Vol. 12(1), pp. 1-9, February 2019
DOI: 10.5897/AJMCSR2018.0754
Article Number: 704794060293
ISSN: 2006-9731

http://www.academicjournals.org/AJMCSR

ACADEMIC JOURNALS

African Journal of Mathematics and Computer Science Research

Review

Comparative study of the direct and inverse finite element methods for pricing American options

Bolujo Joseph Adegboyegun

Department of Mathematics, Faculty of Science, Ekiti State University, P. M. B. 5363, Ado-Iworoko Road, Ekiti State, Nigeria.

Received 16 July, 2018; Accepted 5 February 2019

This paper investigates the computational performance of direct and inverse finite element methods for pricing American options. The underlying concept of the direct approach is similar to that of conventional finite element method. But the inverse approach is a relatively new development that involves trading the roles of financial variables. Based on the same constitutive model and linear elements, a performance analysis of the two approaches is carried out against the benchmark solution. Furthermore, we present experimental results on their accuracy-efficiency trade-off. Results indicate that although both approaches possess good convergence to the benchmark result, the inverse method is more efficient in term of the acceptable computing time and accuracy.

Key words: Direct finite elements, Inverse finite elements, American-style options, Black-Scholes model.

INTRODUCTION

Derivative securities are financial instruments that derive their values from the performance of an underlying entity (asset, interest rate or index). Options are the most common derivative securities that frequently traded in financial market. An option is a financial contract that gives the holder a right, but not an obligation, to buy or sell a certain amount of a specified asset (the underlying asset) at a predetermined price (exercise price) on (European option) or before (American option) a prescribed future date (expiration date). While a closed-form solution for European options can be found using the Black-Scholes formula, pricing American option is much more challenging because of the nonlinearity associated with the early exercise policy. One popular way to proceed is to solve the option problems using PDE-based numerical techniques such as the finite element method (Zhang et al., 2015; Arregui et al., 2017).

The finite element method (FEM), just like the other PDE-based numerical methods for pricing American options, whilst appealing, suffers from the slow rate of convergence. The method requires intensive computation before a solution of reasonable accuracy can be obtained.

*Corresponding author. E-mail: bja998@uowmail.edu.au.

AMS subject classification: 91G20; 91G30; 91G80

Moreover, it is naturally difficult for the traditional FEM to accurately calculate the American option price near maturity because of its singular behaviour. Nevertheless, FEM can be made more efficient by considering its inverse formulation (inverse finite elements). The inverse finite element method (iFEM) is a numerical approach in which an optimization algorithm is coupled with finite element method to find optimal values for a set of target parameters which enter the finite element simulation (Chemisky et al., 2015). A user defined objective function serves to measure the optimality of the parameters. Studies involve the use of inverse finite element method (iFEM) are much more limited in the literature. The approach was initially used by Alexandrou (1989) in solving nonlinear problems associated with phase change, and with solidification. A characteristic feature of such a nonlinear problem is a demarcation line which separates two domains with different material properties. Comparably, American options have free boundary that separate the region where it is optimal to hold the option from where exercise is optimal.

The essential concept of the iFEM is to find the location at which, the dependent variable has a predefined value. In other words, while the dependent variable is fixed, solution is obtained for the independent variable without inverting the equations. In the conventional finite element method, the question 'what is the option price at a specific location (the nodes of the elements)?' is addressed. In contrast, the iFEM addresses the question 'at what location (the nodes of the elements) does the option has a specific value?'. The overall purpose of this paper is to investigate the advantages of switching the roles of dependent and independent financial variables in numerical valuation. Specifically, we compare the computational efficiency by carrying out a critical performance analysis of the two approaches against some benchmark solutions. The results of comparison as well as experimental results on their accuracy-efficiency trade-off are presented.

Governing equation and boundary conditions

To compare the computational performance of the dFEM and iFEM, we adopt the simple Black-Scholes model for an American put option without the dividend yield. The choice of this model allows the evaluation of our results within a framework that permits objective comparison with the existing solutions. Let $P(S,t)$ denote the value of an American put option, with *S* being the price of the underlying asset and *t* is the current time. Under the Black and Scholes (1973) framework, the differential system that governs the price of an American put option can be written as:

$$\frac{\partial P}{\partial t}+\frac{1}{2}\sigma^2S^2\frac{\partial^2P}{\partial S^2}+rS\frac{\partial P}{\partial S}-rP=0,$$

$$\begin{aligned}&P(S,T)=\max(K-S,0),\\&P(S_f(t),t)=K-S_f(t),\\&\frac{\partial P}{\partial S}(S_f(t),t)=-1,\\&lim_{S\to\infty}P(S,t)=0,\end{aligned}\tag{1}$$

in which r is the risk-free interest rate, σ is the volatility, K is the strike price, and T is the expiration time. Equation 1 is defined on $S\in[S_f(t)+\infty), t\in[0,T]$, where $S_f(t)$ is the optimal exercise boundary. The value of $S_f(t)$ is a priori unknown and needs to be determined as part of the solution of the problem. At $t=T$, it has been established that $S_f(T)=K$.

To facilitate the development of the algorithms, we first introduce the dimensionless independent variables x,τ in place of S and t, respectively, and new dependent variables $u(x,\tau), x_f(\tau)$ in place of $P(S,t), S_f(t)$ as:

$$x=ln\frac{S}{K},\ x_f(\tau)=ln\frac{S_f(t)}{K},\ u=\frac{P+S}{K}-1, and\ \tau=\frac{\sigma^2(T-t)}{2}$$

With the new variables, Equation 1 becomes a dimensionless system, which includes a governing differential equation together with the following corresponding initial and boundary conditions

$$\begin{aligned}&\frac{\partial u}{\partial\tau}-\frac{\partial^2u}{\partial x^2}+(1-\gamma)\frac{\partial u}{\partial x}+\gamma u+\gamma=0,\\&u(x,0)=max(e^x-1,0),\\&u(x_f(\tau),\tau)=0,\\&\frac{\partial u}{\partial x}(x_f(\tau),\tau)=0,\\&lim_{x\to\infty}u(x,\tau)=e^x-1,\end{aligned}\tag{2}$$

where B is defined on $x_f(\tau)\le x<+\infty, 0\le\tau\le\frac{\sigma^2T}{2}$. The parameter, γ is the dimensionless interest rate, and is related to the original risk-free interest rate by γ = $\frac{2r}{\sigma^2}$. Note that due to the introduction of the time to expiration τ as the difference between the expiration time, T and the current time, t, the terminal condition in Equation 1 has become an initial condition in Equation 2. Moreover, since the optimal exercise price, $S_f(t)$, is equal to the strike price, K at the expiration time, T, using the above transformed variable, we must have $x_f(0)$ = 0.

For computational purposes, the common practice in the literature is to truncate the semi-infinite domain [$x_f(\tau)$, +∞) to a finite interval Ω = [$x_{\min}$, $x_{\max}$]. While for a large

price of the underlying asset, the option value is negligible and is taken to be zero. Then, it is reasonable to truncate the pricing domain into a bounded domain complemented with appropriate boundary conditions. A considerable body of research has demonstrated that the upper bound of the underlying price $S_{\max}$ is three or four times of the strike price, it is reasonable to set $x_{\max} = ln\,5$. On the other hand, since $u(x,\tau = 0$ for x ≤ $x_f(\tau)$, there is no need to show what exactly xmin is. However, for symmetric purposes, some published works set [$x_{\min}$, $x_{\max}$]. This is, however, not the case for the iFEM where $x_{\min}$ is set to zero because we only focus on the positive region. The reasons for this choice are explained further in the subsequent section. What follow is the implementation of the two approaches using the set up in Equation 2.

Formulation of the numerical techniques

Here, presents the dFEM and iFEM implementation of the solution of the nonlinear system of Equation 2. The underlying idea behind the dFEM is similar to that of conventional finite element method. However, for ease of reference, it is briefly outlined. On the other hand, the iFEM is relatively new development and can have applications elsewhere. This method would be extended to other pricing formulations and models in the subsequent papers.

The direct finite element approach (dFEM)

Following the standard Galerkin weighted residue formulation (Rao, 2017), a residual equation is constructed by adopting $v(x)$ as the weighting function. The weighted residual or equilibrium statement for the governing differential equation in 2 reads

$$R = \int_\Omega \left[\frac{\partial u}{\partial \tau} - \frac{\partial^2 u}{\partial x^2} + (1-\gamma)\frac{\partial u}{\partial x} + \gamma u + \gamma \right] v\, dx = 0, \tag{3}$$

where $\Omega = [x_{\min}, x_{\max}]$.

In order to reduce the regularity condition on the option price, u, we integrate Equation 3 by parts via divergence theorem, to obtain

$$R = \int_\Omega \left[\frac{v\partial u}{\partial \tau} - \frac{\partial v}{\partial x}\frac{\partial u}{\partial x} + (1-\gamma)\frac{v\partial u}{\partial x} + v\gamma u + v\gamma \right] dx = 0 \tag{4}$$

Next, we define the solution u in terms of the basis function φ_i and time-dependent coefficient w_i(τ). Similarly, the weighting function v is written in terms of φ_i and an arbitrary constant α_i:

$$u \approx \sum_{i=0}^{N} w_i(\tau)\,\varphi_i(x), \quad v \approx \sum_{i=0}^{N} \alpha_i\,\varphi_i(x), \tag{5}$$

where w_i are some unknown time-dependent coefficients to be determined.

Subsequently, we derive the system of nonlinear ordinary differential equation which yields the semi-discrete solution u. With approximation (5), Equation 4 reads:

$$R = \frac{dw}{d\tau}A + w(\tau)B + F = 0, \tag{6}$$

where

$$A = \int_\Omega \varphi_i(x)\varphi_j(x)\,dx,$$

$$B = \int_\Omega \left(\varphi'_i(x)\varphi'_j(x) - (\gamma-1)\varphi'_i(x)\varphi_j(x) + \gamma\varphi_i(x)\varphi_j(x)\right) dx,$$

and

$$F = \int_\Omega \gamma\varphi_j(x)\,dx,$$

with the solution vector

$w := (w_1, \ldots, w_N)^T$ and $\frac{dw}{d\tau} = \dot{w} := (\dot{w_1}, \ldots, \dot{w_N})^T$.

To complete the discretization of (6) as a fully discretized version of (4), we approximate the time derivative appearing in (6) with a standard finite difference scheme. The time interval is decomposed into equidistant points $0 =: \tau^0 < \tau^1 \ldots < \tau^M := T$ with time step $\Delta\tau = \tau^k - \tau^{k-1}$. The finite difference approximation for the $\frac{dw}{d\tau}$ at time τ is

$$\frac{dw}{d\tau} = \frac{w^{(n+1)} - w^{(n)}}{\Delta\tau} \tag{7}$$

By using Equation 7, Equation 6 yields the familiar θ-scheme for $\theta \in [0,1]$

$$R = Kw^{(n+1)} - \bar{K}w^{(n)} + F\Delta\tau = 0, \tag{8}$$

where $\bar{K} = A - \bar{B}\theta\Delta\tau$, $\mathrm{K} = \mathrm{A} + \mathrm{B}\theta\Delta\tau$ and $\bar{\theta} = 1 - \theta$. Choosing $\theta = 1$ yields an explicit Euler's scheme while $\theta = 0$ yields an implicit Euler scheme. $\theta = \frac{1}{2}$ corresponds to the Crank-Nicolson scheme. Here, we select the latter due to its consistency and convergence error rate $O(\Delta\tau^2)$.

Finally, with $= w^{(n+1)}$, $Q = K\bar{w}^{(n)} - F\Delta\tau$ and after specifying the appropriate time-dependent boundary underlying boundary conditions, we obtain a non-singular system of algebraic equations:

$$R = KW - Q = 0, \tag{9}$$

where W is the vector of the unknown nodal values of the entire domain, K and $\bar{K}$ are the constrained master stiffness matrices and F is the master column matrix. Note that the nonlinear system of Equation 9 is to be

solved forward in time, that is, the terms indexed by n are known, while the terms with index $n + 1$ are to be determined. For numerical computation, we evaluate the element matrices using a linear basis function and assemble all the element matrices to obtain *K* and *Q*.

Next, we solve for the unknown *W* as the zeros of Equation 9 using the popular PSOR algorithm. Solving problems of Equation 9 is still a difficult task, because the terms *K* and *Q* are both functions of the unknown boundary after imposing the constrained boundary conditions making the system nonlinear. The coupling of the two types of unknowns (the optimal exercise price and the option values) makes Equation 9 much more computational challenging.

Inverse finite element approach (iFEM)

As mentioned earlier, the iFEM was used by Alexandrou (1989) in solving nonlinear problems associated with phase change in mechanics. However, in quantitative finance, to the best of our knowledge, the current literatures are a paper by Zhu and Chen (2013) and recently by Adegboyegun (2018). Based on an algorithm proposed Alexandrou (1989) and Zhu and Chen (2013) detailed a numerical scheme for locating the optimal exercise boundary for American put options with no dividend yield. The approach involves the use of simulated finite elements to inversely predict desired quantities that are spatially varying with time. Any assumptions included in the finite element model and in the whole simulation of the experiment determine the quality of the inverse solution (Chemisky et al., 2015). In quantitative finance, iFEM is employed by craving the boundaries of the finite elements to remain on "isotherms" of the asset price, whereas, the option value is specified a priori everywhere in the domain. Thus, the option is constant along the boundaries of unknown locations, which are permitted to change as the adopted optimization algorithm (Newton iteration method) proceeds. In this way, the Neumann boundary condition in the PDE system (2) is satisfied.

Furthermore, since $x_f\ (0) = 0$ we must have $\max(e^x - 1, 0) \geq 0$ when x is in the range $x_f \leq x \leq +\infty$. Consequently, the initial condition in Equation 9 can be simplified as $u(x, 0) = e^x - 1$. To realistically implement the approach, the range of option price, *P* must be known a *priori*. However, in Equation 1, it is not difficult to show that *P* would fall within $[0, K - S_f(t)]$, which varies with respect to time. However, the difficulty is overcome in Equation 2. After introducing the dimensionless variable, the transformed option price, u falls within $[0, e^x - 1]$, in which the unknown boundary is removed.

To ensure a reasonably accurate solution, we show that *u* is a strictly monotonically increasing function with respect to x for $x \in (x_f, +\infty)$. As previously shown by Zhu and Chen (2013), this is done by evaluating $\frac{\partial u}{\partial x} = \left(\frac{\partial P}{\partial S} + 1\right)\frac{S}{K}$. Here, $\frac{\partial u}{\partial x}$ is greater than zero because the delta of an American put option is more than −1 for $S \in (S_f, +\infty)$. Thus, u is strictly monotonically increasing with respect to x for $x \in (x_f, +\infty)$.

Next, we proceed to the detail implementation of iFEM. The first step is to deal with the time derivative appearing in the governing partial differential equation. In contrast to the conventional FEM, where $\frac{\partial u}{\partial \tau}$ is approximated by difference scheme, here, it is decomposed into the hedge parameter, delta, that is, $\frac{\partial u}{\partial x}$ and the velocity of the mesh, $\frac{\partial u}{\partial \tau}$. Now, according to the concept of the iFEM, the option price, u is obtained at selected underlying price which varies with respect to time, and therefore, we obtain

$$\frac{du}{d\tau} = \frac{\partial u}{\partial \tau} + \frac{\partial u}{\partial x}\frac{\partial u}{\partial \tau} \tag{10}$$

where $\frac{du}{d\tau}$ is a total derivative, that is, the rate of change of the option price at a node. Recall that the option price is distributed and kept constant at the computational nodes. Hence, $\frac{du}{d\tau} = 0$ Moreover, it is obvious that in this case the mesh is not fixed but moves with velocity $\text{mesh} = \frac{dx}{d\tau}$. Therefore,

$$\frac{\partial u}{\partial \tau} = -\frac{\partial u}{\partial x} V\text{mesh} \tag{11}$$

Using Equation 2 and following the conventional finite element formulation, residual equation can be constructed for the governing PDE in Equation 2 as:

$$R(x) = \int_0^{x_{max}} \left(\frac{\partial^2 u}{\partial x^2} + (\gamma - 1 + V\text{mesh})\frac{\partial u}{\partial x} - \gamma u - \gamma\right) v dx = 0 \tag{12}$$

With the velocity of the mesh being approximated by the first order finite difference $V\text{mesh} \approx Px = \frac{x_{\tau+\Delta\tau} - x_\tau}{\Delta\tau}$, and by integrating by parts, Equation 12 reduces to:

$$\bar{R}(x) = \int_0^{x_{max}} \left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + (\gamma - 1 + Px)v\frac{\partial u}{\partial x} + \gamma v u + v\gamma\right) dx = 0, \tag{13}$$

where x_{max} is the location of the last node (the limit of the yielded domain). This limit (which is indeed a key parameter) is obtained as a function of time automatically with the solution. The iFEM is implemented by considering that the option price in the yielded part varies from $u = 0$ at the rotating surface $x = 0$ (that is, location of the free boundary at expiry) to $u = e^{x_{max}} - 1$ at $x = x_{max}$.

At this stage, three advantages of iFEM are obvious:

(1) The solution is limited to the yielded part of the option,

the singularity is automatically removed, and hence the solution corresponds to the ideal constitutive model without any regularization.
(2) The boundary conditions are applied and satisfied exactly.
(3) There is a reduction of the total number of unknowns due to a priori known option values of designated underlying asset.

Remarks

*V*mesh is approximated by Px, and thus, the original equilibrium statement $R(x)$ in Equation 12 is replaced by $\bar{R}(x)$ in Equation 13. This is because of the truncation error brought in by the numerical approximation of *V*mesh by first order finite difference. The error can be reduced by adopting higher-order approximation method. For simplicity, we have adopted the first order approximation in the current work while the implementation with higher order approximation should be similar. The remaining part of the iFEM formulation involves the selection of suitable shape functions, the computation of the element matrices and assembling of the finite element contributions all follow the dFEM procedures. Finally, after specifying the appropriate time-dependent underlying boundary conditions, we obtain a non-singular system of algebraic equations of the form:

$$\bar{R} = K * W * - Q *, \tag{14}$$

where *W** are the nodal values of the entire domain, *K** and *Q** are respectively, the constrained master stiffness matrix and the constrained master column matrix and they are given by:

$K *_{i,j} = \int_{\Omega} \left(\varphi'_i(x)\varphi'_j(x) + (1-\gamma-Px)\varphi'_i(x)\varphi_j(x) + \gamma\varphi_i(x)\varphi_j(x)\right) dx$, and
$Q *_{i,j} = \int_{\Omega} \gamma\varphi_j(x)\, dx.$

The resulting non-linear system of Equation 14 is solved using a Newton-Raphson scheme with its quadratic convergence characteristic. We remark that the structure of the nonlinear Equation 14 is different from that of dFEM Equation 9. While *W* in Equation 9 are unknown nodal values, *W** in Equation 14 are kept as known constant values along the nodes of unknown location. The Equation 14 formulation results to the elimination of the requirement to specify the spacing of the ‘‘isotherms” of the underlying along the moving boundary, which is replaced by the specification of the option value. For a reasonably accurate solution, the monotonicity of *W** is required. Otherwise, the coordinate *x* that satisfies Equation 14 is not unique. This will lead to difficulties in deciding the correct location for a fixed nodal value, even if the convergence of the adopted iteration method is guaranteed. Fortunately, in our case, *W** are strictly monotonically increasing with respect to *x*, as demonstrated earlier. Therefore, no such problem needs to be further considered.

For numerical computation, the Jacobian of the Newton-Raphson procedure is saved using an element-by-element storage and solved by an iterative method based on a modification of the biconjugate gradient stabilized method. The Jacobi preconditioning was used to speed up convergence and the derivatives of the residual equations $\bar{R}(x)$ are obtained with respect to the unknown nodal locations *x*. For converged results, usually two to three iterations in the Newton-Raphson procedure are necessary at each time step and the solution advances to the next time step when all unknowns converge to the stopping criterion set to a relative error of 10^{-7}. An algorithm that guarantees the convergence of iFEM is proposed by Zhu and Chen (2013) and is summarized as follows:

(i) At the zeroth time step, the nodal location x_0 is initialized as $x_0 = [a_1 \ldots a_{N+1}]$, where $a_1 = x_{min}$, $a_{N+1} = x_{\max}$, and $a_{i-1} < a_i < a_{i+1}\ (2 < \mathrm{i} < \mathrm{N})$, with N being the number of elements in the whole computational domain.
(ii) The Newton iteration scheme is then adopted to find the exact nodal location of the *k*th $(k > 1)$ time step, that is, x_k^*. The initial guess of the solution is set as the final solution of the (k − 1) th time step, that is, $x_k^0 = x_{k-1}^*$
The specific implementation of the Newton iteration for this time step is a follow:
(iii) Suppose that x_k^n is obtained after nth iteration $(n \geq 0)$, we compute the residual $\bar{R}(x_k^n)$ through (14), and the corresponding Jacobian matrix $J_{\bar{R}(x_k^n)}$
(iv) Calculate the unknown nodal locations at the $(n+1)th$ iteration step through
$x_k^{n+1} = x_k^n - J^{-1}_{\bar{R}(x_k^n)}$.
(v) Repeat steps 1 and 2 until $\| x_k^{n+1} - x_k^n \| < \epsilon$ is satisfied. Set the solution of the kth time step to $x_k^* = x_{k-1}^{n+1}$, which completes the Newton iteration for the kth time step.

In the above algorithm, the location of the fixed boundary is excluded, since it is already the solution of the corresponding nodal value, and no iteration is further needed. If the location of the fixed boundary were still taken into consideration, the residual associated with this point would be zero, resulting in the corresponding row of the Jacobian matrix being zero. Consequently, the Jacobian matrix would be highly singular, and the Newton iteration fails. The convergence of the above scheme was discussed recently by Zhu and Chen (2013) and several other literatures, hence, the aspect is left out in this work.

Numerical experiments

Here, we report the results of numerical experiments and some detailed comparisons are made between the dFEM

Table 1. The variation of *RMSRE* when the grid sizes are gradually increased. *M*: the number of time intervals; *N*: the number of elements.

	M=10		***M=20***		***M=40***		***M=80***	
	dFEM	**iFEM**	**dFEM**	**iFEM**	**dFEM**	**iFEM**	**dFEM**	**iFEM**
N=5	0.1045	0.0775	0.0741	0.0529	0.0486	0.0369	0.0306	0.0266
N=10	0.0917	0.0725	0.0592	0.0477	0.0402	0.0310	0.0287	0.0201
N=25	0.0872	0.0688	0.0574	0.0440	0.0358	0.0266	0.0214	0.0153
N=50	0.0748	0.0688	0.0531	0.0427	0.0322	0.0251	0.0193	0.0137

and iFEM for pricing American options. To provide a fair and meaningful comparison, linear basis function is used for each of the discretization. Evaluation of the results is conducted with Zhu (2006) analytical solution as a benchmark. Using this equivalent set up, the goal is to compare the two methods in terms of computational performance. In order to facilitate objective comparison, we conduct the experiments on the examples presented by Zhu (2006) and Zhu and Chen (2013). The parameters used are: the strike price $K = 100$, the risk-free interest rate $r = 10\%$, the volatility of the underlying asset $\sigma = 30\%$ and the tenor of the contract being $T = 1$ year.

To compare the results of the two approaches, we focus mainly on the comparisons based on the optimal exercise prices, $S_f(t)$ instead of the option value, *P* since $S_f(t)$ is more difficult to be accurately calculated than the option price. Once $S_f(t)$ is accurately determined, the pricing problem becomes a fixed boundary problem and the calculation of the option price is straightforward.

Comparison in terms of accuracy

For us to compare the solution accuracy of dFEM and iFEM with respect to various levels of discretization and the number of time intervals, we use the RMSRE (root mean square relative error), which is defined as:

$$RMSRE = \sqrt{\frac{1}{I}\sum_{i=1}^{I}\left(\frac{a_i - \bar{a}_i}{a_i}\right)^2},$$

where $\bar{a}_i's$ are the nodal values of S_f associated with dFEM and iFEM, $a_i's$ are the S_f obtained from the Zhu's analytical result and *I* is the number of sample points used in the RMSRE. With the RMSRE, comparison of the overall difference of the computed numerical results and the exact solution based on Zhu's analytical result can be clearly demonstrated. In our numerical experiments, *I* was set to be 50 in all the results presented. In order to have a good comparison of the error associated with each method, the RMSRE when the number of steps in both spatial and temporal directions are gradually increased are tabulated respectively in Table 1. From this table, one can clearly see that the dFEM produces consistently larger error than the iFEM when equal size element and number of time interval are used. The results suggest that iFEM yields a more accurate result than dFEM. This clear difference may relate to the different convergence schedules; the adopted PSOR scheme in dFEM has slow convergence, whereas, the full Newton iterative scheme adopted in iFEM has a quadratic convergence rate. Moreover, a careful observation of the table shows that the differences between the RMSRE of the two methods on a coarse grid resolution, say *N = 5* and *M = 10* is quite substantial, but the difference is not well pronounced on a relatively fine resolution of *N = 50* and *M = 80*. A reason adduced to this observation is that fine grid resolution produces better results, and hence, less difference in RMSRE. One should also notice from Table 1 that when the time interval parameter *M*, reduces (corresponding to an increase in the size of the time step), the *RMSREs* for both methods become larger.

Another important observation is that a reduced time interval worsened the convergence conditions of the adopted *PSOR* and Newton iterative schemes for both dFEM and iFEM, respectively. This is as a result of large discretization errors when dealing with the time derivative, $\frac{dW}{d\tau}$, and the velocity of the mesh, Vmesh associated with the dFEM and iFEM schemes, respectively. A reduced number of elements also produces large *RMSREs* for the two methods. The error in this case, however, relates to the finite element discretization acting on the residual equation. Note that in all the computations, linear basis function is used. Numerical solutions based on quadratic shape function would have a smaller *RMSREs* than those computed on linear function.

Having compared the dFEM and iFEM based on the variation of *RMSRE* when the grid sizes are gradually increased, it is also important to compare the accuracy of both approaches based on a fine resolution of a grid size. Such a comparison is presented in Figure 1 with a grid resolution of $N = 100$ and $M = 100$. As shown in Figure 1, it can be easily seen that although both numerical results show a good convergence to the Zhu's analytical results, the results based on iFEM better approximate the

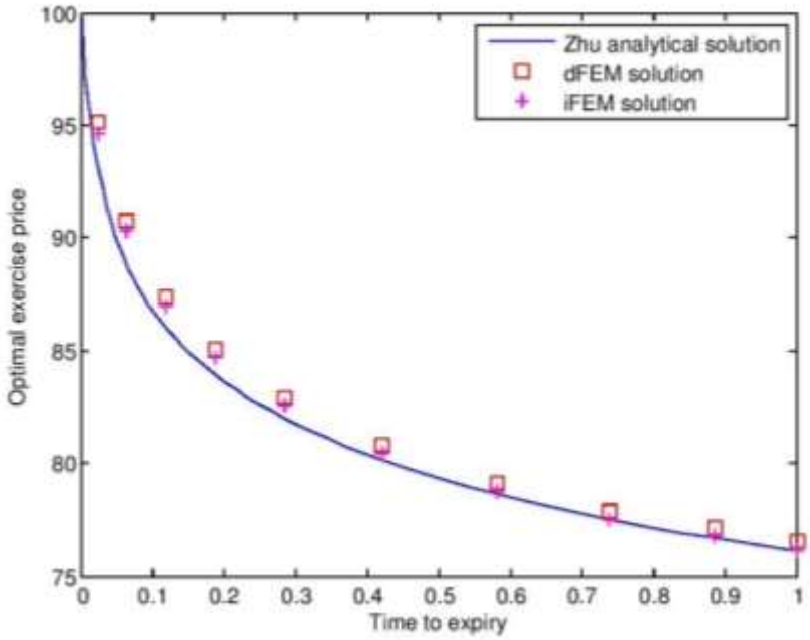

Figure 1. Comparison of Sf for dFEM and iFEM.

benchmark solution. A close examination of Figure 1 reveals that the iFEM almost coincide with the benchmark solution at the expiration date, $t = T = 1$ (year), the optimal exercise price calculated by Zhu's analytical method is $S_f(T)$= 76.113, whereas, they are $S_f(T)$ = 76.245 and $S_f(T)$= 76.804. Furthermore, Figure 1 also reveals that the dFEM underestimate the free boundary values when the time close to expiry. This is due to the presence of singularity at expiry, which is not possible for most of the numerical algorithms to deal with. However, in the case of iFEM, the algorithm is designed such that the location of the optimal exercise price at expiry is known a priori and is already included in the algorithm.

Comparison in terms of efficiency

There have been two thrusts in the development of algorithms as far as real-world tasks are concerned. One has emphasized higher accuracy; the other faster implementation (Uijlings et al., 2015). These two thrusts, however, have been independently pursued, without addressing the accuracy versus efficiency trade-offs. The importance of accuracy of an algorithm diminishes when response time is slow for a given task. The converse is also true; importance of a fast algorithm diminishes if the accuracy and precision are insufficient for subsequent financial interpretations. Comparing the dFEM and iFEM in terms of computational performance is not an easy task. Although the accuracy-efficiency characteristic is algorithm-dependent, an understanding of a general pattern is crucial in evaluating algorithm performance as far as real-world tasks are concerned.

As expected, in our numerical experiments, the computing time for both methods increases with increase in the grid size. However, the dFEM incur less computational cost than the iFEM under the same grid resolutions. In fact, for the grid resolution *M = N = 100*, the computational cost for the dFEM is just 48 s, whereas, it takes iFEM more than 120 seconds for the same grid resolution. But again, this nice feature does not make the dFEM more efficient than iFEM, because efficiency of an algorithm does not depend only on the speed of calculation, but also on the accuracy. The task of establishing a "trade-off" between accuracy and efficiency shall be our goal subsequently.

Accuracy versus efficiency

In our discussions pertaining to accuracy and speed, all the illustrative results are based on linear basis function. A 2-D curve characterizing the accuracy-efficiency (AE) trade-off is used to evaluate the performance of the methods. On the curve, accuracy (abscissae) is measured by the RMSRE, (calculated using the Zhu (2006) analytical result as the base value), and computational efficiency (ordinate) is measured by the total CPU time consumed at each run. These curves are generated by setting parameter of the algorithms in the temporal direction to a fixed value, *M = 25*, while varying resolution

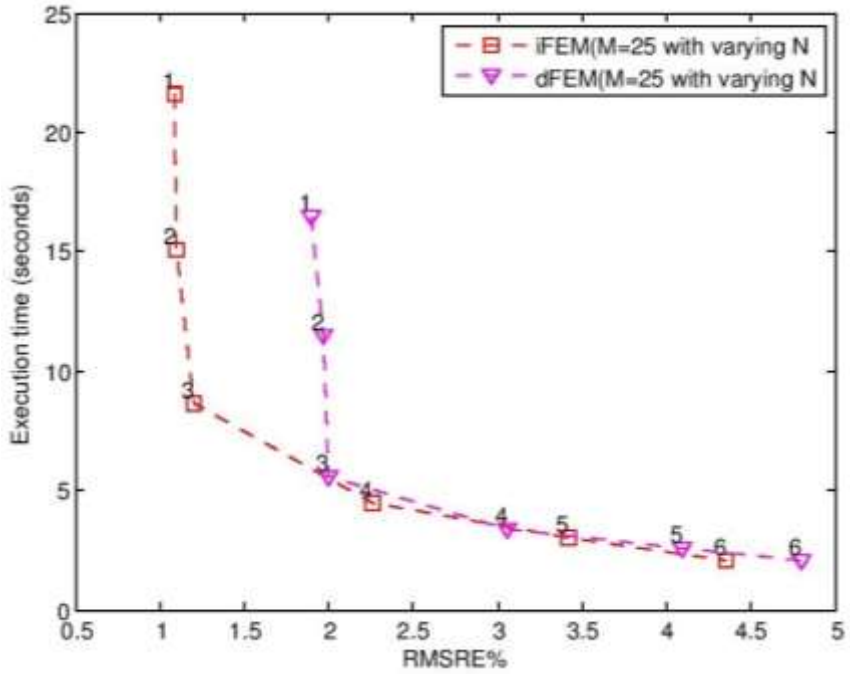

Figure 2. Accuracy versus efficiency.

in the spatial direction. Note that similar curve is obtained when grid resolution in spatial direction is fixed and the resolution in temporal direction is varied.

In the AE family of performance curves depicted in Figure 2, six different resolutions were used in the computation. Each curve corresponds to the iFEM and dFEM algorithm, respectively. A point on the performance curve denotes a certain parameter setting (grid resolution). As clearly shown, the accuracy is inversely varying with the speed of calculation for the two methods (curves); an expected result. A higher accuracy usually implies a lower run time and verse-versa for any resolution. It can also be easily observed that the dFEM curve shows a greater speed of calculation but with larger error, whereas iFEM has significantly reduced error with higher computing time under equivalent grid resolution as the dFEM. Following the explanation Cuadrado et al. (2001), the distance from the origin to AE curve represents the overall performance (efficiency) of the algorithm. Performance point close to the origin (small error and low execution time) is indicative of better algorithm operating point. In terms of AE performance, iFEM appears more flexible and effective, because the curve is closer to the origin in about five out of the six resolutions (from the 2nd−6th). A close examination of the curves shows that at any point between the 3rd to 6th resolution, iFEM curve is closer to the origin than dFEM curve. A nice feature of the iFEM as indicated in AE curve is that using any point on the curve between the 3rd and 4th resolutions, which appear to be the closest region on the iFEM curve to the origin, a high computational performance (efficiency) in terms of a satisfactory computing time and accuracy is achieved.

The AE curve is also useful in determining the computational cost when the same order of accuracy is maintained. For example, let us consider RMSRE = 2.5% on the curves where errors for the two methods exist. It is not difficult to see that the iFEM cost along the CPU time axis is lower than its dFEM counterpart, reaffirming the fact that when the same accuracy was to be maintained, the dFEM requires a very fine grid resolution, which lead to higher computational cost, and thus, iFEM could be the better option.

Conclusion

In this work, we have compared the direct and inverse finite element methods for pricing American put options. Based on the results of our numerical experiments, the dFEM, while appealing in terms of CPU time savings, produces larger error than its inverse counterpart for similar grid resolutions. Furthermore, by using the performance accuracy-efficiency curves to establish the trade-offs, the iFEM is indeed more flexible and efficient, as a higher performance in terms of a satisfactory computing time and accuracy is achieved. The results presented in this work demonstrate that the iFEM deserves consideration as an alternative numerical technique for pricing American-style options. In the subsequent works, we will exploit other advantageous features of the inverse finite element method while

considering option pricing problems under different formulations and frameworks.

CONFLICT OF INTERESTS

The author has not declared any conflict of interests.

REFERENCES

Adegboyegun BJ (2018). An inverse finite element method for pricing American options under linear complementarity formulations. Mathematics Applied in Science and Technology 10(1):1-17.

Alexandrou AN (1989). An inverse finite element method for directly formulated free boundary problems. International Journal for Numerical Methods in Engineering 28(10):2383-2396.

Arregui I, Salvador B, Vázquez C (2017). PDE models and numerical methods for total value adjustment in European and American options with counterparty risk. Applied Mathematics and Computation 308:31-53.

Black F, Scholes M (1973). The pricing of options and corporate liabilities. Journal of Political Economy 81(3):637-654.

Chemisky Y, Meraghni F, Bourgeois N, Cornell S, Echchorfi R, Patoor E (2015). Analysis of the deformation paths and thermomechanical parameter identification of a shape memory alloy using digital image correlation over heterogeneous tests. International Journal of Mechanical Sciences 96:19-24.

Cuadrado J, Gutierrez R, Naya MA, Morer P (2001). A comparison in terms of accuracy and efficiency between a MBS dynamic formulation with stress analysis and a nonlinear FEA code. International Journal for Numerical Methods in Engineering 51(9):1033-1052.

Rao SS (2017). The finite element method in engineering. Butterworth-Heinemann.

Uijlings J, Duta IC, Sangineto E, Sebe N (2015). Video classification with densely extracted hog/hof/mbh features: an evaluation of the accuracy/computational efficiency trade-off. International Journal of Multimedia Information Retrieval 4(1):33-44.

Zhang K, Song H, Li J (2015). Front-fixing FEMs for the pricing of American options based on a PML technique. Applicable Analysis 94(5):903-931.

Zhu SP (2006). An exact and explicit solution for the valuation of American put 22 options. Quantitative Finance 6(3):229-242.

Zhu SP, Chen WT (2013). An inverse finite element method for pricing American options. Journal of Economic Dynamics and Control 37(1):231-250.