be obtained using only cashflow derivatives.

Note that, separately, both the PV and CVA will depend on the exercise indicator derivatives, but they do not depend on the indicators when they are combined in the total price. Sometimes such a separation is necessary—different desks can hedge the PV and the CVA.

As explained in the previous section, if the portfolio contains barrier instruments with automatic (non-optimal) conditions, its total price Greeks cannot be calculated using the cashflow derivatives.

Moreover, the optimal exercise taking into account the whole portfolio's valuation adjustments for Bermudans is rarely available: for a portfolio containing multiple swaptions, such an optimization is a very hard task. That is why people use the "risk-free" exercises $\mathcal{I}(t)$ that deliver the standard formula for the CVA,

$$\text{CVA} = \mathbb{E}\left[\int_0^T dt\tilde{\lambda}(t)\frac{(v(t)-C(t))^+}{N(t)}\right], \tag{36}$$

effectively replacing $\hat{\mathcal{I}}(t)S(t) + (1-\hat{\mathcal{I}}(t))V(t)$ by the FV (19). In this case, the indicators are not optimal for the CVA; thus, it will depend on its derivatives and the cashflow derivatives alone will not be sufficient. If one ignores the delta functions coming out of the exercise differentiation, there can be a sizable mismatch between the AD and the bump-and-reprice of an XVA Greek; see Section 9 for further discussion.

Now let us consider a portfolio of vanilla instruments and establish conditions on the *collateral* when the cashflow derivatives can deliver the XVA Greeks. The portfolio can be considered as a generalized swap and, abusing notations, we denote its analytically available cashflows as c_j (paid at dates T_j). Denote the sum of discounted cashflows as

$$q(t) = N(t)\sum_{j=1,T_j\geq t}^{M}\frac{c_j}{N(T_j)}.$$

Then, the FV (coinciding with the CV) is simply the conditional expectation

$$v(t) = \mathbb{E}[q(t)|\mathcal{F}_t].$$

Let's start with a CVA containing simple averages[4] for some function $f(v)$ corresponding to the collateral as a deterministic function of the portfolio value, i.e., $f(v) = (v - C(v))^+$. Then, the average full derivative will be

$$\frac{\partial \mathbb{E}\left[f(v(t))\right]}{\partial \theta} = \mathbb{E}\left[\frac{\partial f(v(t))}{\partial \theta}\right] = \mathbb{E}\left[f'(v(t))\frac{\partial v(t)}{\partial \theta}\right]. \tag{37}$$

Applying a derivative to both sides of the equation (by the tower law)

$$\mathbb{E}\left[f'(v(t)(v(t) - q(t))\right] = 0,$$

we obtain

$$\frac{\partial}{\partial \theta}\mathbb{E}\left[f'(v(t)(v(t) - q(t))\right] = \mathbb{E}\left[f'(v(t)\frac{\partial}{\partial \theta}(v(t) - q(t))\right] + \underbrace{\mathbb{E}\left[\frac{\partial}{\partial \theta}f'(v(t)(v(t) - q(t))\right]}_{\text{zero by tower law}} = 0,$$

which leads to

$$\mathbb{E}\left[f'(v(t))\frac{\partial v(t)}{\partial \theta}\right] = \mathbb{E}\left[f'(v(t))\frac{\partial q(t)}{\partial \theta}\right].$$

Substituting into (37), we conclude that for a Greek for the single time average of a future price we need only the "cashflows" derivatives over the parameters

$$\frac{\partial \mathbb{E}\left[f(v(t))\right]}{\partial \theta} = \mathbb{E}\left[f'(v(t))\frac{\partial q(t)}{\partial \theta}\right].$$

This property was first presented in [1] for $f(x) = x^+$.

Now, consider a multi-time dependence of a general XVA (18)

$$\mathbb{E}\left[f(v(t_1), \ldots, v(t_n))\right]$$

corresponding to a general collateral $C(t_n) = C(t_n; v(t_1), \ldots, v(t_n))$ with non-zero minimum transfer amount and threshold $f = (v(t_n) - C(t_n; v(t_1), \ldots, v(t_n)))^+$. We have

$$\frac{\partial}{\partial \theta}\mathbb{E}\left[f(v(t_1), \ldots, v(t_n))\right] = \sum_{i=1}^{n}\mathbb{E}\left[\frac{\partial f(v(t_1), \ldots, v(t_n))}{\partial v(t_i)}\frac{\partial v(t_i)}{\partial \theta}\right].$$

One cannot replace directly $\partial v(t_i)/\partial\theta$ by $\partial q(t_i)/\partial\theta$ in the expression above because, in general,

$$\mathbb{E}\left[\frac{\partial f(v(t_1), \ldots, v(t_n))}{\partial v(t_i)}(v(t_i) - q(t_i))\right] \neq 0$$

for $i < n$. The tower law does not work here due to the dependence of $\partial f/\partial v(t_i)$ on Brownian increments *after* t_i. However, one can employ a *branching* diffusion as was first suggested in [1]. Namely, one branches the diffusion at time t_i with two independent Brownian motions $W^{(0)} = W$ and $W^{(1)}$, and the above inequality becomes an equality. Indeed, the first Brownian motion coincides with the initial one and goes into the cashflows q, while for the second simulated $\partial f(v^{(1)}(t_1), \ldots, v^{(1)}(t_n))/\partial v(t_i)$,[5]

$$\mathbb{E}\left[\frac{\partial f(v^{(1)}(t_1), \ldots, v^{(1)}(t_n))}{\partial v(t_i)}(v(t_i) - q(t_i))\right] = 0.$$

This trick was proposed in [1] for $f = (v(t_{n-1}) - v(t_n))^+$. The corresponding function on the branch (1) was calculated using a regression proxy. If we have a general path-dependence for the XVA, the branching logic becomes cumbersome (with n re-simulations for the timestep t_n), slow, and hard to control on the script level. Thus, for a general XVA with a general collateral, one should differentiate the conditional expectation.

Below we sum up the conditions when the cashflow derivatives are not sufficient to calculate XVA Greeks:

[4] We have removed the numeraire for simplicity.

[5] We use the superscript (1) to indicate the branched Brownian motion.

- One needs separate Greeks for the PV, CVA, DVA, etc. (E.g., when they are hedged by different desks.)
- The portfolio contains barrier options.
- The portfolio Bermudans have exercises calculated under the default-free assumptions: the exercises are optimal for the PV (this is a common practice due to numerical optimization and logistic difficulties).
- The collateral is generally dependent on multiple time horizons (e.g., with non-zero minimum transfer amount and threshold).

For all other cases, it is sufficient to use cashflow derivatives for XVA Greeks.

5 Key formulas for conditional expectation derivatives

In practice, we use numerical methods, namely least-squares MC, for the procedures in this paper. Thus, each CV is presented as a vector of path values $V(t) = \{V(t)[p]\}$ for $p = 1, \ldots, N_p$, and the corresponding conditional expectation is calculated using regression.

For brevity, we fix two times t_1 and t_2 and denote continuation values and discount factors with a corresponding time superscript: $V(t_i) = V^{(i)}$, $D(t_i) = D^{(i)}$. Here, the discount factor is one divided by the numeraire:

$$D(t) = 1/N(t).$$

We will write $X(t_1) = X$ for the states at time t_1.

For the numerical calculation of the conditional expectation

$$V^{(1)}(x) = \mathbb{E}\left[\left.\frac{V^{(2)}D^{(2)}}{D^{(1)}}\right| X = x\right], \tag{38}$$

we need to fix basis functions $\{\phi_n(X)\}_{n=1}^{N_b}$ and determine coefficients α_n that minimize the expectation

$$\chi^2 = \mathbb{E}\left[\left(\tilde{V}^{(2)} - \sum_n \alpha_n \phi_n(X)\right)^2 D^{(1)}\right],$$

where we define

$$\tilde{V}^{(2)} = \frac{V^{(2)}D^{(2)}}{D^{(1)}}.$$

In practice, the unconditional expectation χ^2 is calculated over the MC paths:

$$\chi^2 \simeq \frac{1}{N_p}\sum_{p=1}^{N_p}\left(\tilde{V}^{(2)}[p] - \sum_n \alpha_n \phi_n(X[p])\right)^2 D^{(1)}[p]. \tag{39}$$

Slightly abusing notations, we denote by $V^{(1)}[p]$ the pth path of results for the MC regression on our basis $\phi_n(X)$. The stochastic variable defined in such a way is obviously an approximation to the exact conditional expectation in (38). The errors come from two sources: the main one traditionally is due to the incomplete basis, while the second source is the MC statistical error. In all of the regression-based methods, the choice of the basis is the key. Below, we treat all of our equations in the MC sense, i.e., by assuming that simple averages are MC sums $\mathbb{E}[X] = (1/N_p)\sum_p X[p]$ and that conditional averages are results of the corresponding MC regressions (39).

Minimizing χ^2 over the α_n coefficients yields the set of equations

$$\sum_n \alpha_n \mathbb{E}\left[\phi_n(X)\phi_m(X)D^{(1)}\right] = \mathbb{E}\left[V^{(2)}\phi_m(X)D^{(2)}\right].$$

These can be easily solved:

$$\alpha_n = \sum_m C^{-1}_{nm}\mathbb{E}\left[\phi_m(X)\tilde{V}^{(2)}D^{(1)}\right],$$

where we have denoted the matrix

$$C_{nm} = \mathbb{E}\left[\phi_n(X)\phi_m(X)D^{(1)}\right].$$

This gives us

$$V^{(1)}(x) = \sum_n \alpha_n \phi_n(x) = \sum_{nm} \phi_n(x) C^{-1}_{nm} \mathbb{E}\left[\phi_m(X)\tilde{V}^{(2)} D^{(1)}\right]. \tag{40}$$

For a set of given paths we have

$$V^{(1)}[p] = V^{(1)}(X[p]) = \frac{1}{N_p} \sum_{nm,p'} \phi_n(x[p]) C^{-1}_{nm} \phi_m(x[p']) D^{(2)}[p'] V^{(2)}[p'].$$

As mentioned in the introduction, this linear operation $V^{(2)} \to V^{(1)}$ "mixes the paths".

Below, we present various derivatives of the regression that we will need later. Namely,

$$\frac{\partial V^{(1)}[p]}{\partial \tilde{V}^{(2)}[p']} = \frac{1}{N_p} \sum_{nm} \phi_n(x[p]) C^{-1}_{nm} \phi_m(x[p']) D^{(1)}[p'], \tag{41}$$

$$\begin{aligned}\frac{\partial V^{(1)}[p]}{\partial X[p']} &= \delta_{pp'} V^{(1)'}[p] \\ &+ \frac{1}{N_p} \sum_{nm} \phi_n(X[p]) C^{-1}_{nm} \left\{\phi'_m(X[p'])(\tilde{V}^{(2)}[p'] - V^{(1)}[p']) - \phi_m(X[p']) V^{(1)'}[p']\right\} D^{(1)}[p'],\end{aligned} \tag{42}$$

$$\frac{\partial V^{(1)}[p]}{\partial D^{(1)}[p']} = \frac{1}{N_p} \sum_{nk} \phi_n(X[p]) C^{-1}_{nk} \phi_k(X[p']) \left(\tilde{V}^{(2)}[p'] - V^{(1)}[p']\right), \tag{43}$$

where $V^{(1)'}[p]$ is the derivative of the regression result over the state, i.e.,

$$V^{(1)'}[p] = \sum_n \alpha_n \phi'_n(X[p]) = \frac{1}{N_p} \sum_{nm,p'} \phi'_n(X[p]) C^{-1}_{nm} \phi_m(X[p']) \tilde{V}^{(2)}[p'] D^{(1)}[p'],$$

and $\phi'_n(x)$ is the derivative of $\phi_n(x)$ over its argument. See [3] for more details. Note that for simplicity we have considered the one-state case $N_X = 1$; the formulas (42) can be easily transformed to the multistate case.

In what follows, for ease of notation, we will silently include the discount factors into the states set, i.e., we identify the discount factor with 0th state: $D(t) = X_0(t)$. This permits us to "absorb" the derivatives $\partial V/\partial D$ into the set $\{\partial V/\partial X_k\}_{k=0}^{N_X}$, except when specifically mentioned.

There are two sorts of $V(t)[p]$ dependencies on the states: the diagonal and the cross dependencies. Consider, for example, the derivative (42). Its diagonal dependence $\delta_{pp'} V^{(1)'}[p]$ corresponds to a derivative over the state in (40):

$$\frac{\partial V^{(1)}(x)}{\partial x} = \sum_n \alpha_n \phi'_n(x).$$

The cross dependence, related with the second term of (42), arises due the dependence of $V(t)$ on the various statistical characteristics of the states (e.g., on the average of $X(t)$ or on the states' moments) and is located in the regression coefficients α_n. One can further retranslate it into a dependence on the model parameters:

$$V(t)[p] = V\left(t, X[p]; \frac{1}{N_p}\sum_{p'} X(t)[p'], \frac{1}{N_p}\sum_{p'} X(t)^2[p'], \dots\right) = V\left(t, X[p]; \theta\right). \tag{44}$$

We will use this logic in the next section to accelerate calculations.

We conclude this section with a comment on the derivative matrices. Note that their size is large ($N_p \times N_p$) and they heavily mix the paths. However, their rank is equal to the number of basis functions N_b, which is often quite small with respect to the number of paths. We will use this fact to proceed with the AD for this seemingly difficult case of exotic options.

6 Adjoint differentiation for XVA

Analysis of the standard AD algorithm for XVA Greeks applied to backward instruments can be found in Appendix A. Here we will comment on the differences in our approach and one found in a recent paper [6]. We also summarize the memory requirements from that analysis.

Capriotti et al. (2016) applied the standard AD to an XVA for a Bermudan instrument. Our approach generalizes these results to 1) an arbitrary XVA with a general collateral and 2) an arbitrary scripted portfolio leveraging an algorithmic exposure method [2].

Another difference with [6] is in regards to the three separate tapes discussed below. Such a structure can be helpful in memory optimization; see the beginning of Appendix A for more information. Our algorithm's "tape" consists of the simulation, instrument, and XVA tapes:

- *Simulation tape*
  This step is done during simulation. The storage size of the values in (7) and (8) is proportional to $N_X^2 \times N_p \times N_t$ for time-dependent parameters.
- *Backwards pricing tape*
  The information to store in this tape includes:
  - all of the continuation values for all of the observation dates,
  - the derivatives of the continuation values during the regressions of the observation dates; i.e., the values (41)–(43).

  We write one tape per instrument. The tape is dominated by the storage of the derivatives "dV/dx" of the CVs, which has size:
  $$2N_X \times N_{\text{obs}} \times N_b \times N_p, \tag{45}$$
  where N_b is the number of basis functions used during the regression. The factor of two represents the fact that we store the basis functions as well as their derivatives.
- *Forward XVA aggregation tape*
  We store all the operations that occur during the forward XVA aggregation. The number of operations is proportional to the number of exposures:
  $$O(1)N_{\text{obs}} \times N_p.$$

Recall that N_p is the number of paths, N_{obs} is the number of observation dates, N_X is the number of model states, N_b is the number of basis functions, and N_t is the number of simulation steps, which is often close to N_{obs}.

Note that the instrument tape dominates the storage and can require a significant amount of memory.

7 Backward differentiation for XVA

This approach to the algorithmic differentiation was proposed in [3], where it was applied to PVs. It worked with *continuation values* in place of the final payoffs, which are used in the usual AD. This allowed us to avoid the instrument tape since the differentiation is done *during* the backward pricing. This saves a lot of memory as well as the time needed to access it. Moreover, it makes the code simpler and easier to debug. Here we generalize the BD method to XVAs.

7.1 Backward differentiation for pricing

Below we will briefly describe the BD algorithm for pricing given in [3]; we will also add a slight modification when differentiation of conditional expectations is not necessary.

The derivative calculations are performed during the backward pricing. Each CV $V(t)$ for a time step t stores the derivatives over the states, the path discount factor,[6] and the model parameters, i.e.,

$$\frac{\partial V(t)[p]}{\partial X(t)[p']} \quad \text{and} \quad \frac{\partial V(t)[p]}{\partial \theta}. \tag{46}$$

We note that the derivatives attached to the CV are updated during its backward propagation, so there is no need to store them for all time steps. Also, we will implicitly assume that summations are over multiple state indexes in order to make the formulas more transparent.

[6] We recall that we included the discount factors into the states set, absorbing the $\partial V/\partial D$ derivatives into $\partial V/\partial X$, except when specifically mentioned.

The derivatives over the states are represented by a *diagonal* matrix plus a *cross-term* matrix of reduced rank $N_{V(t)}$ much smaller than the number of paths. (For example, the rank after the regression is equal to the number of basis functions N_b.) Explicitly, we have

$$\frac{\partial V(t)[p]}{\partial X(t)[p']} = \Delta(t)[p]\delta_{pp'} + \sum_{n,m=1}^{N_{V(t)}} A_n(t)[p]\Omega_{nm}(t)B_m(t)[p']$$

or, in matrix notation,

$$\underbrace{\begin{Bmatrix} \begin{bmatrix} \\ \frac{\partial V(t)}{\partial X(t)} \\ \\ \end{bmatrix} \end{Bmatrix}}_{N_p} = \underbrace{\begin{Bmatrix} \begin{bmatrix} \\ \Delta(t) \\ \\ \end{bmatrix} \end{Bmatrix}}_{N_p} + \underbrace{\begin{Bmatrix} \begin{bmatrix} \\ A(t) \\ \\ \end{bmatrix} \end{Bmatrix}}_{N_{V(t)}} \underbrace{\begin{bmatrix} \\ \Omega(t) \\ \\ \end{bmatrix}}_{N_{V(t)}} \underbrace{\begin{bmatrix} \\ B(t)^T \\ \\ \end{bmatrix}}_{N_p} \Big\}_{N_V(t)}$$

The backward pricing script operations modify the CV derivatives in an algorithmic way. We will start with a general case when the regression is necessary and finish with the simplified algorithm when we can skip such differentiation.

7.1.1 Backward differentiation with the regression differentiation

The algorithm is as follows; for more details see [3]:

1. Arithmetic operations.
   Arithmetic operations with CVs modify the derivatives according to the differentiation rules. For example, the maximum operation $V = \max(V_1, V_2)$ has the following derivatives:

$$\begin{aligned}\frac{\partial \max(V_1, V_2)}{\partial X(t)[p']} = &\left(1_{V_1(t)[p]<V_2(t)[p]}\Delta_2(t)[p] + 1_{V_1(t)[p]>V_2(t)[p]}\Delta_1(t)[p]\right)\delta_{pp'} \\ &+ \sum_{nm} 1_{V_1(t)[p]<V_2(t)[p]} A_{2,n}(t)[p]\Omega_{2,nm}(t)B_{2,m}(t)[p'] \\ &+ \sum_{nm} 1_{V_1(t)[p]>V_2(t)[p]}\, A_{1,n}(t)[p]\Omega_{1,\,nm}(t)\, B_{1,m}(t)[p'].\end{aligned}$$

   We see that the operation increases the rank unless $B_1(t) \sim B_2(t)$, i.e., their columns span the same space or include it. Note that the rank is also increased by the summation operation, unless $A_1 \sim A_2$ or $B_1 \sim B_2$.
   Another example of an arithmetic operation is the discounting of a CV, e.g., for its backward propagation: $V(t) \to V(t)D(t)$. This will transform the diagonal term and the matrix A while keeping the same rank (see [3]).
2. Creation of a cashflow (financial object).
   The creation of cashflows or financial objects (FX, zero bonds, etc.) is performed as follows. A cashflow, usually an analytical function of states or parameters

$$V(t) = c(t, X(t), \theta),$$

   gives rise to the following derivatives:

$$\frac{\partial V(t)[p]}{\partial \theta} = \frac{\partial c(t)[p]}{\partial \theta} \quad \text{and} \quad \frac{\partial V(t)[p]}{\partial X(t)[p']} = \frac{\partial c(t)[p]}{\partial X(t)[p]}\delta_{p,p'}.$$

3. Backward propagation.
The backward propagation is the key part of the algorithm. We consider it in detail for the discounted regression

$$V(s_j) = \mathbb{E}\left[V(s_{j+1}) \frac{D(s_{j+1})}{D(s_j)} \middle| \mathcal{F}_{s_j} \right].$$

The induction works as follows. We suppose that we are at a time step s_{j+1} with a CV $V(s_{j+1})[p]$ and its derivatives $\partial V(s_{j+1})[p]/\partial X(s_{j+1})[p']$ and $\partial V(s_{j+1})[p]/\partial\theta$. We want to proceed backwards to a time step s_j to obtain $V(s_j)$ and its derivatives $\partial V(s_j)[p]/\partial X(s_j)[p']$ and $\partial V(s_j)[p]/\partial\theta$. Note the inclusion of $D(t)$ into the states $X(t)$.

First, we discount the CV: $V(T) \to V(T)D(T)$, following the multiplication rules. Then we perform the simple propagation of the result without the regression for the moment. We do this using the iterative substitution of the states $X(t_{n+1}) = G(t_n; X(t_n), \theta)$ as in Section 2:

$$\left.\frac{\partial V(T)[p]}{\partial X(t)[p']}\right|_t = \left.\frac{\partial V(T)[p]}{\partial X(T)[p']} \frac{\partial X(T)[p']}{\partial X(t)[p']}\right|_t, \tag{47}$$

$$\left.\frac{\partial V(T)[p]}{\partial \theta}\right|_t = \left.\frac{\partial V(T)[p]}{\partial \theta} + \sum_{p'} \frac{\partial V(T)[p]}{\partial X(T)[p']} \frac{\partial X(T)[p']}{\partial \theta}\right|_t. \tag{48}$$

Recall that we denote the CV with the substituted values as $V(T)|_t$. Next, we discount $V(T) \to V(T)/D(t)$, which effectively gives us the derivatives of $\tilde{V}(T) = V(T)D(T)/D(t)|_t$ over the states $X(t)$ and the parameters.

Finally, we calculate the regression derivatives $\partial V(t)[p]/\partial \tilde{V}(T)[p']$ and $\partial V(t)[p]/\partial X(t)[p']$ by (41)–(43) to obtain the desired result via the following matrix multiplication:

$$\frac{\partial V(t)[p]}{\partial X(t)[p']} = \underbrace{\frac{\partial V(t)[p]}{\partial X(t)[p']}}_{\text{from the regression}} + \sum_{p''} \left.\frac{\partial V(t)[p]}{\partial \tilde{V}(T)[p'']} \frac{\partial \tilde{V}(T)[p'']}{\partial X(t)[p']}\right|_t, \tag{49}$$

$$\frac{\partial V(t)[p]}{\partial \theta} = \sum_{p''} \left.\frac{\partial V(t)[p]}{\partial \tilde{V}(T)[p'']} \frac{\partial \tilde{V}(T)[p'']}{\partial \theta}\right|_t. \tag{50}$$

The final derivative over the states (49) is composed of the derivatives obtained from the regression (the first term) and from the substitution of the states (the second term).

4. The results.
At the end of the backward pricing, we obtain the derivatives of all the payoffs Q_i associated with the continuation values V_i at the origin:

$$\frac{\partial Q_i}{\partial \theta} = \frac{1}{N_p} \sum_p \frac{\partial V_i(t_0)[p]}{\partial \theta}.$$

7.1.2 Backward Differentiation without the regression differentiation

As mentioned in Section 4.1, PV Greeks for Bermudan instruments depend on the cashflow derivatives and one can ignore the regression differentiation; see (33). For BD this means the following: *if a general backward instrument does not contain an explicit step function in its pricing script, one can simply skip the regression step* (49)–(50)*, replacing it by*

$$\frac{\partial V(t)[p]}{\partial X(t)[p']} = \left.\frac{\partial \tilde{V}(T)[p]}{\partial X(t)[p']}\right|_t \quad \text{and} \quad \frac{\partial V(t)[p]}{\partial \theta} = \left.\frac{\partial \tilde{V}(T)[p]}{\partial \theta}\right|_t.$$

The rest of the steps in the algorithm (arithmetic operations, cashflows, etc.) are the same as in Section 7.1.1.

Thus, the BD with its natural backward propagation in parallel with the pricing is very efficient for the calculation of Bermudan Greeks. It does not require any instrument tape and delivers an excellent Greeks performance. In Section 9, we observe a 50 times acceleration for BD over the bump-and-reprice method when using ∼100 parameters.

7.2 Generalization to the XVA

Now we modify this procedure to obtain the FV derivatives over the parameters in order to apply them to the XVA. For this we add the OVs to the algorithmic rules. Namely, along with the derivatives over the CVs (46), we should manipulate the OVs

$$\frac{\partial v(t,s_i)[p]}{\partial X(t)[p']} \quad \text{and} \quad \frac{\partial v(t,s_i)[p]}{\partial \theta} \quad \text{for all } i \text{ such that } \ s_i \geq t.$$

As we have seen in (24)–(26), the operations on the OVs are induced from those on the CVs. Recall that as we consider the instrument (regression) dates to be (much) less frequent than the observation dates, we can state that the most frequent operation involving the OVs is the propagation (26) between the observation dates. Although the values of the OVs are not changed in this propagation, their derivatives should be propagated using the simulation substitution (47)–(48):

$$\begin{aligned}
\frac{\partial v(s_j,s_i)[p]}{\partial X(s_j)[p']} &\equiv \left.\frac{\partial v(s_{j+1},s_i)[p]}{\partial X(s_j)[p']}\right|_{s_j} = \left.\frac{\partial v(s_{j+1},s_i)[p]}{\partial X(s_{j+1})[p']}\frac{\partial X(s_{j+1})[p']}{\partial X(s_j)[p']}\right|_{s_j}, \\
\frac{\partial v(s_j,s_i)[p]}{\partial \theta} &\equiv \left.\frac{\partial v(s_{j+1},s_i)[p]}{\partial \theta}\right|_{s_j} = \left.\frac{\partial v(s_{j+1},s_i)[p]}{\partial \theta} + \sum_{p'}\frac{\partial v(s_{j+1},s_i)[p]}{\partial X(s_{j+1})[p']}\frac{\partial X(s_{j+1})[p']}{\partial \theta}\right|_{s_j},
\end{aligned}$$

for all valid OVs with $i \geq j+1$. These propagations can slow down the calculation. Indeed, they should be done for each observation date and each OV, so that the number of `double` multiplications is proportional to

$$N_{\text{obs}}^2 \times N_X^2 \times N_v \times N_p.$$

We have taken into account the fact that the derivatives $\partial X(t_j)/\partial\theta|_{t_i}$ are non-zero for $O(1)$ elements due to our assumption that the parameters are time dependent. Due to the factor N_{obs}^2, such a number can be quite high. Preferably, we should eliminate the dependence on the states from the OVs. Such a method was discussed in [3] in order to accelerate the CV derivatives. The goal was to perform a path-by-path calculation of the backward derivatives.

We return to the idea of the cross derivatives as a hidden state dependence; see (44). If we know the *entire* derivatives $\partial X(t)[p']/\partial\theta|_{t_0}$, we can calculate the CV derivatives over the parameters:

$$\frac{\partial V(t)[p]}{\partial \theta} = \sum_{p'}\left.\frac{\partial V(t)[p]}{\partial X(t)[p']}\frac{\partial X(t)[p']}{\partial \theta}\right|_{t_0}.$$

Fortunately, the entire derivative is quite often known analytically. Even when this is not the case, it can be rapidly calculated by the pathwise AD procedure (11) for the low cost found in (10).

This trick permits us to replace the derivatives storage of

$$\frac{\partial V(t)}{\partial X(t)} = \Delta(t) + A(t)\Omega(t)B^T(t) \quad \text{and} \quad \frac{\partial V(t)}{\partial \theta},$$

which contains high-rank cross derivatives, with simple derivatives over the parameters:[7]

$$\frac{\partial V(t)}{\partial X(t)} = 0 \quad \text{and} \quad \frac{\partial V(t)}{\partial \theta} \to \frac{\partial V(t)}{\partial \theta} + \left.\frac{\partial V(t)}{\partial X(t)}\frac{\partial X(t)}{\partial \theta}\right|_{t_0}, \tag{51}$$

where we use the matrix notation $[\partial V(t)/\partial X(t)]_{p,p'} = \partial V(t)[p]/\partial X(t)[p']$. Essentially, the state dependencies are pushed into the derivative over the parameters.

Thus, we can "transfer" a CV's dependence on the states into a dependence on the parameters (for derivatives) after each regression. Other arithmetical operations neither mix the paths nor introduce explicit dependence on the states. Certain cashflows can also introduce explicit dependence on the states but can be treated in a similar way as a result of a regression.

Equipped with this dependence "transfer", we return to the OVs. Their initialization rule $v(s_j,s_j) = V(s_j)$ at s_j takes into account the new CV derivatives (51) and assigns these only over the parameters:

$$\frac{\partial v(s_j,s_j)}{\partial X(s_j)} = 0 \quad \text{and} \quad \frac{\partial v(s_j,s_j)}{\partial \theta} = \frac{\partial V(s_j)}{\partial \theta}.$$

[7] We can also keep the diagonal derivatives over the states $\Delta(t)$ as explained in [3].

This operation is quite important and permits to avoid all the propagation operations:

$$\frac{\partial v(s_j,s_i)[p]}{\partial X(s_j)[p']}=0,$$
$$\frac{\partial v(s_j,s_i)[p]}{\partial \theta}=\frac{\partial v(s_{j+1},s_i)[p]}{\partial \theta}.$$

To conclude this section, we give explicit formulas for the OV derivatives of one of the most complicated operations: the maximum. We suppose that on the observation date s_j, one of the CVs has encountered a maximum operation (e.g., for a Bermudan swaption (16)):

$$V(s_j)=1_{V_1(s_j)>V_2(s_j)}V1(s_j)+1_{V_1(s_j)\le V_2(s_j)}V2(s_j).$$

This induces the following update to its OVs:

$$v(s_j,s_i)=1_{V_1(s_j)>V_2(s_j)}v_1(s_j,s_i)+1_{V_1(s_j)\le V_2(s_j)}v_2(s_j,s_i),$$

for $i\ge j$. Differentiating this over the parameters,[8] we obtain

$$\begin{aligned}\partial_\theta v(s_j,s_i)=1_{V_1(s_j)>V_2(s_j)}\partial_\theta v_1(s_j,s_i)+1_{V_1(s_j)\le V_2(s_j)}\partial_\theta v_2(s_j,s_i)\\ +\delta(V_1(s_j)-V_2(s_j))(\partial_\theta V_1(s_j)-\partial_\theta V_2(s_j))(v_1(s_j,s_i)-v_2(s_j,s_i))\end{aligned}\quad(52)$$

for $i\ge j$. Above, we have used $\partial_\theta=\partial/\partial\theta$. Note that the third term on the right-hand side of (52) includes the smoothed delta function (61); see Appendix B for a detailed discussion of this function.

Finally, at the end of the backward pricing, equipped with this algorithm, we obtain the derivatives of all the future values $v_m(s_i)=v_m(0,s_i)$ for all instruments:

$$\frac{\partial v_m(s_i)[p]}{\partial\theta}.\quad(53)$$

7.3 Aggregation of the XVA

Summing up the individual instrument FVs into the entire portfolio exposure, we come up with their derivatives over the parameters

$$\frac{\partial u(s_j)[p]}{\partial\theta}=\sum_m\frac{\partial v_m(s_j)[p]}{\partial\theta}$$

for all observation dates s_j. These derivatives will be used as the building blocks of the XVA sensitivities. There are two ways to do this:

- The first way is to come up with an XVA tape delivering, among other things, derivatives over the exposure (56). Applying the known entire derivatives $\partial u(s_j)/\partial\theta$, we update the derivatives of the XVA over the parameters
$$\frac{\partial Q}{\partial\theta}\mathrel{+}=\sum_{j,p}\frac{\partial Q}{\partial u(s_j)[p]}\frac{\partial u(s_j)[p]}{\partial\theta}.$$
Then, we apply the classical AD to the derivatives $\{\partial Q/\partial\theta,\partial Q/\partial X\}$ to obtain the required full derivatives $\partial Q/\partial\theta$.
- The second way is related to the bumped XVAs. In particular, we simply produce the bumped exposures
$$u(t;\theta+\Delta\theta)\simeq u(t;\theta)+\frac{\partial u(t)}{\partial\theta}\Delta\theta$$
using a typical bump for each parameter. Then, we substitute the bumped exposures in the XVA formula (18):
$$\mathrm{XVA}(\theta+\Delta\theta)=\sum_i\mathbb{E}\left[\frac{F_i(u(s_1;\theta+\Delta\theta),\dots,u(s_i;\theta+\Delta\theta))}{N(s_i,\theta+\Delta\theta)}\right]w_i.$$
The bumped numeraire can be obtained from its derivatives over the parameters which are readily available via the classic AD method. Finally, we have
$$\frac{\partial Q}{\partial\theta}=\frac{\mathrm{XVA}(\theta+\Delta\theta)-\mathrm{XVA}(\theta)}{\Delta\theta}.$$

[8] Here, we suppose that derivatives over the states was transferred into the derivatives over parameters.

7.4 Speed

We begin by calculating the number of multiplications in the regression setup before addressing other time-consuming operations. A single regression is dominated by $N_p \times N_b \times N_X \times N_\theta$ multiplications, where we suppose that we have pushed all the derivatives over the states into those over the parameters (see [3]). Thus, to calculate the derivatives in all the regressions we need

$$N_{\text{obs}} \times N_p \times N_b \times N_X \times N_\theta \tag{54}$$

multiplications.

A smaller source of calculation costs is related with the derivatives transfer (51). Unless we calculate the full derivatives analytically, their AD calculation (11) will take $\frac{1}{2}N_X^2 \times N_p \times N_{\text{obs}}^2$ multiplications, as seen from (10) performed for each observation date. Other time-consuming operations include:

- memory allocation/access for the FV derivatives,
- multiplications during the arithmetical operations (the maximum operation is quite heavy due to the involvement of multiple derivatives of the FVs),
- regression derivatives preparation,
- XVA aggregation.

7.5 Storage

There is no instrument tape when performing BD. The main storage requirement is related to the OV derivatives (53):

$$N_p \times N_\theta \times N_{\text{obs}}.$$

This is comparable with the storage needed for the classic AD given in (45).

8 Adjoint differentiation vs. Backward Differentiation

Here we present the main differences and similarities of the AD and BD approaches.

- Speed

  Both methods are much faster than the bump-and-reprice approach. The gain comes from the fact that the bump-and-reprice method has to recalculate the regressions during each bumped run. This regression time is mainly spent 1) finding the regression basis, which is potentially dependent on the CV; 2) calculating the basis covariance matrix C; and 3) inverting C. Given a sufficiently large number of basis functions, the default time is significant and dominates the pricing time.

  As we will see in Section 9, BD is around 10–15 times faster than the bump-and-reprice method for the XVA and around 20–50 faster for the PV.

  Formally, AD has N_θ times fewer multiplications than BD; this can be seen by comparing (54) and (60). This ratio is often attenuated by a factor of $O(1)$ due to a sparse structure of certain full derivatives. However, the CPU is also spent on writing and reading the tape during the AD, which results in the BD performance being comparable with that of the AD. We do not perform numerical experiments with the standard AD since our goal is to study tapeless methods of algorithmic differentiation, but we can estimate the AD time as being 4–10 times slower than the pricing time using the common "rule of thumb".

- Tape and storage

  BD does not use the instrument tape, so its implementation is certainly easier. BD and AD have comparable storage requirements.

- Multiple XVAs

  BD is designed to deliver model Greeks for all of the XVAs (CVA, DVA, FVA, etc.) in one pass. AD, on the other hand, should be rerun for each new XVA.

- Coding/debugging

  As we mentioned above, the AD implementation is complicated and can sometimes require the rewriting of entire libraries. Its tape structure adds a lot of complexity to the coding, debugging, and maintenance work. The BD implementation is much simpler and more compatible with the coding structure, especially given the underlying pricing script approach.

- Combination with bump-and-reprice
  If some very exotic instruments are not covered by the BD, we can always calculate the derivatives of their exposures by the bump-and-reprice method and integrate them into the general BD procedure. This is not possible for the AD method, in which all instruments should be subject to the tape and AD.

9 Numerical experiments

In this section, we demonstrate the efficiency of the backward algorithmic differentiation numerically using the HW1F model with constant parameters: 2% volatility, 5% reversion, and 2% rate.

The exotic instrument at hand is a 10-year Bermudan ATM swaption with coinciding quarterly fixing and exercise dates from 0.25 years to 10.25 years. The payment dates are shifted forward by 0.25 years, with the first occurring at 0.5 years and the last at 10.5 years.

We cover two experiments: the first one is related to the PV Greeks while the second one involves the XVA Greeks. We calculate sensitivities to different buckets of the volatility (Vega) and forward rates of the yield curve (Delta). Namely, we bump these parameters either on the annually spaced intervals (11 buckets) or on the simulation intervals plus one more interval starting at the last simulation date and ending in a year (43 buckets). For both experiments, we perform Monte Carlo simulation with **5,000 high-quality, low-discrepancy paths** and 126 monthly time steps. The conditional expectation is calculated with regression on a set of **17 smooth basis functions** similar to [7]. To obtain the Monte Carlo statistics (the mean and standard deviation) we use **64 independent runs**.

Let us start with the PV. The pricing mechanism was described in Section 3; it includes the CV transformations (13)–(14) and the conditional expectations (17) (this involves about 80 regressions). The unit notional swaption price is 0.107.

In Tables 1 and 2 below, we present the timing results (in seconds) for the calculations in three cases in which we have 22, 54, or 86 parameters total. The timing results are given for pricing alone, for Greeks computed with backward differentiation (BD), and for Greeks computed with the bump-and-reprice brute force approach (BnR). We also give the ratio of the times for the two Greek calculation approaches to show the acceleration of the BD method with respect to the BnR method, and we give the ratio of the BD time to the pricing time to show how Greeks calculation time under BD scales with respect to a single pricing time. We report results with (Table 1) and without (Table 2) regression differentiation: the algorithms are described in Section 7.1.1 and 7.1.2, respectively.

According to the adjoint differentiation rule of thumb (i.e., AD is 4–10 times slower than the pricing), BD performs very decently. Note that for pricing Greeks, BD is up to over 20 times faster than the BnR procedure with regression differentiation, and almost up to 60 times faster without it. As mentioned in Section 7.1.2, the BD method is very natural and is extremely efficient for handling the PV Greeks when the algorithm does not include the regression differentiation.

Vol Buckets	Rate Buckets	Pricing	BD	BnR	BnR/BD	BD/Pricing
11	11	0.7	1.3	15.3	12.2	1.8
11	43	0.7	1.5	36.9	24.4	2.2
43	43	0.7	2.5	59.2	23.5	3.7

Table 1: Timing results for pricing Greeks with regression differentiation (per run in seconds).

Vol Buckets	Rate Buckets	Pricing	BD	BnR	BnR/BD	BD/Pricing
11	11	0.7	0.8	15.2	20.2	1.1
11	43	0.7	0.8	38.3	46.5	1.2
43	43	0.7	1.0	58.6	57.2	1.5

Table 2: Timing results for pricing Greeks without regression differentiation (per run in seconds).

As we mentioned in the introduction, the quality of the Greeks calculated using BD and BnR is often similar given a sufficiently large bump size. For smaller bumps, the finite difference derivatives calculation can lead to a bigger statistical error (i.e., a higher standard deviation over multiple independent Monte

Carlo runs), but, potentially, a smaller bias. In Tables 3 and 4 below, we illustrate how the quality of Vega and Delta compares between the BD and BnR methods. We used a 5 bp bump size for Vega and 1 bp bump size for Delta, and we observe a similar quality between the BD and BnR methods for both the mean and standard deviation (**Std Dev**).

Date	Vega				Delta			
	Mean		Std Dev		Mean		Std Dev	
	BD	BnR	BD	BnR	BD	BnR	BD	BnR
1.00	1.490	1.493	0.018	0.016	-0.028	-0.028	0.002	0.002
2.00	1.193	1.192	0.015	0.015	-0.197	-0.197	0.005	0.005
3.00	0.863	0.864	0.011	0.011	-0.266	-0.267	0.009	0.010
4.00	0.633	0.633	0.008	0.009	-0.280	-0.280	0.011	0.010
5.00	0.465	0.464	0.007	0.006	-0.284	-0.283	0.009	0.008
6.00	0.337	0.336	0.005	0.005	-0.273	-0.275	0.010	0.009
7.00	0.235	0.236	0.004	0.004	-0.271	-0.270	0.011	0.011
8.00	0.153	0.154	0.005	0.005	-0.265	-0.265	0.013	0.013
9.00	0.087	0.087	0.002	0.002	-0.276	-0.276	0.011	0.010
10.00	0.033	0.033	0.001	0.001	-0.296	-0.297	0.012	0.010
11.00	0.002	0.002	0.000	0.000	5.609	5.612	0.011	0.011

Table 3: Price Vega and Delta means and standard deviations over 64 runs with regression differentiation.

Date	Vega				Delta			
	Mean		Std Dev		Mean		Std Dev	
	BD	BnR	BD	BnR	BD	BnR	BD	BnR
1.00	1.488	1.493	0.018	0.016	-0.027	-0.028	0.002	0.002
2.00	1.195	1.192	0.019	0.015	-0.197	-0.197	0.006	0.005
3.00	0.862	0.864	0.016	0.011	-0.266	-0.267	0.010	0.010
4.00	0.635	0.633	0.014	0.009	-0.280	-0.280	0.012	0.010
5.00	0.466	0.464	0.014	0.006	-0.283	-0.283	0.012	0.008
6.00	0.338	0.336	0.013	0.005	-0.275	-0.275	0.014	0.009
7.00	0.237	0.236	0.009	0.004	-0.269	-0.270	0.017	0.011
8.00	0.155	0.154	0.007	0.005	-0.267	-0.265	0.016	0.013
9.00	0.087	0.087	0.005	0.002	-0.278	-0.276	0.018	0.010
10.00	0.033	0.033	0.002	0.001	-0.296	-0.297	0.021	0.010
11.00	0.002	0.002	0.000	0.000	5.610	5.612	0.028	0.011

Table 4: Price Vega and Delta means and standard deviations over 64 runs without regression differentiation.

We note that the regression differentiation has smaller standard deviations due to the "numerical discontinuity" of the PV Greeks calculated without the regression differentiation (29).

Interestingly, the swaption price is highly sensitive to the last discount factor. Indeed, if we enter into a swap at some date we receive a set of fixed-for-floating cash flows equivalent to 1 at this exercise date, `-strike*dcf` for all other fixing dates and, finally, -1 at the last day. Thus, the equivalent payment of -1 is present whenever we exercise. Its derivative over the last discount factor is approximately the exercise probability. The first equivalent payment of 1 will be spread over the exercise dates, leading to small derivatives over the intermediate discount factors.

Let us pass now to the XVA Greeks, where our valuation adjustment is the CVA (36) with collateral equal to the forward value of the future swaption price. After a monthly discretization of the observation dates $\{s_i\}$ (coinciding with the simulation dates), we obtain

$$\mathrm{CVA} \simeq \sum_i \mathbb{E}\left[\frac{(v(t_i)-C_i)^+}{N(s_i)}\right] \Delta s_i \tilde{\lambda},$$

with time-independent $\tilde{\lambda} = 0.01$. The forward collateral

$$C_i = \frac{\mathbb{E}\left[v(t_i)/N(s_i)\right]}{P(0, s_i)}$$

makes the CVA "at-the-money". The XVA calculation includes around 250 regressions. The unit notional XVA itself is 0.002588.

In Table 5 below, we present the timing results (in seconds) for the XVA calculations with 22, 54, and 86 parameters. The XVA Greeks acceleration is less impressive than for the pricing, but the results are still excellent: BD is 10–14 times faster than BnR.

Vol Buckets	Rate Buckets	XVA	BD	BnR	BnR/BD	BD/XVA
11	11	1.8	3.9	39.4	10.1	2.2
11	43	1.8	6.8	97.1	14.3	3.8
43	43	1.8	11	155.1	14.1	6.1

Table 5: Timing results for XVA Greeks with regression differentiation (per run in seconds).

As we did for the pricing, we use a 5 bp bump size for Vega and 1 bp bump size for Delta, and we observe a similar quality between the BD and BnR methods for both the mean and standard deviation; see Table 6 below.

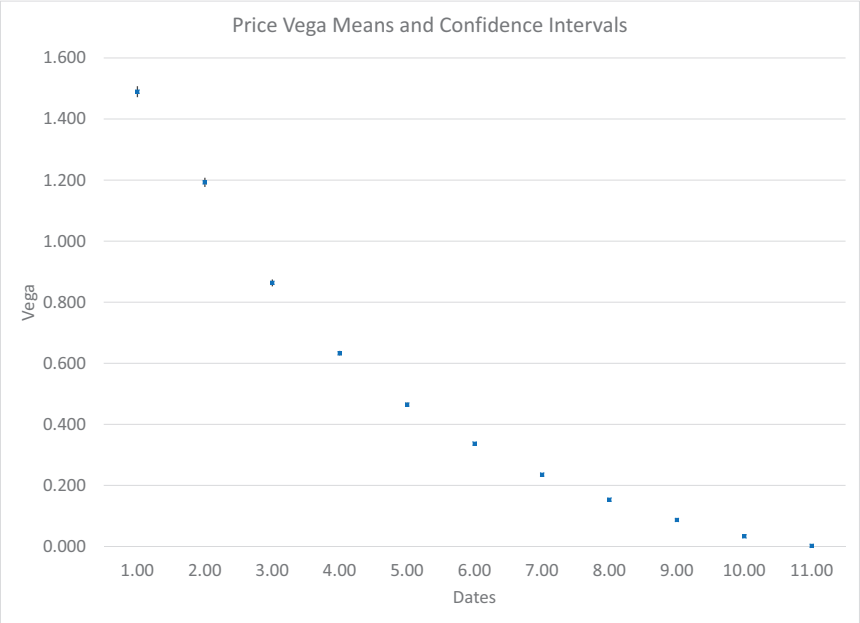
Date	Vega				Delta			
	Mean		Std Dev		Mean		Std Dev	
	BD	BnR	BD	BnR	BD	BnR	BD	BnR
1.00	0.0420	0.0415	0.0049	0.0036	-0.0010	-0.0011	0.0013	0.0017
2.00	0.0274	0.0275	0.0037	0.0028	-0.0036	-0.0039	0.0029	0.0038
3.00	0.0191	0.0195	0.0018	0.0022	-0.0033	-0.0031	0.0025	0.0038
4.00	0.0130	0.0132	0.0012	0.0016	-0.0047	-0.0044	0.0018	0.0030
5.00	0.0085	0.0085	0.0011	0.0015	-0.0036	-0.0039	0.0024	0.0026
6.00	0.0047	0.0049	0.0006	0.0010	-0.0038	-0.0035	0.0021	0.0026
7.00	0.0021	0.0021	0.0005	0.0009	-0.0020	-0.0020	0.0020	0.0022
8.00	0.0004	0.0003	0.0002	0.0008	-0.0004	-0.0004	0.0016	0.0020
9.00	-0.0005	-0.0004	0.0001	0.0005	0.0014	0.0014	0.0009	0.0017
10.00	-0.0005	-0.0005	0.0001	0.0002	0.0041	0.0042	0.0006	0.0017
11.00	0.0000	-0.0001	0.0000	0.0001	0.0614	0.0617	0.0021	0.0036

Table 6: XVA Vega and Delta means and standard deviations over 64 runs.

We would like to stress that without accounting for the delta function found in (52) we would not get the correct results; this is demonstrated in Table 7, where we compare the Vega and Delta results using BD, BnR, and BD without accounting for the delta function.

Lastly, we provide some visual illustrations of how the BD Vega and Delta look for both price and XVA in Figures 1 and 2. Each plot includes the bucket Greek's mean and one-standard-deviation confidence interval for both the Vegas and Deltas. Note that for Delta, we excluded the last bucket since its Delta is much larger than the rest and this change in scale makes it difficult to observe the behavior of the rest of the buckets.

We observe that the XVA standard deviation is much larger than that for the pricing: it is due to presence of a product of the step-functions in indicators (21) in the future price definitions (19). Equivalently, at the BD level, this noise is due to the delta functions in the derivatives over the swaption FVs (52). In Appendix B one can find a practical smoothing technique for the delta function.



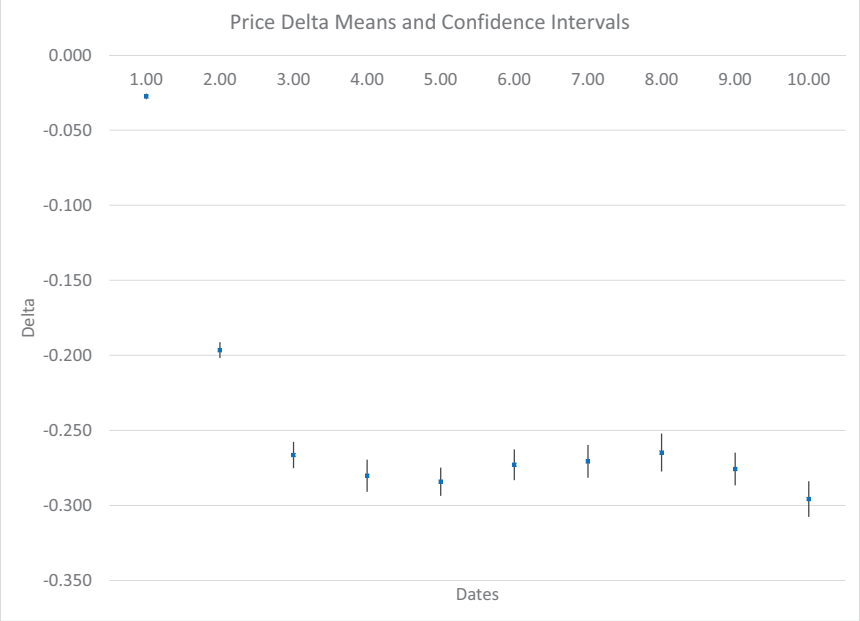
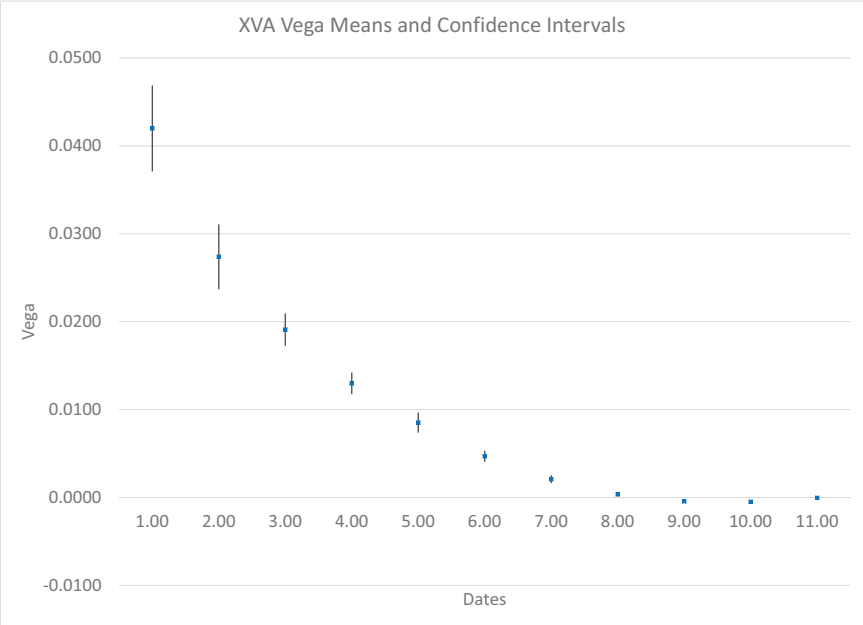


Figure 1: BD price Vega (top) and Delta (bottom) plots of means and confidence intervals.



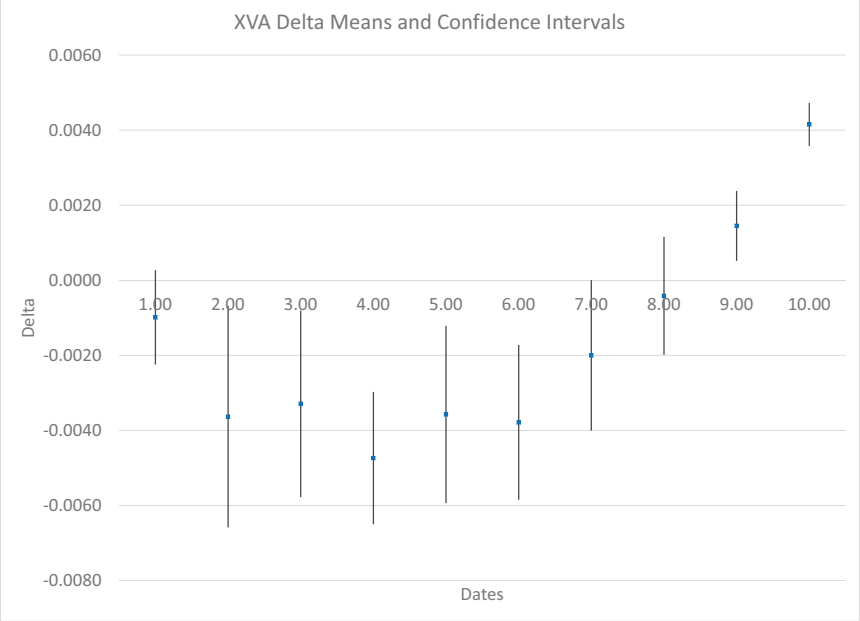


Figure 2: BD XVA Vega (top) and Delta (bottom) plots of means and confidence intervals.

Date	Vega			Delta		
	BD	**BnR**	**BD w/o δ-function**	**BD**	**BnR**	**BD w/o δ-function**
1.00	0.0420	0.0415	0.0331	-0.0010	-0.0011	-0.0001
2.00	0.0274	0.0275	0.0265	-0.0036	-0.0039	-0.0020
3.00	0.0191	0.0195	0.0205	-0.0033	-0.0031	-0.0041
4.00	0.0130	0.0132	0.0147	-0.0047	-0.0044	-0.0051
5.00	0.0085	0.0085	0.0098	-0.0036	-0.0039	-0.0052
6.00	0.0047	0.0049	0.0059	-0.0038	-0.0035	-0.0047
7.00	0.0021	0.0021	0.0030	-0.0020	-0.0020	-0.0038
8.00	0.0004	0.0003	0.0010	-0.0004	-0.0004	-0.0024
9.00	-0.0005	-0.0004	-0.0001	0.0014	0.0014	-0.0005
10.00	-0.0005	-0.0005	-0.0004	0.0041	0.0042	0.0022
11.00	0.0000	-0.0001	0.0000	0.0614	0.0617	0.0536

Table 7: XVA Vega and Delta means over 64 runs with regression differentiation.

10 Conclusion

We have introduced the backward differentiation method for XVA Greeks of arbitrary callable exotics. It is a generalization of the backward differentiation method for PV Greeks that also leverages the algorithmic exposure method. The Backward Differentiation method does not require the instrument tape and is much simpler than the standard algorithmic differentiation approach.

We have analyzed the conditions necessary for the cashflows derivatives to be sufficient for calculating PV and XVA Greeks, including the important case of PV Greeks for Bermudan swaptions. We also considered the most complex situation where differentiation of the regression is also necessary. This is almost always the case for XVA Greeks when the corresponding portfolio contains instruments with exercises. Using the example of a Bermudan swaption, we provided timing and convergence benchmarks for XVA Greeks using the backward differentiation method and compared this to results for the bump-and-reprice method to demonstrate the former's high efficiency.

The authors are indebted to their colleagues at Numerix, especially to Gregory Whitten, for supporting this work and to Luke Mohr for the excellent editing. AA is thankful to Jesper Andreasen for stimulating discussions.

References

[1] Andreasen, J. CVA on an iPad Mini. Global Derivatives, ICBI, Amsterdam, 2014.

[2] Antonov, A., Issakov, S., and Mechkov, S. Backward induction for future values. *Risk Magazine* (January 2015), 92–97.

[3] Antonov, A. Algorithmic differentiation for callable exotics. September 15, 2016. Available at https://ssrn.com/abstract=2839362.

[4] Capriotti, L., and M. Giles. Fast correlation Greeks by adjoint algorithmic differentiation. *Risk Magazine* (April 2010).

[5] Capriotti, L., and Giles, M. Adjoint Greeks made easy. *Risk Magazine* (September 2012), 96–102.

[6] Capriotti, L., Jiang, Y. and Macrina, A. AAD and least squares Monte Carlo: Fast Bermudan-tyle options and XVA Greeks. September 23, 2016. Available at https://ssrn.com/abstract=2842631.

[7] Carriere, J. Valuation of the early-exercise price for options using simulations and nonparametric regression. *Insurance: Mathematics and Economics* (December 1996), 19–30.

[8] Davidson, C. Structured products desks join the AAD revolution. *Risk Magazine* (July 2015).

[9] Giles, M., and Glasserman, P. Smoking adjoints: fast Monte Carlo Greeks. *Risk Magazine* (January 2006), 88–92.

[10] ISDA Workshop on AAD and alternatives for CVA sensitivities. September 29, 2016. London.

[11] Sherif, N. Banks fear costs from loss of AAD under simpler FRTB rules. *Risk Magazine* (August 2016).

A Adjoint differentiation algorithm

In this appendix we consider the application of the standard AD algorithm to XVA Greeks for exotic instruments that are priced by backward induction by extending the analysis in [3] for PV Greeks. The storage requirements and speed of this method are summarized at the end of this section.

Once the tapes are prepared, we begin the adjoint algorithm that is desecribed below:

1. Derivatives attribution

   The XVA, denoted Q, bears two (adjoint) derivatives: derivatives with respect to the required parameters, $\partial Q/\partial\theta_k$ for all k, and intermediate derivatives with respect to the model states and discount factors, $\partial Q/\partial X(t_i)[p]$ and $\partial Q/\partial D(t_i)[p]$ for all time step indexes i and paths p.

2. Backward calculation of the XVA derivatives as functions of the stored exposures

   During this step we associate to every timestep portfolio exposure vector $u(t)$ a vector of the payoff derivatives over it:

   $$u(t)[p] \to \frac{\partial Q}{\partial u(t)[p]},$$

   where $[p]$ indicates a path number. We then update these during the propagation with algorithmically induced rules:

   - *Initialization of a derivative over the exposure.*
     The XVA update at t is equal to

     $$Q \mathrel{+}= \Delta t N_p^{-1} w \sum_p u^+(t)[p] D(t)[p],$$

     where N_p is the number of paths and w is a weight related to the default probability and a loss given default. Clearly,

     $$\frac{\partial Q}{\partial u(t)[p]} = \Delta t N_p^{-1} w D(t)[p] 1_{u(t)[p]>0} \quad \text{and} \quad \frac{\partial Q}{\partial D(t)[p]} \mathrel{+}= \Delta t N_p^{-1} w u^+(t)[p].$$

     This operation initializes $\partial Q/\partial u(t)[p]$ (a property of the exposure) and updates $\partial Q/\partial D(t)[p]$ (a property of the XVA). Recall that according to our convention the discount factor is identified with the 0th state and we denote $\partial Q/\partial D = \partial Q/\partial X_0$.

   - *Arithmetic operations.*
     A *summation* $u(t) = u_1(t) + u_2(t)$ will update derivatives over, say, partial exposures u_1 and u_2 given $\partial Q/\partial u(t)[p]$. The logic is based on the substitution $Q(u(t)) = Q(u_1(t) + u_2(t))$, leading to

     $$\frac{\partial Q}{\partial u_n(t)[p]} = \frac{\partial Q}{\partial u(t)[p]} \quad \text{for} \quad n = 1, 2. \tag{55}$$

     In the same way, the *multiplication* derivatives $v(t) = u_1(t)u_2(t)$ leads to the following derivatives

     $$\frac{\partial Q}{\partial u_n(t)[p]} = \frac{\partial Q}{\partial v(t)[p]} u_{3-n}(t)[p] \quad \text{for} \quad n = 1, 2.$$

     The other operations are done analogously. We note that explicit calls to financial objects (e.g., those underlying the collateral construction) can introduce an explicit XVA behavior from the states, parameters, and discount factors.

   - *The result.*
     When the procedure is finished at time zero, we know the derivatives with respect to the exposures, the parameters, and the states, i.e.,

     $$\frac{\partial Q}{\partial u(s_j)[p]}, \quad \frac{\partial Q}{\partial \theta}, \quad \frac{\partial Q}{\partial X(s_j)[p]}, \tag{56}$$

     for all observation dates s_j, paths p, and parameters θ.

3. Forward calculation of the XVA derivatives over the states and parameters

   This step is done in the forward direction by substituting the exposures into the derivatives. It is very important to state that it can be done *separately* for individual instruments. Denote an individual instrument's exposure as $u_a(t)$. The full portfolio's exposure is the sum

   $$u(t) = \sum_a u_a(t).$$

Clearly, the derivatives of the XVA over individual instruments' exposures can be calculated using the summation rule above, i.e.,

$$\frac{\partial Q}{\partial u_a(t)[p]} = \frac{\partial Q}{\partial u(t)[p]}$$

Then, the forward substitution can be done *separately* for each instrument when updating the XVA derivatives

$$\frac{\partial Q}{\partial \theta_k} \text{ and } \frac{\partial Q}{\partial X(t_i)[p]}$$

because these derivatives are never used directly in instrument exposure derivatives manipulations. Thus, for each individual instrument we start with the initialization of the derivatives:

$$\frac{\partial Q}{\partial u_a(0,s_i)[p]} = \frac{\partial Q}{\partial u(s_j)[p]},$$

which is due to our identification rule of the FV via the OV at zero, i.e., $u_a(s_i) = u_a(0,s_i)$, and proceed forward over all the observation dates following the same induced rules as in the previous step. At a given observation date s_j we have the following derivatives:

$$\frac{\partial Q}{\partial V(s_j)[p]} \quad \text{and} \quad \frac{\partial Q}{\partial v(s_j,s_i)[p]} \quad \text{for all} \quad i \geq j, \tag{57}$$

where we silently omit the CV index $V_m \to V$ and will introduce it explicitly where needed.

Some examples of operations we can encounter during the induction step at the observation date s_j are:

- *Arithmetical operations.*
  Consider here the summation operation (the others, including `-`, `*`, and `/`, are done similarly). A summation operation on the CVs at the given observation date s_j,

  $$V(s_j) = V_1(s_j) + V_2(s_j),$$

  induces the following update of OVs:

  $$v(s_j,s_i) = v_1(s_j,s_i) + v_2(s_j,s_i)$$

  for $i \geq j$. This will give us the following analog of (55):

  $$\frac{\partial Q}{\partial V_n(s_j)[p]} = \frac{\partial Q}{\partial V(s_j)[p]} \quad \text{and} \quad \frac{\partial Q}{\partial v_n(s_j,s_i)[p]} = \frac{\partial Q}{\partial v(s_j,s_i)[p]}$$

  for $n = 1,2$ and $i \geq j$.
- *Creation of a cashflow (financial object).*
  This operation deals with the creation of FX rates, zero bonds, etc. in terms of CVs that are known functions of the states or parameters. Assign such a cashflow to some CV, say V_1:

  $$V_1(t) = c(t, X(t), \theta).$$

  This leads to the substitution

  $$Q(V_1(t)) = Q(c(t, X(t), \theta)),$$

  which gives rise to the updates

  $$\frac{\partial Q}{\partial \theta} \mathrel{+}= \frac{\partial Q}{\partial V_1(t)[p]} \frac{\partial c(t)[p]}{\partial \theta} \quad \text{and} \quad \frac{\partial Q}{\partial X(t)[p]} \mathrel{+}= \frac{\partial Q}{\partial V_1(t)[p]} \frac{\partial c(t)[p]}{\partial X(t)[p]}.$$

- *Maximum.*
  We suppose that on the observation date s_j one of the CVs has encountered a maximum operation (e.g., as in (16) for a Bermudan swaption):

  $$V(s_j) = 1_{V_1(s_j) > V_2(s_j)} V_1(s_j) + 1_{V_1(s_j) \leq V_2(s_j)} V_2(s_j).$$

  This induces the following update of the corresponding OVs:

  $$v(s_j,s_i) = 1_{V_1(s_j) > V_2(s_j)} v_1(s_j,s_i) + 1_{V_1(s_j) \leq V_2(s_j)} v_2(s_j,s_i) \tag{58}$$

for $i \geq j$.
Consider first the CV derivatives. Substituting $Q(V(s_j)[p]) = Q(\max(V_1(s_j)[p], V_2(s_j)[p]))$ gives us the following derivatives over $V_1(s_j)$ and $V_2(s_j)$:

$$\frac{\partial Q}{\partial V_1(s_j)[p]} \mathrel{+}= 1_{V_1(s_j)[p] > V_2(s_j)[p]} \frac{\partial Q}{\partial V(s_j)[p]},$$
$$\frac{\partial Q}{\partial V_2(s_j)[p]} \mathrel{+}= 1_{V_1(s_j)[p] < V_2(s_j)[p]} \frac{\partial Q}{\partial V(s_j)[p]}.$$

Then we make the substitution $Q(v(s_j, s_i))$ using (58), which updates the following derivatives:

$$\frac{\partial Q}{\partial V_1(s_j)[p]} \mathrel{+}= \delta(V_1(s_j)[p] - V_2(s_j)[p])(v_1(s_j, s_i)[p] - v_2(s_j, s_i)[p]) \frac{\partial Q}{\partial v(s_j, s_i)[p]},$$
$$\frac{\partial Q}{\partial V_2(s_j)[p]} \mathrel{+}= \delta(V_2(s_j)[p] - V_1(s_j)[p])(v_2(s_j, s_i)[p] - v_1(s_j, s_i)[p]) \frac{\partial Q}{\partial v(s_j, s_i)[p]},$$
$$\frac{\partial Q}{\partial v_1(s_j, s_i)[p]} \mathrel{+}= 1_{V_1(s_j)[p] > V_2(s_j)[p]} \frac{\partial Q}{\partial v(s_j, s_i)[p]},$$
$$\frac{\partial Q}{\partial v_2(s_j, s_i)[p]} \mathrel{+}= 1_{V_1(s_j)[p] < V_2(s_j)[p]} \frac{\partial Q}{\partial v(s_j, s_i))[p]},$$

for $i \geq j$. The δ function that we use here is, of course, the"smoothed" version in (61). The details can be found in Appendix B.

- *Regression.*
This item is the most complicated and time-consuming. Consider the playback of a regression from s_j to s_{j+1}. Our set of derivatives (57) will be transformed by this propagation: the seasoned derivative $\partial Q / \partial v_m(s_{j+1}, s_j)$ will be removed, while those over the OVs (26) remain the same:

$$\frac{\partial Q}{\partial v(s_{j+1}, s_i)[p]} = \frac{\partial Q}{\partial v(s_j, s_i)[p]} \quad \text{for} \quad i \geq j+1.$$

The derivatives over the CVs will be transformed as follows. The dependence on $V(s_{j+1})$ will come from the regression of $V(s_j)$ as well as the OV initiations $V(s_{j+1}) = v(s_{j+1}, s_{j+1})$.
As we have seen in Section 3, the derivative of the regression

$$V(s_j) = \mathbb{E}\left[V(s_{j+1}) D(s_{j+1}) / D(s_j) | \mathcal{F}_{s_j}\right]$$

is related to the matrix

$$\frac{\partial V(s_j)[p]}{\partial \tilde{V}(s_{j+1})[p']} = \frac{1}{N_p} \sum_{nm} \phi_n(X((s_j))[p]) C^{-1}_{nm} \phi_m(X((s_j))[p']) D(s_j)[p'],$$

which has a small rank with respect to the number of paths N_p and which we have calculated in (41), denoting $\tilde{V}(s_{j+1}) = V(s_{j+1}) D(s_{j+1}) / D(s_j)$. This gives the derivative of the PV over the discounted CV as

$$\frac{\partial Q}{\partial \tilde{V}(s_{j+1})[p]} = \sum_{p'} \frac{\partial Q}{\partial V(s_j)[p']} \frac{\partial V(s_j)[p']}{\partial \tilde{V}(s_{j+1})[p]},$$

which has a computational cost of $N_p \times N_b$. Using the arithmetic operation logic described above (for the multiplication and division by the discount factors), we can transform this value into the regression derivative $\partial Q / \partial V(s_{j+1})[p]$. The final derivative is obtained after the OV initiations $V(s_{j+1}) = v(s_{j+1}, s_{j+1})$, leading to

$$\frac{\partial Q}{\partial V(s_{j+1})[p]} \mathrel{+}= \frac{\partial Q}{\partial v(s_{j+1}, s_{j+1})[p]}.$$

In the same way, we update the derivatives over the states:

$$\frac{\partial Q}{\partial X(s_j)[p]} \mathrel{+}= \sum_{p'} \frac{\partial Q}{\partial V(s_j)[p']} \frac{\partial V(s_j)[p']}{\partial X(s_j)[p]}, \tag{59}$$

where the matrix $\partial V(s_j)[p'] / \partial X(s_j)[p]$ also has a reduced rank as discussed in Section 3. The cost of one regression is $2N_X \times N_p \times N_b$.

The overall cost of the whole tape playback is dominated by the regression derivatives and is proportional to

$$2N_{\text{obs}} \times N_X \times N_p \times N_b$$

multiplications.

At the end of the day (instrument maturity) the instrument tape playback returns derivatives over the parameters and the states $\partial Q/\partial\theta_k$ and $\partial Q/\partial X(t_i)[p]$ for all parameters k, timesteps i, and paths p. As mentioned above, in the case of a portfolio containing multiple instruments, one can simply sum up the above derivatives of the parameters and the states to obtain the derivatives for the entire portfolio.

The overall cost is dominated by (59) and is equal to

$$O(1)N_{\text{obs}} \times N_X \times N_p \times N_b$$

multiplications per instrument.

4. Backward calculation of the PV derivatives
   At the end of the previous step (instrument tape playback) we have the XVA derivatives over the parameters and the states corresponding to the entire portfolio. To obtain the final derivatives over the parameters one simply runs the classic AD procedure (11). This operation is path-by-path and, in general, is much faster than one requiring regressions (the cost is given in (10)).

Summing up our analysis of the XVA AD we can state that the overall cost is dominated by the backward instrument calculation in Step 2 and is therefore

$$2N_r \times N_X \times N_p \times N_b \tag{60}$$

multiplications. We note that, strictly speaking, the multiplication cost is accompanied by a linear in N_θ term but that this has a tiny coefficient.

The storage cost is mainly determined by the cost of the instrument tape and can be significant. The following steps can be used to save on memory costs:

- ☐ Simulate the model and calculate the instrument future prices as usual (without the tape).
- ☐ Write the XVA tape.
- ☐ For each instrument, calculate future prices backward and write the pricing tape, then produce its AAD.
- ☐ Collect XVA derivatives from each instrument, which will form a forward instrument as in Section 2, then proceed with the path-by-path AD.

B Smoothed delta function

Let $X, Y \sim \mathcal{N}(0,1)$ and $\langle X, Y\rangle = \rho$, or in other words, $X = \rho Y + \sqrt{1-\rho^2}Z$, and $\langle Z, Y\rangle = 0$. Then

$$\begin{aligned}
\mathbb{E}[1_{X>x}] &= \mathbb{P}(X > x) \\
&= \int_{\mathbb{R}} \mathbb{P}(\rho Y + \sqrt{1-\rho^2}Z > x \mid Y = y)\phi(y)dy \\
&= \int_{\mathbb{R}} \mathbb{P}\left(Z > \frac{x-\rho y}{\sqrt{1-\rho^2}}\right)\phi(y)dy \\
&= \mathbb{E}\left[\Phi\left(\frac{\rho Y - x}{\sqrt{1-\rho^2}}\right)\right].
\end{aligned}$$

Note that the left-hand side depends only on x, and the right-hand side no longer depends on either X or Z. Since X and Y have the same distribution, we can write, for any $\rho \in (-1, 1)$,

$$\mathbb{E}[1_{X>x}] = \mathbb{E}\left[\Phi\left(\frac{\rho X - x}{\sqrt{1-\rho^2}}\right)\right].$$

It is easy to generalize this to $X \sim \mathcal{N}(\mu, \sigma)$:

$$\mathbb{E}[1_{X>x}] = \mathbb{E}\left[1_{\frac{X-\mu}{\sigma} > \frac{x-\mu}{\sigma}}\right] = \mathbb{E}\left[\Phi\left(\frac{\rho\frac{X-\mu}{\sigma} - \frac{x-\mu}{\sigma}}{\sqrt{1-\rho^2}}\right)\right] = \mathbb{E}\left[\Phi\left(\frac{\mu + \rho(X-\mu) - x}{\sigma\sqrt{1-\rho^2}}\right)\right].$$

Note that the left-hand side is an expectation of a discontinuous function of x, the Heaviside function, whereas the right-hand side is an expectation of a continuous function which for $|\rho| \approx 1$ looks a lot like the Heaviside function. We will use this fact to adapt the expression in the right-hand side's expectation as a smoothed version of the Heaviside function, and its derivative as a smoothed version of the δ-function, to use whenever we require such a smoothed version for differentiation. Namely, we define

$$\begin{aligned} H_\epsilon(x) &:= \Phi\left(\frac{\mu + \sqrt{1-\epsilon^2}(X-\mu) - x}{\sigma\epsilon}\right), \\ \delta_\epsilon(x) &:= \frac{1}{\sigma\epsilon}\phi\left(\frac{\mu + \sqrt{1-\epsilon^2}(X-\mu) - x}{\sigma\epsilon}\right), \end{aligned} \tag{61}$$

where $\mu = \mathbb{E}[X]$ and $\sigma = \sqrt{\mathbb{V}[X]}$ for X with any distribution. Note that in our experiments we use $\epsilon = 10^{-3}$.