Real-Time Risk Management with Adjoint Algorithmic Differentiation

Yupeng Jiang

Department of Mathematics, University College London

June 28, 2016

Joint work with L. Capriotti and A. Macrina

Introduction

Financial computations, no matter whether by Monte Carlo or PDE approaches, can be complex and time-consuming.

The traditional way of computing sensitivities is bumping, which requires double computations. This adds complexity and inefficiency.

The main idea of AAD is to decompose functions into basic operations which are easy to differentiate on a computer.

We present the application of AAD to regression based Monte Carlo approaches such as those that are widely used for Bermudan options and for XVA applications

How a function is implemented?

Considering a computer implemented function of the form

```
Y = FUNCTION(X)
```

mapping a vector $X \in \mathbb{R}_n$ to a vector $Y \in \mathbb{R}_m$ through a sequence of steps

$$X \to \dots \to U \to V \to \dots \to Y$$

Here, the real vectors U and V represent intermediate variables used in the calculation and each step can be a distinct high-level function or even a specific instruction.

The adjoints and chain rule

We define the adjoint of any intermediate variable V_k by

$$\bar{V}_k = \sum_{j=1}^{m} \bar{Y}_j \frac{\partial Y_j}{\partial V_k}$$

where $\bar{Y}$ is a vector in $\mathbb{R}_m$ For each of the intermediate variables U_i, by applying the chain rule, we get

$$\bar{U}_i = \sum_{j=1}^{m} \bar{Y}_j \frac{\partial Y_j}{\partial U_i} = \sum_{j=1}^{m} \bar{Y}_j \sum_{k} \frac{\partial Y_j}{\partial V_k} \frac{\partial V_k}{\partial U_i}$$

which corresponds to the adjoint mode equation for the intermediate step represented by the function $V = V(U)$.

AAD algorithm

We start from the adjoint of the outputs $\bar{Y}$, and apply this rule to each step in the calculation, working

$$\bar{X} = \texttt{FUNCTION\_b}(X, \bar{Y})$$

by

$$\bar{X} \leftarrow ... \leftarrow \bar{U} \leftarrow \bar{V} \leftarrow ... \leftarrow \bar{Y}$$

until we obtain the inputs adjoints $\bar{X}$.

In the adjoint mode, the cost does not increase with the number of inputs. One particularly important theoretical result is

$$\frac{\text{Cost}\left[\texttt{FUNCTION\_b}\right]}{\text{Cost}\left[\texttt{FUNCTION}\right]} < \omega_A$$

where $\omega_A \in [3, 4]$.

Bermudan options and dynamic programming principle

The value of a Bermudan option is the supremum of the option value over all the possible exercise policies $E(\tau)$, where $\tau \in \mathcal{T}(t)$, the strikable time.

$$\frac{V(t)}{N(t)} = \sup_{\tau \in \mathcal{T}(t)} \mathbb{E}_t \left[\frac{E(\tau)}{N(\tau)} \right] . \tag{1}$$

We indicate with $H_\eta(t)$ the holding value of the Bermudan option when $T_{\eta(t)} \leq t < T_{\eta(t)+1}$, we have

$$\frac{H_{\eta(t)}(t)}{N(t)} = \mathbb{E}_t \left[\frac{V(T_{\eta(t)+1})}{N(T_{\eta(t)+1})} \right] . \tag{2}$$

Bermudan options and dynamic programming principle

It leads to the so-called dynamic programming formulation, namely,

$$\frac{H_{\eta(t)}(t)}{N(t)} = \mathbb{E}_t\Big[\max\Big(\frac{E(T_{\eta(t)+1})}{N(T_{\eta(t)+1})}, \frac{H_{\eta(t)+1}(T_{\eta(t)+1})}{N(T_{\eta(t)+1})}\Big)\Big] \,, \tag{3}$$

for $T_\eta \leq t < T_{\eta+1}$, and $\eta = 1, \ldots, M-1$.

The dynamic programming formulation above implies that the stopping time defining optional exercise as seen as time t is given by

$$\tau^\star = \inf[T_m \geq t : E(T_m) \geq H_m(T_m)] \,. \tag{4}$$

XVA and nested Monte Carlo

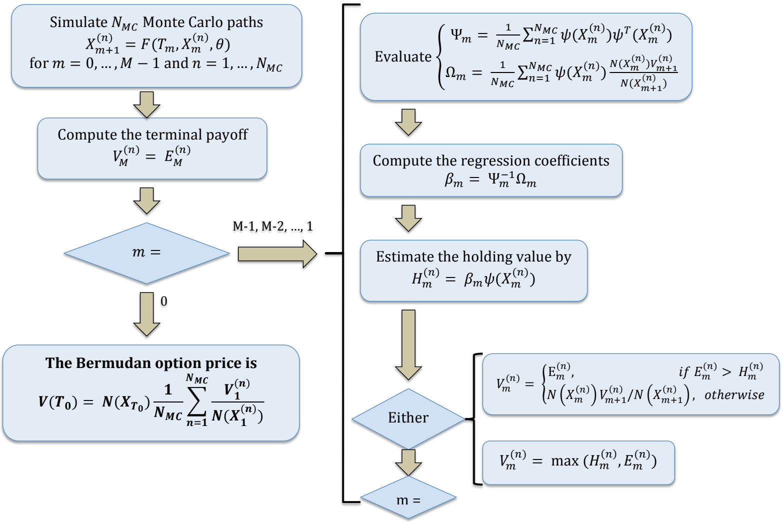
For simplicity, we consider the calculation of the CVA and of the DVA as the main measures of a dealer's counterparty credit risk. This can be evaluated at time $t_0 = 0$ as

$$\mathrm{XVA} = -\mathbb{E}\left[\mathbb{I}(\tau_c \leq \mathrm{T})\frac{\mathrm{L_c}}{\mathrm{N}(\tau_c)}\left(\mathrm{V}(\tau_c)\right)^{+} + \mathbb{I}(\tau_d \leq \mathrm{T})\frac{\mathrm{L_d}}{\mathrm{N}(\tau_d)}\left(\mathrm{V}(\tau_d)\right)^{-}\right], \quad (5)$$

In general, its calculation requires a MC simulation due to the correlated variables.

However, the conditional future NPV can rarely be expressed in closed form. It means we requires a nested Monte Carlo to evaluate the XVA, which is inefficient in both time and memory.

The valuation of Bermudan options



Low estimator of Bermudan option prices

Independently simulate another N_{MC} Monte Carlo paths

$$\tilde{X}_{m+1}^{(n)} = F(T_m, \tilde{X}_m^{(n)}, \theta)$$

for $m = 0, \dots, M-1$ and $n = 1, \dots, N_{MC}$

↓

Apply the βs obtained in previous algorithm.
Estimate holding value by

$$\widetilde{H}_m^{(n)} = \beta_m^T \psi\left(\tilde{X}_m^{(n)}\right)$$

↓

Evaluate the discounted optimal exercising value by

$$\tilde{E}_\tau^{(n)} = \frac{\tilde{E}_1^{(n)} \mathbb{I}\left(\tilde{E}_1^{(n)} > \widetilde{H}_1^{(n)}\right)}{\widetilde{N}_1^{(n)}} + \sum_{m=2}^{M} \left[\left(\prod_{j=1}^{m-1} \mathbb{I}\left(\widetilde{H}_j^{(n)} > \tilde{E}_j^{(n)}\right)\right) \frac{\tilde{E}_m^{(n)} \mathbb{I}\left(\tilde{E}_m^{(n)} > \widetilde{H}_m^{(n)}\right)}{\widetilde{N}_m^{(n)}}\right]$$

↓

The Bermudan option price low estimator is

$$\boldsymbol{V(T_0)} = \frac{\mathbf{1}}{\boldsymbol{N_{MC}}} \sum_{\boldsymbol{n=1}}^{\boldsymbol{N_{MC}}} \tilde{E}_\tau^{(n)}$$

Computing XVA by regression-based Monte Carlo

Simulate N_{MC} Monte Carlo paths of $X_m^{(n)}$

$$X_{m+1}^{(n)} = F(T_m, X_m^{(n)}, \theta)$$

and the intensity paths of both parties

$$\lambda_{m+1}^{c,(n)} = G_c(T_m, \lambda_m^{c,(n)}, \theta)$$

$$\lambda_{m+1}^{d,(n)} = G_c(T_m, \lambda_m^{d,(n)}, \theta)$$

for $m = 0, \dots, M-1$ and $n = 1, \dots, N_{MC}$

Compute the survival probability of each counterparty by

$$SP_m^{c,(n)} = exp\left[-\sum_{j=0}^{m-1} \lambda_j^{c,(n)}(T_{j+1} - T_j)\right]$$

$$SP_m^{d,(n)} = exp\left[-\sum_{j=0}^{m-1} \lambda_j^{d,(n)}(T_{j+1} - T_j)\right]$$

Approximate the holding value of the p-th asset in the portfolio by regression

$$H_{p,m}^{(n)} = \beta_{p,m}^T \, \psi_p(X_{T_m}^{(n)})$$

$$V_{p,m}^{(n)} = \begin{cases} max\left(H_{p,m}^{(n)}, E_{p,m}^{(n)}\right), & if\ T_m \in \mathcal{T}_p(T_0) \\ H_{p,m}^{(n)}, & otherwise \end{cases}$$

$$V_m^{(n)} = \sum_{p=1}^{P} V_{p,m}^{(n)}$$

$$XVA^{(n)} = -\sum_{m=1}^{M}\left[\left(SP_{m-1}^{c,(n)} - SP_m^{c,(n)}\right)\frac{\left(V_m^{(n)}\right)^+}{N_m^{(n)}} + \left(SP_{m-1}^{d,(n)} - SP_m^{d,(n)}\right)\frac{\left(V_m^{(n)}\right)^-}{N_m^{(n)}}\right]$$

$$\boldsymbol{XVA} = \frac{\boldsymbol{1}}{\boldsymbol{N_{MC}}}\sum_{\boldsymbol{n=1}}^{\boldsymbol{N_{MC}}} \boldsymbol{XVA^{(n)}}$$

Function regularisations

AAD requires the function to be Lipschitz continuous in each step of the AAD procedure.

Most of the financial derivatives with non-Lipschitz continuities, one can express their payoff by applying the Heaviside function $\mathcal{H}(x)$.

One can approximate $\mathcal{H}(x)$ by the call spread formula

$$\mathcal{H}(x-K) \approx \mathcal{H}_\delta^{\mathrm{cs}}(x-K) = \left(\min\left(\frac{x-(K-\delta)}{2\delta}, 1\right)\right)^+, \tag{6}$$

where $\delta \ll K$.

As $\delta \to 0$, most derivatives of $\mathcal{H}_\delta^{\mathrm{cs}}(x-K)$ in the sample space are vanishingly small, but a few of them is very large instead. It leads to large variances of the path-wise derivative estimator.

AAD for Bermudan options

$$\bar{V}_1^{(n)} = \frac{\bar{V}(T_0)N_0}{N_{MC}N_1^{(i)}}$$

$$\bar{X}_M^{(n)} += \bar{V}_M^{(n)} \frac{\partial E_M^{(n)}}{\partial X_M^{(n)}}$$

For $m = M-1, \dots, 0$, we collect

$$\bar{\theta} += \sum_{n=1}^{N_{MC}} \bar{X}_{m+1}^{(n)} \frac{\partial F}{\partial \theta}(T_m, X_m^{(n)}, \theta)$$

Do **for $m = 1, \dots, M$**

Either

$$\bar{E}_m^{(n)} = \bar{V}_m^{(n)} \, \mathbb{I}\left(H_m^{(n)} < E_m^{(n)}\right)$$

$$\bar{V}_{m+1}^{(n)} = \frac{\bar{V}_m^{(n)} N_m^{(i)}}{N N_{m+1}^{(i)}} \mathbb{I}\left(H_m^{(n)} > E_m^{(n)}\right)$$

$$\bar{N}_m^{(i)} = \mathbb{I}\left(H_m^{(n)} > E_m^{(n)}\right)\left[\bar{V}_m^{(n)} \frac{V_{m+1}^{(n)}}{N_m^{(n)}} - \bar{V}_{m-1}^{(n)} \frac{N_{m-1}^{(n)} V_m^{(n)}}{\left(N_m^{(n)}\right)^2}\right] \text{ for } 1 < m < M$$

$$\bar{N}_1^{(i)} = \mathbb{I}\left(H_1^{(n)} > E_1^{(n)}\right) \bar{V}_1^{(n)} \frac{V_2^{(n)}}{N_2^{(n)}},$$

$$\bar{N}_M^{(i)} = \mathbb{I}\left(H_M^{(n)} > E_M^{(n)}\right) - \bar{V}_{M-1}^{(n)} \frac{N_{M-1}^{(n)} V_M^{(n)}}{\left(N_M^{(n)}\right)^2}$$

or

$$\bar{E}_m^{(n)} = \bar{V}_m^{(n)} \, \mathbb{I}\left(H_m^{(n)} < E_m^{(n)}\right)$$

$$\bar{H}_m^{(n)} = \bar{V}_m^{(n)} \, \mathbb{I}\left(H_m^{(n)} > E_m^{(n)}\right)$$

then

$$\bar{\beta}_m += \sum_{n=1}^{N_{MC}} \bar{H}_m^{(n)} \psi(X_m^{(n)})$$

$$\bar{\psi}\left(X_m^{(n)}\right) = \bar{H}_m^{(n)} \beta_m$$

$$\bar{\Omega}_m = \Psi_m^{-1} \bar{\beta}_m$$

$$\bar{\Psi}_m = -\Omega_m \bar{\beta}_m^{T}$$

$$\mathcal{N}_m^{(i)} = \frac{N_m^{(i)}}{N_{m+1}^{(i)}},$$

$$\bar{\mathcal{N}}_m^{(i)} = \frac{\psi^T\left(X_m^{(n)}\right) V_{m+1}^{(n)} \bar{\Omega}_m}{N_{MC}}.$$

$$\bar{\psi}\left(X_m^{(n)}\right) += \frac{1}{N_{MC}}\left[\left(\bar{\Psi}_m + \bar{\Psi}_m^{\,T}\right) \psi\left(X_m^{(n)}\right) + \mathcal{N}_m^{(i)} V_{m+1}^{(n)} \bar{\Omega}_m\right],$$

$$\bar{X}_m^{(n)} += \frac{\bar{\mathcal{N}}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_m^{(i)}}{\partial X_m^{(n)}}, \quad \bar{X}_{m+1}^{(n)} += -\frac{\bar{\mathcal{N}}_m^{(i)} \mathcal{N}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_{m+1}^{(i)}}{\partial X_{m+1}^{(n)}},$$

$$\bar{\theta} += \frac{\bar{\mathcal{N}}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_m^{(i)}}{\partial \theta} - \frac{\bar{\mathcal{N}}_m^{(i)} \mathcal{N}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_{m+1}^{(i)}}{\partial \theta}.$$

$$\bar{X}_m^{(n)} += \bar{\psi}^T\left(X_m^{(n)}\right) \frac{\partial \psi}{\partial X_m^{(n)}}$$

AAD for low estimator

For $m = M - 1, \ldots, 0$, we collect

$$\bar{\theta} \mathrel{+}= \sum_{n=1}^{N_{MC}} \bar{X}_{m+1}^{(n)} \frac{\partial F}{\partial \theta}(T_m, X_m^{(n)}, \theta)$$

$$\bar{\beta}_m = \sum_{n=1}^{N_{MC}} \bar{H}_{p,m}^{(n)} \psi_p(X_{T_m}^{(n)})$$

$$\bar{\psi}_p\left(X_{T_m}^{(n)}\right) = \bar{H}_{p,m}^{(n)} \beta_m$$

$$\bar{X}_m^{(n)} = \bar{E}_m^{(n)} \frac{\partial E_m^{(n)}}{\partial X_m^{(n)}} + \bar{N}_m^{(n)} \frac{\partial N_m^{(n)}}{\partial X_m^{(n)}} + \bar{\psi}_p{}^{T}(X_{T_m}^{(n)}) \frac{\partial \psi_p}{\partial X_m^{(n)}}(X_{T_m}^{(n)})$$

$$\bar{\theta} = \sum_{n=1}^{N_{MC}} \bar{N}_m^{(n)} \frac{\partial N_m^{(n)}}{\partial \theta}$$

Apply the call spread regularisation formula, we can approximate

$$\bar{\tilde{H}}_m^{(n)} = \frac{\bar{\tilde{E}}_\tau^{(n)}}{2\delta} \mathbb{I}\left(\tilde{H}_m^{(n)} \in \left[\tilde{E}_m^{(n)} - \delta, \tilde{E}_m^{(n)} + \delta\right]\right)\left(P_m^{(n)} - Q_m^{(n)}\right)$$

$$\bar{\tilde{E}}_m^{(n)} = \bar{\tilde{E}}_\tau^{(n)} \left[R_m^{(n)} + \frac{\mathbb{I}\left(\tilde{E}_m^{(n)} \in \left[\tilde{H}_m^{(n)} - \delta, \tilde{H}_m^{(n)} + \delta\right]\right)}{2\delta} \left(P_m^{(n)} - Q_m^{(n)}\right) \right]$$

$$\bar{\tilde{N}}_m^{(n)} = -\bar{\tilde{E}}_\tau^{(n)} \frac{Q_m^{(n)} \mathbb{I}\left(\tilde{E}_m^{(n)} > \tilde{H}_m^{(n)}\right)}{\tilde{N}_m^{(n)}}$$

$$\bar{\tilde{E}}_\tau^{(n)} = \frac{\bar{V}(T_0)}{N_{MC}}$$

$P_m^{(n)}$, $Q_m^{(n)}$ and $R_m^{(n)}$

$$P_m^{(n)} = \begin{cases} \frac{\tilde{E}_2^{(n)} \mathbf{1}_{\tilde{E}_2^{(n)} > \tilde{H}_2^{(n)}}}{\tilde{N}_2^{(n)}} + \sum_{k=3}^{M} \frac{\tilde{E}_k^{(n)} \mathbf{1}_{\tilde{E}_k^{(n)} > \tilde{H}_k^{(n)}}}{\tilde{N}_k^{(n)}} \prod_{j=2}^{k-1} \mathbf{1}_{\tilde{H}_j^{(n)} > \tilde{E}_j^{(n)}}, & m = 1 \\ \sum_{k=m+1}^{M} \frac{\tilde{E}_k^{(n)} \mathbf{1}_{\tilde{E}_k^{(n)} > \tilde{H}_k^{(n)}}}{\tilde{N}_k^{(n)}} \prod_{j=2, j \neq m}^{k-1} \mathbf{1}_{\tilde{H}_j^{(n)} > \tilde{E}_j^{(n)}}, & m \in [2, M-1] \\ 0, & m = M \end{cases},$$

$$Q_m^{(n)} = \begin{cases} \frac{\tilde{E}_m^{(n)}}{\tilde{N}_m^{(n)}} \prod_{j=1}^{m-1} \mathbf{1}_{\tilde{H}_j^{(n)} > \tilde{E}_j^{(n)}}, & m \neq 1 \\ \frac{\tilde{E}_1^{(n)}}{\tilde{N}_1^{(n)}}, & m = 1 \end{cases},$$

$$R_m^{(n)} = \begin{cases} \frac{\mathbf{1}_{\tilde{E}_m^{(n)} > \tilde{H}_m^{(n)}}}{\tilde{N}_m^{(n)}} \prod_{j=1}^{m-1} \mathbf{1}_{\tilde{H}_j^{(n)} > \tilde{E}_j^{(n)}}, & m \neq 1 \\ \frac{\mathbf{1}_{\tilde{E}_1^{(n)} > \tilde{H}_1^{(n)}}}{\tilde{N}_1^{(n)}}, & m = 1 \end{cases}.$$

AAD for XVA Greeks

For $m = M-1, \dots, 0$, we collect

$$\bar{\theta} {+}{=} \sum_{n=1}^{N_{MC}} \bar{X}_{m+1}^{(n)} \frac{\partial F}{\partial \theta}(T_m, X_m^{(n)}, \theta)$$
$$\bar{\theta} {+}{=} \sum_{n=1}^{N_{MC}} \bar{\lambda}_{m+1}^{c,(n)} \frac{\partial G_c}{\partial \theta}(T_m, \lambda_m^{c,(n)}, \theta)$$
$$\bar{\theta} {+}{=} \sum_{n=1}^{N_{MC}} \bar{\lambda}_{m+1}^{d,(n)} \frac{\partial G_c}{\partial \theta}(T_m, \lambda_m^{d,(n)}, \theta)$$

$$\bar{X}_m^{(n)} = \bar{E}_m^{(n)} \frac{\partial E_m^{(n)}}{\partial X_m^{(n)}} + \bar{N}_m^{(n)} \frac{\partial N_m^{(n)}}{\partial X_m^{(n)}} + \bar{\psi}_p{}^{T}(X_{T_m}^{(n)}) \frac{\partial \psi_p}{\partial X_m^{(n)}}(X_{T_m}^{(n)})$$
$$\bar{\theta} = \sum_{n=1}^{N_{MC}} \bar{N}_m^{(n)} \frac{\partial N_m^{(n)}}{\partial \theta}$$
$$\bar{\lambda}_m^{c,(n)} = -\sum_{j=m+1}^{M} \overline{SP}_j^{c,(n)} SP_j^{c,(n)} (T_{j+1} - T_j)$$
$$\bar{\lambda}_m^{d,(n)} = -\sum_{j=m+1}^{M} \overline{SP}_j^{d,(n)} SP_j^{d,(n)} (T_{j+1} - T_j)$$

$$\bar{H}_{p,m}^{(n)} = \begin{cases} \bar{V}_{p,m}^{(n)} \, \mathbb{I}\left(H_{p,m}^{(n)} > E_{p,m}^{(n)}\right), & \text{if } T_m \in \mathcal{T}_p(T_0) \\ \bar{V}_{p,m}^{(n)}, & \text{otherwise} \end{cases}$$
$$\bar{E}_{p,m}^{(n)} = \begin{cases} \bar{V}_{p,m}^{(n)} \, \mathbb{I}\left(H_{p,m}^{(n)} < E_{p,m}^{(n)}\right), & \text{if } T_m \in \mathcal{T}_p(T_0) \\ 0, & \text{otherwise} \end{cases}$$

$$\bar{\beta}_m = \sum_{n=1}^{N_{MC}} \bar{H}_{p,m}^{(n)} \psi_p(X_{T_m}^{(n)})$$
$$\bar{\psi}_p\left(X_{T_m}^{(n)}\right) = \bar{H}_{p,m}^{(n)} \beta_m$$

$$\bar{V}_{p,m}^{(n)} = \bar{V}_m^{(n)}$$

$$\bar{V}_m^{(n)} = \frac{\overline{XVA}}{N_{MC} N_m^{(n)}} \left[\left(SP_{m-1}^{c,(n)} - SP_m^{c,(n)}\right) \mathbb{I}\left(V_m^{(n)} > 0\right) + \left(SP_{m-1}^{d,(n)} - SP_m^{d,(n)}\right) \mathbb{I}\left(V_m^{(n)} < 0\right)\right]$$
$$\bar{N}_m^{(n)} = -\frac{\overline{XVA}\, V_m^{(n)}}{N_{MC} \left(N_m^{(n)}\right)^2} \left[\left(SP_{m-1}^{c,(n)} - SP_m^{c,(n)}\right) \mathbb{I}\left(V_m^{(n)} > 0\right) + \left(SP_{m-1}^{d,(n)} - SP_m^{d,(n)}\right) \mathbb{I}\left(V_m^{(n)} < 0\right)\right]$$
$$\overline{SP}_m^{c,(n)} = \overline{XVA} \left[\frac{V_{m+1}^{(n)}}{N_{MC} N_{m+1}^{(n)}} - \frac{V_m^{(n)}}{N_{MC} N_m^{(n)}}\right] \mathbb{I}\left(V_m^{(n)} > 0\right) \text{ for } m \le M-1, \text{ and } -\frac{\overline{XVA}\, V_m^{(n)}}{N_{MC} N_m^{(n)}} \mathbb{I}\left(V_m^{(n)} > 0\right) \text{ for } m = M$$
$$\overline{SP}_m^{d,(n)} = \overline{XVA} \left[\frac{V_{m+1}^{(n)}}{N_{MC} N_{m+1}^{(n)}} - \frac{V_m^{(n)}}{N_{MC} N_m^{(n)}}\right] \mathbb{I}\left(V_m^{(n)} < 0\right) \text{ for } m \le M-1, \text{ and } -\frac{\overline{XVA}\, V_m^{(n)}}{N_{MC} N_m^{(n)}} \mathbb{I}\left(V_m^{(n)} < 0\right) \text{ for } m = M$$

Sensitivities in $\bar{\beta}_m$

For transferring the sensitivities in $\bar{\beta}_m$

$$\bar{V}_1^{(n)} = 0$$

Do

$$\bar{X}_M^{(n)} += \bar{V}_M^{(n)} \frac{\partial E_M^{(n)}}{\partial X_M^{(n)}}$$

For $m = M-1, \dots, 0$, we collect

$$\bar{\theta} += \sum_{n=1}^{N_{MC}} \bar{X}_{m+1}^{(n)} \frac{\partial F}{\partial \theta}(T_m, X_m^{(n)}, \theta)$$

for $m = 1, \dots, M$

$$\bar{X}_m^{(n)} += \bar{\psi}^T\left(X_m^{(n)}\right) \frac{\partial \psi}{\partial X_m^{(n)}}$$

$$\bar{\psi}\left(X_m^{(n)}\right) += \frac{1}{N_{MC}}\left[\left(\bar{\Psi}_m + \bar{\Psi}_m{}^T\right) \psi\left(X_m^{(n)}\right) + \mathcal{N}_m^{(i)} V_{m+1}^{(n)} \bar{\Omega}_m\right],$$

$$\bar{X}_m^{(n)} += \frac{\bar{\mathcal{N}}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_m^{(i)}}{\partial X_m^{(n)}}, \quad \bar{X}_{m+1}^{(n)} += -\frac{\bar{\mathcal{N}}_m^{(i)} \mathcal{N}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_{m+1}^{(i)}}{\partial X_{m+1}^{(n)}},$$

$$\bar{\theta} += \frac{\bar{\mathcal{N}}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_m^{(i)}}{\partial \theta} - \frac{\bar{\mathcal{N}}_m^{(i)} \mathcal{N}_m^{(i)}}{N_{m+1}^{(i)}} \frac{\partial N_{m+1}^{(i)}}{\partial \theta}.$$

$$\bar{\beta}_m += \sum_{n=1}^{N_{MC}} \bar{H}_m^{(n)} \psi(X_m^{(n)})$$

$$\bar{\psi}\left(X_m^{(n)}\right) = \bar{H}_m^{(n)} \beta_m$$

$$\bar{\Omega}_m = \Psi_m{}^{-1} \bar{\beta}_m$$

$$\bar{\Psi}_m = -\Omega_m \bar{\beta}_m{}^T$$

$$\mathcal{N}_m^{(i)} = \frac{N_m^{(i)}}{N_{m+1}^{(i)}},$$

$$\bar{\mathcal{N}}_m^{(i)} = \frac{\psi^T\left(X_m^{(n)}\right) V_{m+1}^{(n)} \bar{\Omega}_m}{N_{MC}}.$$

Either

$$\bar{E}_m^{(n)} = \bar{V}_m^{(n)}\, \mathbb{I}\left(H_m^{(n)} < E_m^{(n)}\right)$$

$$\bar{V}_{m+1}^{(n)} = \frac{\bar{V}_m^{(n)} N_m^{(i)}}{N N_{m+1}^{(i)}} \mathbb{I}\left(H_m^{(n)} > E_m^{(n)}\right)$$

$$\bar{N}_m^{(i)} = \mathbb{I}\left(H_m^{(n)} > E_m^{(n)}\right)\left[\bar{V}_m^{(n)} \frac{V_{m+1}^{(n)}}{N_m^{(n)}} - \bar{V}_{m-1}^{(n)} \frac{N_{m-1}^{(n)} V_m^{(n)}}{\left(N_m^{(n)}\right)^2}\right] \text{ for } 1 < m < M$$

$$\bar{N}_1^{(i)} = \mathbb{I}\left(H_1^{(n)} > E_1^{(n)}\right) \bar{V}_1^{(n)} \frac{V_2^{(n)}}{N_2^{(n)}},$$

$$\bar{N}_M^{(i)} = \mathbb{I}\left(H_M^{(n)} > E_M^{(n)}\right) - \bar{V}_{M-1}^{(n)} \frac{N_{M-1}^{(n)} V_M^{(n)}}{\left(N_M^{(n)}\right)^2}$$

$$\bar{E}_m^{(n)} = \bar{V}_m^{(n)}\, \mathbb{I}\left(H_m^{(n)} < E_m^{(n)}\right)$$

$$\bar{H}_m^{(n)} = \bar{V}_m^{(n)}\, \mathbb{I}\left(H_m^{(n)} > E_m^{(n)}\right)$$

Sensitivities in $\bar{\beta}_m$

For the Bermudan option low estimator, Capriotti, L., Jiang, Y., and Macrina, A. (2016) shows that, the sensitivities of model parameters transferred from $\bar{\beta}_m$ tend to zero when the exercising boundary (determined by β_m) approaches optimal.

It means for the Bermudan option low estimator, if the regression boundary is precise, the Greeks components in $\bar{\beta}_m$ can be negelected.

However, this is no longer true for XVA evaluations. Because in XVA exvaluation, we do not have any optimality to guarantee the Greeks components in $\bar{\beta}_m$ tends to be zero.

Measure the error of AAD Monte Carlo by binning

Though AAD gives us the value of the sensitivities, we do not have the Monte Carlo variance of our estimators for the sensitivities.

One time-consuming approach is to compute the sample variance by repeating the algorithm many times.

We divide N_{MC} paths into N_b equally sized "bins". Then we convert $\bar{V}(T_0)$ or $\overline{XVA}$ to the sensitivities for each bin. Hence for N_b sets of sensitivities, we can evaluate their sample mean $\hat{\mu}_{N_b}$ and variance $\hat{\sigma}_{N_b}$. Then we have the sensitivities Monte Carlo variance

$$\frac{\hat{\sigma}_{N_b}}{N_b}$$

.

Best of two Bermudan option

We price a Bermudan option on the maximum of two assets under the standard lognormal model. The payoff function is given by

$$(\max\{X_1(t), X_2(t)\} - K)^+ \tag{7}$$

where K is the strike price and t is the option executable time.

In particular, we choose $X_1(0) = X_2(0) = 1$, $\sigma_1 = \sigma_2 = 0.2$ and $d_1 = d_2 = 0.1$. We compute the Bermudan option prices, *Delta*s and *Vega*s for $K = 0.9$, 1.0 and 1.1 with risk-free rate $r = 0.05$, respectively.

We also compare these with their "exact" value, which is obtained by a PDE approach.

Bermudan Greeks: Monte Carlo AAD v.s. PDE

K = 0.9	Price	*Delta*	*Vega*
Exact value	0.20107	0.41423	0.45740
MC and AAD	0.20107	0.39338	0.51001
StD	0.00024	0.00495	0.00463
K = 1.0	Price	*Delta*	*Vega*
Exact value	0.13959	0.33588	0.48440
MC and AAD	0.13907	0.33249	0.49003
StD	0.00024	0.00201	0.00393
K = 1.1	Price	*Delta*	*Vega*
Exact value	0.09431	0.25635	0.46253
MC and AAD	0.09395	0.25925	0.46237
StD	0.00017	0.00138	0.00230

Table: Prices, *Delta*s and *Vega*s for the Bermudan option described in (7) with three different value of strikes. The smoothing component δ is chosen to be 0.005. The Monte Carlo paths for each set are 400,000. The standard deviation (short by StD) is computed via 20 bins for each set of simulations.

Sensitivities in $\bar{\beta}_m$: Bermudan options

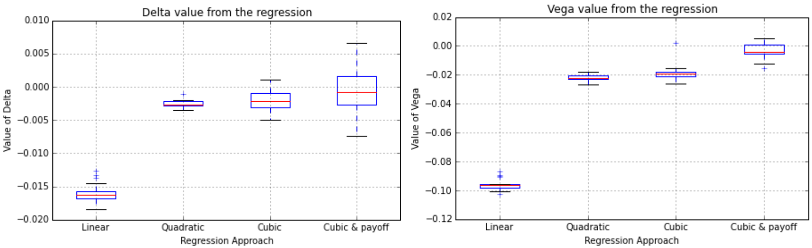


Figure: The *Delta* and *Vega* from the regression part against four regression functionals. The smoothing component δ is chosen to be 0.005. The sample mean and standard deviation are computed from 20 bins in a 400,000 paths Monte Carlo.

Smoothed v.s. Unsmoothed

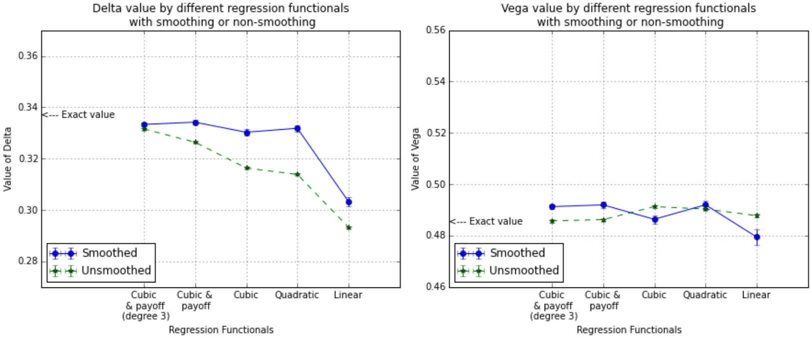


Figure: *Delta*s and *Vega*s by smoothing with call-spread approach and keeping unsmoothed against five regression functionals. The sample mean and standard deviation, depicted by the points and error-bars respectively, are computed from 20 bins in a 3,000,000 paths Monte Carlo. The boundaries for both sets are flexible. Two graphs are plotted under the same scale.

XVA sensitivities for Bermudan option

	σ_0	σ_1	σ_2	σ_3
AAD	0.00303	0.00305	0.00298	0.00287
StD	1.0e-4	4.6e-5	4.2e-5	5.7e-5
Bmp	0.00301	0.00304	0.00299	0.00286
StD	5.3e-5	4.7e-5	3.3e-5	3.4e-5
	σ_4	σ_5	σ_6	σ_7
AAD	0.00271	0.00255	0.00239	0.00220
StD	4.7e-5	5.6e-5	3.4e-5	5.0e-5
Bmp	0.00268	0.00254	0.00236	0.00220
StD	2.7e-5	3.0e-5	3.4e-5	2.6e-5
	σ_8	σ_9	σ_{10}	σ_{11}
AAD	0.00194	0.00184	0.00165	0.00153
StD	3.9e-5	4.1e-5	2.9e-5	3.2e-5
Bmp	0.00194	0.00183	0.00164	0.00151
StD	2.6e-5	2.1e-5	9.1e-6	3.8e-6

Table: The CVA sensitivities with respect to the piecewise volatility.

XVA sensitivities for Bermudan option

	λ_0	λ_1	λ_2	λ_3
AAD	0.0332	0.0332	0.0331	0.0330
StD	5.2e-5	8.4e-5	9.0e-5	1.2e-4
Bmp	0.0332	0.0332	0.0331	0.0330
StD	5.2e-5	8.0e-5	9.4e-5	1.2e-4
	λ_4	λ_5	λ_6	λ_7
AAD	0.0326	0.0320	0.0314	0.0306
StD	1.7e-4	1.6e-4	1.3e-4	1.6e-4
Bmp	0.0326	0.0320	0.0314	0.0306
StD	1.7e-4	1.6e-4	1.3e-4	1.6e-4
	λ_8	λ_9	λ_{10}	λ_{11}
AAD	0.0297	0.0288	0.0277	0.0265
StD	2.0e-4	2.2e-4	2.2e-4	2.2e-4
Bmp	0.0297	0.0288	0.0277	0.0265
StD	2.0e-4	2.2e-4	2.2e-4	2.2e-4

Table: The CVA sensitivities with respect to the piecewise hazard rates.

Sensitivities in $\bar{\beta}_m$: XVA

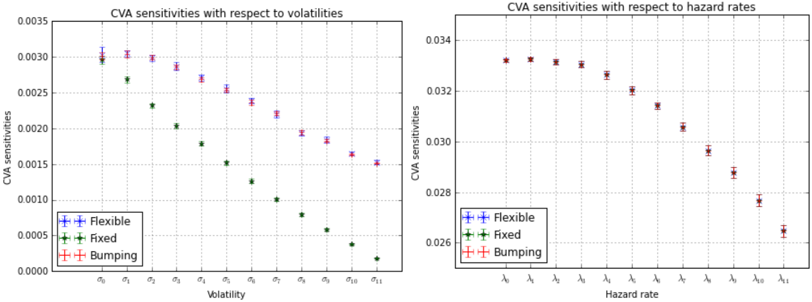


Figure: The graph of the CVA sensitivities with respect to the piecewise volatility and hazard rate, computed by AAD with flexible boundaries, fixed boundaries and bumping. Three sets of the results are all computed with a 50,000 paths Monte Carlo, where the standard deviations are computed with 10 bins.

CPU ratio: AAD v.s. Bumping

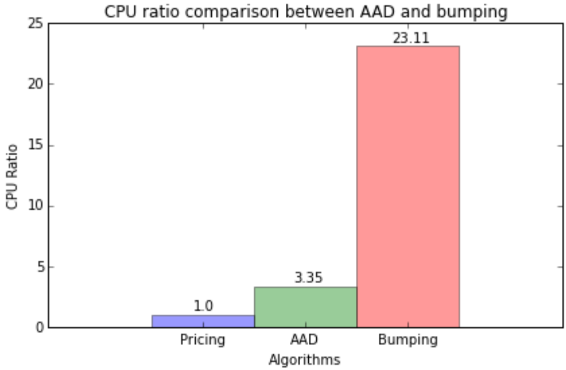


Figure: The CPU ratio of AAD and bumping for the CVA sensitivities computation, where the time consumption of the pricing algorithm is set as basis one.

Reference

Capriotti, L., Jiang, Y., and Macrina, A. (2016). AAD and Regression Monte Carlo: Fast Bermudan and XVA Greeks. UCL working paper.

Capriotti, L. and Giles, M. (2010b). Fast correlation risk by adjoint algorithmic differentiation. Risk, 23:79–85.

Capriotti, L., Jiang, Y., and Macrina, A. (2015). Real-time risk management: An AAD-PDE approach. International Journal of Financial Engineering, 02(04).

Giles, M. B. (2008). Collected matrix derivative results for forward and reverse mode algorithmic differenti- ation. In Advances in Automatic Differentiation, pages 35–44.

Glasserman, P. (2004). Monte Carlo Methods in Financial Engineering. Springer, New York.

Griewank, A. (2000). Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation. Frontiers in Applied Mathematics, Philadelphia.