null
Journal of the KSIAM

Editor-in-Chief:
Seo, Jin Keun (Yonsei University, Korea)

Associate Editor-In-Chief:
Chen, Zhiming (Chinese Academy of Sciences, China)
Lee, June-Yub (Ewha w. University, Korea)
Tang, Tao (Hong Kong Baptist University, Hong Kong)

Managing Editor:
Cho, Jin-Yeon (Inha University, Korea)
Kim, Junseok (Korea University, Korea)

Editorial Board

Ahn, Hyung Taek (University of Ulsan, Korea)
Ahn, Jaemyung (KAIST, Korea)
Campi, Marco C. (University of Brescia, Italy)
Han, Pigong (Chinese Academy of Sciences, China)
Hodges, Dewey (Georgia Tech., USA)
Hwang, Gang Uk (KAIST, Korea)
Jung, Il Hyo (Pusan Natl University, Korea)
Kim, Hyoun Jin (Seoul Natl University, Korea)
Kim, Jeong-ho (Inha University, Korea)
Kim, Kyea Hyun (Kyung Hee University, Korea)
Kim, Nakwan (Seoul Natl University, Korea)
Lee, Eunjung (Yonsei University, Korea)
Lee, Hyundae (Inha University, Korea)
Li, Zhilin (North Carolina State University, USA)
Lim, Mikyoung (KAIST, Korea)
Liu, Hongyu (Hong Kong Baptist University, Hong Kong)
Min, Chohong (Ewha w. University, Korea)
Park, Soo Hyung (Konkuk University, Korea)
Shim, Gyoocheol (Ajou University, Korea)
Shin, Eui Seop (Chubuk Natl University, Korea)
Shin, Sang Joon (Seoul Natl University, Korea)
Wang, Zhi Jian (University of Kansas, USA)
Wolf, Jörg (Institut für Mathematik, Germany)

Aims and Scope

Journal of the KSIAM is devoted to theory, experimentation, algorithms, numerical simulation, or applications in the fields of applied mathematics, engineering, economics, computer science, and physics as long as the work is creative and sound.

Copyright

It is a fundamental condition that submitted manuscripts have not been published and will not be simultaneously submitted or published elsewhere. By submitting a manuscript, the authors agree that the copyright for their article is transferred to the publisher if and when the article is accepted for publication. The copyright covers the exclusive rights to reproduce and distribute the article, including reprints, photographic reproductions, microform or any other reproductions of similar nature, and translation. Photographic reproduction, microform, or any other reproduction of text, figures or tables from this journal is prohibited without permission obtained from the publisher.

Subscription

Subscription rate for individuals is $50/year; Domestic university library rate is $150/year; Foreign or industrial institute rate is $300/year.

Contact info.
The KOREAN SOCIETY FOR INDUSTRIAL AND APPLIED MATHEMATICS
KAIST, Yuseong-gu, Daejeon 34141, Republic of KOREA
Lee, Hyung-Chun (e-mail : jksiam@ksiam.org )
Tel : +82-42-350-5708, Fax : +82-42-350-5710

Journal of KSIAM was launched in 1997, is published four times a year by the Korean Society for Industrial and Applied Mathematics. Total or part of the articles in this journal are abstracted in KCI, Mathematical Reviews, CrossRef, Korea Science, and NDSL. Full text is available at http://www.ksiam.org/archive.

Copyright © 2015, the Korean Society for Industrial and Applied Mathematics
Contents

ANALYSIS OF THE UPPER BOUND ON THE COMPLEXITY OF LLL ALGORITHM
YUNJU PARK AND JAEHYUN PARK ................................................................. 107

ARITHMETIC AVERAGE ASIAN OPTIONS WITH STOCHASTIC ELASTICITY OF VARIANCE
Kyu-Hwan Jang  Min-Ku Lee ................................................................. 123

ROBUST A POSTERIORI ERROR ESTIMATOR FOR LOWEST-ORDER FINITE ELEMENT METHODS OF INTERFACE PROBLEMS
Kwang-Yeon Kim ................................................................. 137

APPROXIMATION ORDER OF $C^3$ QUARTIC B-SPLINE APPROXIMATION OF CIRCULAR ARC
Sung Chul Bae and Young Joon Ahn ................................................................. 151

PATH AVERAGED OPTION VALUE CRITERIA FOR SELECTING BETTER OPTIONS
Junseok Kim, Minhyun Yoo, Hyeju Son, Seungyu Lee, Myeong-Hyeon Kim, Yongho Choi, Darae Jeong, and Young Rock Kim ................................................................. 163
ANALYSIS OF THE UPPER BOUND ON THE COMPLEXITY OF LLL ALGORITHM

YUNJU PARK¹ AND JAEHYUN PARK²†

¹DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, KOREA SCIENCE ACADEMY OF KOREA ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY, KOREA
E-mail address: yunjupark@kaist.ac.kr

²DEPARTMENT OF ELECTRONIC ENGINEERING, PUKYONG NATIONAL UNIVERSITY, KOREA
E-mail address: jaehyun@pknu.ac.kr

ABSTRACT. We analyze the complexity of the LLL algorithm, invented by Lenstra, Lenstra, and Lovász as a well-known lattice reduction (LR) algorithm which is previously known as having the complexity of $O(N^4 \log B)$ multiplications (or, $O(N^5(\log B)^2)$ bit operations) for a lattice basis matrix $H(\in \mathbb{R}^{M \times N})$ where $B$ is the maximum value among the squared norm of columns of $H$. This implies that the complexity of the lattice reduction algorithm depends only on the matrix size and the lattice basis norm. However, the matrix structures (i.e., the correlation among the columns) of a given lattice matrix, which is usually measured by its condition number or determinant, can affect the computational complexity of the LR algorithm.

In this paper, to see how the matrix structures can affect the LLL algorithm’s complexity, we derive a more tight upper bound on the complexity of LLL algorithm in terms of the condition number and determinant of a given lattice matrix. We also analyze the complexities of the LLL updating/downdating schemes using the proposed upper bound.

1. INTRODUCTION

Lattice reduction (LR) is a method to find the bases of the given lattice space close to the shortest vector, which has been successfully applied to cryptography, factoring the polynomials ([1–3] and references therein), and multiple-input multiple output (MIMO) communication system [4–7]. One advantageous property of LR is to improve the conditioning of a given lattice basis matrix by multiplying a unimodular matrix which is a square integer matrix with determinant $\pm 1$.

One of famous LR algorithms due to the simplicity is the LLL algorithm invented by Lenstra, Lenstra, and Lovász [1] which has the complexity of $O(N^4 \log B)$ multiplications (or, $O(N^5(\log B)^2)$ bit operations) for the given lattice basis matrix $H(\in \mathbb{R}^{M \times N})$ where $B$ is
the maximum value among the squared norm of columns of $H$. However, it is a quite loose bound and depends only on the matrix size and the norm of each column which does not tell much about how the condition number or the determinant of the lattice basis matrix affect on the complexity of the LLL algorithm. For example, since the reduced lattice bases tend to be orthogonal to each other to become close to the shortest vector, we can intuitively expect that if the lattice basis matrix is well-conditioned it requires less computational complexities. Accordingly, the matrix structures (i.e., the correlation among the columns) of a given lattice matrix, which is usually measured by its condition number or determinant, can affect the computational complexity of the LR algorithm.

In this paper, to see how the matrix structures can affect the LLL algorithm’s complexity, we derive a more tight upper bound on the complexity of LLL algorithm in terms of the condition number and determinant of a given lattice matrix. Accordingly, the proposed upper bound gives an insight into the effects of the lattice structure on the complexities of the LR algorithm. In addition, by using the proposed upper bound, we analyze the LLL updating/downdating algorithms in [8]. From the numerical results, by using the proposed upper bound, we can infer the behavior of the required computational complexities of LLL updating/downdating algorithms according to the structures of a given lattice basis matrix and the updated rows/columns.

The rest of this paper is organized as follows. In Section 2 the LLL algorithm is briefly reviewed and the basic notations and definitions related with the LLL algorithm are introduced. In Section 3 a new upper bound of the LLL algorithm’s complexity is derived. In Section 4 row-wise LLL updating and downdating methods are introduced and they are analyzed by using the proposed upper bound. Several simulation results for various conditions are given in Section 5. Concluding remarks are made in Section 6.

The superscripts in $A^T$ and $A^{-1}$ denote, respectively, the transposition and the inverse of the matrix $A$. $\|A\|_2$ and $\det(A)$ denote 2-norm and determinant of $A$, respectively. $\lfloor a \rfloor$ denotes the nearest integer of a real number $a$. $I_N$ and $0_{M,N}$ denote an $N \times N$ identity matrix and a zero $M \times N$ matrix, respectively, and $e_i$ denotes the $i$-th column of $I_N$. Finally $A(i:j,k:l)$ denotes the submatrix of $A$ with elements from the $i$-th row to the $j$-th row and from the $k$-th column to the $l$-th column.

2. Lattice Reduction: LLL Algorithm

Lattice $\mathcal{L}(H)$ is defined as $\sum_{n=1}^{N} h_n s_n$, where $h_n$, the $n$-th column vector of $H$ ($\in \mathbb{R}^{M \times N}$, $M \geq N$), is a basis vector of lattice and $s_n \in \mathbb{Z}$. A lattice basis matrix $H$ is (LLL) lattice reduced if

$$|\mu_{i,j}| \leq \frac{1}{2} \text{ for } 1 \leq i < j \leq N$$

(2.1)

and

$$3/4\|h^o_{i-1}\|^2 \leq \|h^o_i + \mu_{i-1,i}h^o_{i-1}\|^2 \text{ for } 1 < i \leq N,$$

(2.2)
where \( h^o_j = h_j - \sum_{i=1}^{j-1} \mu_{i,j} h^o_i \) with \( \mu_{i,j} = \langle h_i, h^o_j \rangle / \langle h_i, h^o_i \rangle \), which can be obtained through the Gram-Schmidt orthogonalization process. Equivalently, a lattice basis matrix \( H \) with QR decomposition \( H = QR \), where \( Q(\in \mathbb{R}^{M \times N}) \) is orthogonal and \( R(\in \mathbb{R}^{N \times N}) \) is upper triangular, is lattice reduced if
\[
|r_{i,j}| \leq 1/2|r_{i,i}| \quad \text{for} \quad 1 \leq i < j \leq N \tag{2.3}
\]
and
\[
3/4r^2_{i-1,i-1} \leq r^2_{i,i} + r^2_{i-1,i} \quad \text{for} \quad 1 < i \leq N, \tag{2.4}
\]
where \( r_{i,j} \) is the \((i, j)\)-th element of \( R \). Note that \( \mu_{i,j} = \frac{r_{i,j}}{r_{i,i}} \). If Eqn. (2.1) (equivalently, Eqn. (2.3)) is satisfied, \( H \) is called size-reduced and if Eqn. (2.2) (equivalently, Eqn. (2.4)) is satisfied, \( H \) is called two-reduced.\(^1\) The constant \( \frac{3}{4} \) in Eqn. (2.2) and Eqn. (2.4) could be arbitrarily replaced by any fixed real number within \((1/4, 1)\) [1].

LR algorithm is to reduce the norm of each basis of lattice by a linear combination of bases with integer coefficients which is equivalent to post-multiplying \( H \) with a unimodular matrix \( T \) such that \( HT \) is lattice reduced. In Table 1, LLL algorithm to compute \( T \) such that \( HT \) is lattice reduced is summarized [1].

<table>
<thead>
<tr>
<th>Table 1. LLL algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ( T = I_N, H = QR )</td>
</tr>
<tr>
<td>2 ( i = 2 )</td>
</tr>
<tr>
<td>3 while ( i \leq N )</td>
</tr>
<tr>
<td>4 for ( l = i - 1, ..., 1 )</td>
</tr>
<tr>
<td>5 ( [R, T] = \text{Size-reduction}(R, T, i, l) )</td>
</tr>
<tr>
<td>6 end</td>
</tr>
<tr>
<td>7 if ( 3/4r^2_{i-1,i-1} &gt; r^2_{i,i} + r^2_{i-1,i} )</td>
</tr>
<tr>
<td>8 ( [R, T] = \text{two-reduction}(R, T, i) )</td>
</tr>
<tr>
<td>9 ( i = \max{i-1, 2} )</td>
</tr>
<tr>
<td>10 else</td>
</tr>
<tr>
<td>11 ( i = i + 1 )</td>
</tr>
<tr>
<td>12 end</td>
</tr>
<tr>
<td>13 end</td>
</tr>
</tbody>
</table>

Note that the LLL algorithm is mainly composed of two subroutines – size-reduction and two-reduction routines in Table 2. In size-reduction step, by the modular operation, Eqn. (2.3) can be satisfied. In two-reduction step, if Eqn. (2.4) is not satisfied, the corresponding adjacent columns are swapped and the triangular form of matrix \( R \) is recovered by the Givens rotation. Note that since the Givens rotation preserves the norm of each column Eqn. (2.4) can be satisfied by simply swapping the columns. Then, \( HT \) with the unimodular matrix \( T \) obtained from LLL algorithm satisfies the LLL lattice reduction conditions (Eqns. (2.1) and (2.2)).

\(^1\)Note that since \( r^2_{i-1,i} \leq 1/4r^2_{i-1,i-1} \) from Eqn. (2.3), Eqn. (2.4) becomes \( r^2_{i-1,i-1} \leq 2r^2_{i,i} \).
Table 2. Subroutines of LLL algorithm

<table>
<thead>
<tr>
<th></th>
<th>$[\mathbf{R}, \mathbf{T}] = \text{Size-reduction}(\mathbf{R}, \mathbf{T}, i, l)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\mu = \lfloor r_{l,i}/r_{l,l} \rfloor$</td>
</tr>
<tr>
<td>2</td>
<td>if $\mu \neq 0$</td>
</tr>
<tr>
<td>3</td>
<td>$\mathbf{R}(1 : l, i) = \mathbf{R}(1 : l, i) - \mu \mathbf{R}(1 : l, l)$</td>
</tr>
<tr>
<td>4</td>
<td>$\mathbf{T}(1 : N, i) = \mathbf{T}(1 : N, i) - \mu \mathbf{T}(1 : N, l)$</td>
</tr>
<tr>
<td>5</td>
<td>end</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$[\mathbf{R}, \mathbf{T}] = \text{two-reduction}(\mathbf{R}, \mathbf{T}, i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Swap columns $i - 1$ and $i$ in $\mathbf{R}$ and $\mathbf{T}$</td>
</tr>
<tr>
<td>2</td>
<td>Triangularize $\mathbf{R}$ using Givens rotation matrix $\Theta$</td>
</tr>
<tr>
<td>3</td>
<td>$\mathbf{R}(i - 1 : i, i - 1 : N) = \Theta \mathbf{R}(i - 1 : i, i - 1 : N)$</td>
</tr>
</tbody>
</table>

Note that line 3 and 4 of Size-reduction subroutine in Table 2 can be represented in terms of matrices as

$$\mathbf{R} = \mathbf{R} \mathbf{S}_{li}, \quad \mathbf{T} = \mathbf{T} \mathbf{S}_{li},$$

(2.5)

where $\mathbf{S}_{li} = \mathbf{I}_N - \mu \mathbf{e}_i \mathbf{e}_i^T$. Similarly, Line 1 of two-reduction subroutine in Table 2 can be rewritten as

$$\mathbf{R} = \mathbf{R} \mathbf{P}_i, \quad \mathbf{T} = \mathbf{T} \mathbf{P}_i,$$

(2.6)

where

$$\mathbf{P}_i = \begin{bmatrix} I_{i-2} & 0 & 1 \\ 0 & 1 & 0 \\ I_{N-i} & & & \end{bmatrix}.$$ 

(2.7)

Therefore, $\mathbf{T}$ is a product of sequence of matrices $\mathbf{S}_{li}$ and $\mathbf{P}_i$.

To evaluate the computational complexity of LLL algorithm, we introduce the following lemma [9].

**Lemma 1.** If we define $D \triangleq r_{11}^{2N} \cdot r_{22}^{2(N-1)} \cdots r_{NN}^2$, the size-reduction step does not affect $D$, while the two-reduction step decreases $D$ by a factor of at least 3/4.

Accordingly, the number of while loop iterations in the LLL algorithm of Table 1 is at most $O(\log D_0)^2$, where $D_0 = |h_1|^{2N} \cdot |h_2|^{2(N-1)} \cdots |h_N|^2$. If we set $B = \max\{|h_1|^2, \ldots, |h_N|^2\}$, $O(\log D_0) \approx O(N^2 \log B)$. Since the size-reduction step in the lines 4–6 of Table 1 requires only $O(N^2)$ multiplications, total $O(N^4 \log B)$ multiplications are required. Each real number used during the process of the algorithm is bounded by $O(N \log B)$ bits [1]. Hence, total $O(N^5(\log B)^2)$ bit operations are required in LLL algorithm. Note that because the LLL algorithm’s complexity depends on the number of while loop iterations in the LLL algorithm,

\[2\] As for Lemma 1, in this paper log denotes the logarithm with a basis 4/3.
we would investigate the number of the two-reduction steps as a measure of the computational complexity.

**Remark 1.** The LLL algorithm does not change the absolute value of the determinant of the lattice basis matrix since $\det(T) = \pm 1$. Therefore,

$$\det(H^TH) = \prod_{i=1}^{N} \tilde{r}_{ii}^2 = \prod_{i=1}^{N} r_{ii}^2,$$

where $r_{ii}^2$ is the component of the upper triangular matrix $\tilde{R} \in \mathbb{R}^{N \times N}$ in the QR decomposition of the reduced lattice basis matrix $G = HT$.

### 3. A NEW UPPER BOUND OF THE LLL ALGORITHM’S COMPLEXITY

The complexity derived from Lemma 1 in Section 2 indicates that it depends only on the matrix size and the lattice basis norm. In this section, we would derive a new upper bound which can be used to analyze the complexity of LLL algorithm as well as other variants of the LLL basis algorithms such as the LLL updating/downdating algorithms [8]. From Section 2, the number of two-reduction steps is critical in the complexity analysis and, from $D$ defined in Lemma 1, the upper bound of the number of two-reduction steps can be induced. Accordingly, in what follows, to get a more tight upper bound for the number of two reduction steps, we derive the lower bound of $D$, denoted as $D_l$.

Because the determinant of lattice basis matrix does not change as discussed in Remark 1, we can have the following lemma.

**Lemma 2.** Let $H \in \mathbb{R}^{M \times N}, M \geq N$ have the preconditioning matrix $T \in \mathbb{Z}^{N \times N}$ which is a unimodular matrix computed by LLL algorithm in Table 1. Then $HT$ has a QR decomposition as

$$G = HT = \tilde{Q}\tilde{R},$$

where $\tilde{Q} \in \mathbb{R}^{M \times N}$ is orthogonal and $\tilde{R} \in \mathbb{R}^{N \times N}$ is upper triangular satisfying Eqns. (2.3) and (2.4). Then,

$$\max_{1 \leq i,j \leq N} |\tilde{r}_{ii}| \leq \max_{1 \leq i,j \leq N} |r_{ij}| \leq \kappa(H),$$

where $\kappa(A)$ is the condition number of a matrix $A$.

**Proof.** For the first inequality in Eqn. (3.2) $(\max_{1 \leq i,j \leq N} |\tilde{r}_{ij}| \leq \max_{1 \leq i,j \leq N} |r_{ij}|)$, we let $(i_{mx}, j_{mn}) = \arg_{i,j} \max_{1 \leq i,j \leq N} \frac{|r_{ij}|}{|r_{jj}|}$. Note that the two-reduction step can only change the diagonal elements of $R$ during the LLL process. Accordingly, for the numerator $|r_{i_{mx}j_{mx}}|$, because $|r_{i_{mx}j_{mx}}| \geq |r_{ii}|$ for $1 \leq i \leq N$, two-reduction step is performed during the LLL process only when

$$3/4r_{i_{mx}j_{mx}}^2 > r_{i_{mx}j_{mx+1}}^2 + r_{i_{mx}j_{mx+1}}^2,$$
which alters \( r'_{imximz} \) into \( r'_{imximz} \), with \( |r'_{imximz}| \) \leq |r_{imximz}|. Because the Givens rotation preserves the norm of each column, we have
\[
|r'_{imximz}| \geq |r_{imximz}| + 1.
\]
(3.3)
In addition, because \( r_{imximz}r'_{imximz}1 \) = \( r'_{imximz}r'_{imximz}1 \) from Remark 1, together with Eqn. (3.3), we can get
\[
|r'_{imximz}1| \leq |r_{imximz}|.
\]
Similarly, for the denominator \( r_{jmnjmn} \) (which satisfies that \( |r_{jmnjmn}| \) \leq |\( r_{ij} \)| for \( 1 \leq i \leq N \), the two-reduction step is performed during the LLL process when \( 3/4r_{jmnjmn}^2 > r_{jmnjmn}^2 + r_{jmnjmn}^2 \), resulting
\[
|r'_{jmnjmn}|; |r'_{jmnjmn}1| \geq |r_{jmnjmn}|.
\]
(3.4)
Therefore, during the LLL process, two-reduction step always reduces \( \max_{1 \leq i,j \leq N} \frac{|r_{ii}|}{|r_{jj}|} \).

For the second inequality in Eqn. (3.2) \( (\max_{1 \leq i,j \leq N} \frac{|r_{ij}|}{|r_{jj}|}) \leq \kappa(H) \), it can be easily derived from that \( \kappa(H) = \|R\||R^{-1}\|, \|R\|_2 \geq \max |r_{ii}|, \) and \( \|R\|_2 \geq \max |r_{jj}| \).

From Lemma 2, we denote \( (\check{r}_{iimz}'^{'jmn}) = \arg \max_{1 \leq i,j \leq N} \frac{|r_{ij}|}{|r_{jj}|} \). Note that \( \check{r}_{jmnjmn}^2 \) has a lower bound as \( \check{r}_{jmnjmn}^2 = \kappa(H)^{-2}r_{iimz}'^{'jmn} \). Let \( \check{D} = \check{r}_{11}^2 \check{r}_{22}^2 \cdots \check{r}_{NN}^2 \) for \( R \) as defined in Eqn. (3.1). Because \( \check{r}_{ii}^2 \geq \check{r}_{jmnjmn}^2 \) for \( 1 \leq i \leq N, \check{D} \) has a lower bound when the set of diagonal elements is given as:
\[
|\check{r}_{11}| = |\check{r}_{22}| = \cdots = |\check{r}_{N-1N-1}| = |\check{r}_{jmnjmn}| = \kappa(H)^{-1}|\check{r}_{iimz}| = |\check{r}_{iimz}|.
\]
(3.5)
Accordingly, we have the following corollary.

**Corollary 3.1.** \( \check{D} \) has a lower bound as:
\[
\check{D} \geq D_t = \frac{(\det(H^TH))^{(N+1)/2}}{\kappa(H)^{N-1}}.
\]
(3.6)
**Proof.** Because \( \det(H^TH) = \prod_{i=1}^{N} \check{r}_{ii}^2 \), Eqn. (3.5) implies that
\[
\det(H^TH) = \prod_{i=1}^{N} \check{r}_{ii}^2 = \kappa(H)^{-2(N-1)}|\check{r}_{iimz}|^2N.
\]
(3.7)
That is, \( |\check{r}_{iimz}| = \det(H^TH)^{1/N} \kappa(H)^{2(N-1)/2N} \) and therefore, we can get
\[
\check{r}_{11}^2 = \check{r}_{22}^2 = \cdots = \check{r}_{N-1N-1} = (\det(H^TH))^{1/N} \kappa(H)^{-2/N},
\]
\[
\check{r}_{NN}^2 = (\det(H^TH))^{1/N} \kappa(H)^{2(N-1)/N}.
\]
(3.8)
ANALYSIS OF THE UPPER BOUND ON THE COMPLEXITY OF LLL ALGORITHM

Therefore \( r_i^2 \) in Eqn. (3.8) induce the lower bound \( D_l \) as:

\[
D_l = r_{11}^2 r_{22}^2 \cdots r_{NN}^2 = \frac{(\det(H^T H))^{(N+1)/2}}{(\kappa(H))^{N-1}} \tag{3.9}
\]

Note that \( D_l \) can be represented in terms of \( R \) as \( \sim D_l = (\det(R))^{N+1} \). Therefore we can derive a new upper bound for the required number of two-reduction steps for LLL algorithm as

\[
N_s = O(\log D_n),
\]

where \( D_n = \prod_{i=1}^{N} r_{ii}^{2(N-i+1)} (\kappa(R))^{N-1} \). (3.10)

Remark 2. The new upper bound for the two-reduction steps indicating LLL complexity is dependent not only on the lattice basis matrix size and the lattice basis norm but also on its determinant and the condition number. That is, if the given lattice basis matrix is well conditioned or its determinant is large then the complexity can be diminished.

Remark 3. As a simple example of validity of the new upper bound, let us think about \( \alpha I_N \) where \( \alpha \in \mathbb{R} \). While the previous upper bound for the number of the two-reduction steps is \( O(\log D_0) \) where \( D_0 = |\alpha|^{(N+1)} \), the proposed upper bound is 0 because \( D_N = 1 \) which coincide with that \( \alpha I_N \) is already lattice-reduced.

4. COMPLEXITY ANALYSIS OF LLL UPDATING AND DOWNDATING

In this section, we briefly introduce the row-wise updating/downdating methods [8] and analyze their complexities using the proposed upper bound in Section 3.

4.1. Row-wise Updating. Let \( H(\in \mathbb{R}^{M \times N}, M \geq N) \) have a QR decomposition as

\[
H = QR, \tag{4.1}
\]

where \( Q(\in \mathbb{R}^{M \times N}) \) is orthogonal and \( R(\in \mathbb{R}^{N \times N}) \) is upper triangular and assume that given \( H \) we have the preconditioning matrix \( T(\in \mathbb{Z}^{N \times N}) \) which is a unimodular matrix computed by LLL algorithm in Table 1. Here, the QR decomposition of \( HT(\in \mathbb{R}^{M \times N}) \) is also known as:

\[
G = HT = \tilde{Q}\tilde{R}, \quad \tilde{R} \in \mathbb{R}^{N \times N} \tag{4.2}
\]

where \( \tilde{R} \) satisfies Eqns. (2.3) and (2.4). We then want to find a new preconditioning matrix \( T_u \) after a new row \( h_T^T \) is added as:

\[
G_u = H_u T_u = \begin{bmatrix} h_T^T \\ H \end{bmatrix} T_u, \quad h_T^T \in \mathbb{R}^{1 \times N} \tag{4.3}
\]

where \( G_u \) is lattice-reduced. Because the QR decomposition of \( HT \) is known as Eqn. (4.2) we can update the new row \( h_T^T \) based on \( Q \) and \( R \) rather than using \( Q \) and \( R \) as follows:

\[
G' = \begin{bmatrix} h_T^T T \\ HT \end{bmatrix} = \begin{bmatrix} h_T^T T \\ QT \end{bmatrix} = \begin{bmatrix} 1 & 0_{1 \times N} \\ 0_{N \times 1} & Q \end{bmatrix} \begin{bmatrix} h_T^T T \\ \tilde{R} \end{bmatrix}, \tag{4.4}
\]
where \( \begin{bmatrix} h_T^T & T \end{bmatrix} \) has a Hessenberg form. We can then recover the triangular form by using \( N \) Givens rotations as:

\[
J^T_N \cdots J^T_1 \begin{bmatrix} h_T^T & T \end{bmatrix} = R_u^{in},
\]

where \( J^T_1 \) is the Givens rotation matrix forcing zero on the \((i+1, i)\)-th entry of the Hessenberg matrix in Eqn. (4.5). We then compute the preconditioning matrix \( T_u \) by using \( R_u^{in} \) and \( R \) as initial parameters. Note that \( Q \) is not required during the LLL process.

Here, to analyze the number of two-reduction steps for the updating algorithm, we represent the squared norm of the diagonal elements of \( R_u^{in} \) in Eqn. (4.5) in terms of \( R \) and \( h_T^T T \) (\( \equiv h^T \)) as:

\[
(r_{ii}^2) = (k_i^{(j)})^2 = \tilde{r}_{ii}^2 + \sum_{j=1}^{i-1} \tilde{r}_{jj}^2 \prod_{k=1}^{i-j} \sin^2 \theta^{(k)} + h_{ij}^2 \prod_{j=1}^{i} \sin^2 \theta^{(i-j)},
\]

where

\[
k_i^{(j)} = \sqrt{(k_i^{(j-1)} \sin \theta_i^{(j-1)})^2 + \tilde{r}_{jj}^2} \quad \text{for } 1 \leq j \leq i \leq N,
\]

\[
\theta_i^{(j)} = \cos^{-1} \frac{\tilde{r}_{jj} k_i^{(j)} \sin \theta_i^{(j-1)} + \tilde{r}_{ji} k_j^{(j-1)} \sin \theta_j^{(j-1)}}{k_i^{(j)} k_j^{(j)}} \quad \text{for } 1 \leq j < i \leq N,
\]

with \((k_i^{(0)})^2 = h_i^2\) and \(\theta_i^{(0)} = 90^\circ\), and \(h_i^j\) is the \(i\)-th element of \( h^T \). See also [8] for the details.

To get the upper bound of the number of two-reduction steps, we compute \( D_N \) for \( R_u^{in} \) as defined in Eqn. (3.10):

\[
D_N = (k_1^{(1)})^2 (k_2^{(2)})^2 (k_N^{(N)})^2 \frac{(\kappa(\tilde{R}_u^{in}))^N}{(\det(\tilde{R}_u^{in}))^N+1} = \left( \tilde{r}_{11}^2 + h_1^2 \right)^N \left( \tilde{r}_{22}^2 + \tilde{r}_{12}^2 \sin^2 \theta_2^{(1)} + h_2^2 \sin^2 \theta_2^{(2)} + \tilde{r}_{11}^2 \right)^2 \ldots
\]

\[
= (\tilde{r}_{NN}^2 + \sum_{j=1}^{N-1} \tilde{r}_{jj}^2 \prod_{k=1}^{N-j} \sin^2 \theta_k^{(N-k)} + h_{N}^2 \prod_{j=1}^{N} \sin^2 \theta_j^{(N)} \frac{(\kappa(\tilde{R}_u^{in}))^N}{(\det(\tilde{R}_u^{in}))^N+1})^{N-i+1},
\]

where \( D_P = \tilde{r}_{11}^2 \tilde{r}_{22}^2 \ldots \tilde{r}_{NN}^2 \frac{\kappa(\tilde{R})}{(\det(\tilde{R}))^N+1} \) is associated with \( \text{HT} \), the already reduced lattice bases before adding a new row. Here, \( D_S \) is given as:

\[
D_S = \left( \frac{\kappa(\tilde{R}_u^{in})}{\kappa(\tilde{R})} \right)^{N-1} \left( \frac{(\det(\tilde{R}))}{(\det(\tilde{R}_u^{in}))} \right)^{N+1},
\]
Therefore, the number of two-reduction steps required in the updating algorithm is at most $O(\log D_u)$, where

$$D_u = D_S \prod_{i=1}^{N} \left( 1 + \sum_{j=1}^{i-1} \frac{\tilde{r}_{ij}^2}{\tilde{r}_{ii}^2} \prod_{k=1}^{i-j} \sin^2 \theta_i^{(k)} + \frac{h_i^2}{\tilde{r}_{ii}^2} \prod_{j=1}^{i} \sin^2 \theta_{(i-j)}^{(i-j)} \right)^{N-i+1}. \quad (4.10)$$

Here, since $\tilde{r}_{ij}$ is from the QR decomposition of the already reduced lattice basis matrix in Eqn. (3.1), due to the two LR conditions of Eqns. (2.3) and (2.4), we can easily induce that $\tilde{r}_{ji}^2 \leq \tilde{r}_{ii}^2$ for $j < i$. Therefore, the second term in Eqn. (4.10) is always less than $i - 1$, which leads to the following inequality:

$$D_u \leq \tilde{D}_u, \quad (4.11)$$

where

$$\tilde{D}_u \triangleq D_S \prod_{i=1}^{N} \left( i + \frac{h_i^2}{\tilde{r}_{ii}^2} \prod_{j=1}^{i} \sin^2 \theta_i^{(i-j)} \right)^{N-i+1}. \quad (4.12)$$

Accordingly, the required number of two-reduction steps for the LLL updating algorithm is upper bounded by $O(\log \tilde{D}_u)$.

**Remark 4.** Whenever any new rows are added, due to the last two terms in Eqn. (4.6), the norm of each diagonal term always increases, from which it can be derived that $D_S \leq 1$. That is, from Eqn. (4.12), the upper bound of the number of two-reduction steps can be reduced. In other words, when a new row is added in the current lattice basis matrix, the correlation among the lattice bases tends to be decreased relatively (or the ratio of the norm of diagonal elements to that of off-diagonal element becomes larger). Accordingly, the increase of the row size induces less column subtractions in the size-reduction step, also resulting in less two-reduction steps in the LLL process.

### 4.2. Row-wise Downdating

In this section, given $H$ in Eqn. (4.1) with preconditioning matrix $T$, we introduce the method in [8] to find the new preconditioning matrix $T_d$ after the first row $h_1^T$ is removed:

$$G_d = H_d T_d, \quad H = \begin{bmatrix} h_1^T \\ H_d \end{bmatrix}, \quad H_d \in \mathbb{R}^{M-1 \times N} \quad (4.13)$$

where $G_d$ is lattice reduced. Since the LLL algorithm is independent on the orthogonal matrix of QR decomposition as discussed in Remark 1, we first start the following equation as:

$$H_d^T H_d = H^T H - h_1 h_1^T = \begin{bmatrix} I_N & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} H^T \\ h_1^T \end{bmatrix}. \quad (4.14)$$

Since $H^T H = T^{-T} \tilde{R}^T \tilde{R} T^{-1}$, Eqn. (4.14) can be written as:

$$H_d^T H_d = T^{-T} \tilde{R}^T \tilde{R} T^{-1} - h_1 h_1^T$$
\[
\begin{bmatrix}
T^{-T} \tilde{R}^T & h_1
\end{bmatrix}
\begin{bmatrix}
I_N & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix}
\tilde{R} T^{-1} \\
h_1^T
\end{bmatrix}
= T^{-T} \begin{bmatrix}
\tilde{R}^T & T^T h_1
\end{bmatrix}
\begin{bmatrix}
I_N & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix}
\tilde{R} \\
h_1^T T
\end{bmatrix} T^{-1}.
\]
Therefore,
\[
T^T H_d^T H_d T = \begin{bmatrix}
\tilde{R}^T & T^T h_1
\end{bmatrix}
\begin{bmatrix}
I_N & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix}
\tilde{R} \\
h_1^T T
\end{bmatrix}.
\]
Like Cholesky downdating, we can then recover the triangular form by applying the hyperbolic rotation as [10]
\[
L_N \cdots L_1 \begin{bmatrix}
\tilde{R} \\
h_1^T T
\end{bmatrix} = \begin{bmatrix}
\tilde{R}_d^{in} \\
0
\end{bmatrix},
\]
where \(L_i\) is the hyperbolic rotation matrix forcing zero on the \(i\)-th entry of \(h_1^T T\) \((\triangleq h_i^T T)\).
Therefore, we can find the new preconditioning matrix \(T_d\) with initial parameters \(\tilde{R}_d^{in}\) and \(T\).
The squared norm of the diagonal terms in \(\tilde{R}_d^{in}\) can be given as:
\[
(\tilde{r}_i^{in})^2 = \tilde{r}_i^2 - (P(\tilde{r}_{i;i-1,i}, h_i', \theta_{1:i-1}))^2 \geq 0,
\]
where \(P(\tilde{r}_{1;i-1,i}, h_i', \theta_{1:i-1})\) is a polynomial of \(\tilde{r}_{j,i}, \ h_i', \ \sinh \theta_j\) and \(\cosh \theta_j\) for \(j = 1, \ldots, i-1\).
For example, for \(i = 1\), we can easily get \((\tilde{r}_1^{in})^2 = \tilde{r}_1^2 - h_1^2\). Accordingly, \(D_N\) for \(\tilde{R}_d^{in}\) as defined in Eqn. (3.10) can be computed as:
\[
D_N = (\tilde{r}_1^{in})^2 \cdots (\tilde{r}_N^{in})^2 \frac{(\kappa(\tilde{R}_d^{in}))^{N-1}}{(\det(\tilde{R}_d^{in}))^{N+1}}
= D_P D_S \prod_{i=1}^{N} \left(1 - \frac{1}{\tilde{r}_i^2} (P(\tilde{r}_{i;i-1,i}, h_i', \theta_{1:i-1}))^2 \right)^{N-i+1},
\]
where \(D_P = \tilde{r}_1^{2N} \tilde{r}_2^{2(N-1)} \cdots \tilde{r}_N^{2(\text{det}(\tilde{R}_d^{in}))^{N+1}}\) is associated with \(HT\), the already reduced lattice bases before removing a row and \(D_S\) is given as:
\[
D_S = \frac{(\kappa(\tilde{R}_d^{in}))^{N-1}}{(\kappa(\tilde{R}_d^{in}))^{N+1}}.
\]
Because \(\kappa(\tilde{R}_d^{in}) \geq \kappa(\tilde{R})\) and \(\det(\tilde{R}_d^{in}) \geq \det(\tilde{R}_d^{in})\), \(D_S \geq 1\), in general. Accordingly, the required number of two-reduction steps for the LLL downdating algorithm is upper bounded by \(O(\log D_d)\), where
\[
D_d = D_S \prod_{i=1}^{N} \left(1 - \frac{1}{\tilde{r}_i^2} (P(\tilde{r}_{1;i-1,i}, h_i', \theta_{1:i-1}))^2 \right)^{N-i+1}.
\]
5. Numerical Results

In this section, we discuss the numerical results obtained through Monte–Carlo simulation for various conditions. For each case, we generate 10000 lattice basis matrices randomly. Each entry of lattice basis matrix $H$ is generated according to Gaussian distribution with zero-mean and unit-variance. Newly updated row vector $h^T_r$ also follows the Gaussian distribution with zero-mean and unit-variance. All the simulations are run by using MATLAB.

Table 3. Comparison of the average condition numbers and the average number of two-reduction steps.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\kappa(H)$</th>
<th>$\kappa(HT)$</th>
<th>$N_s$</th>
<th>$\log(D_P)$</th>
<th>$\log(D_N)$</th>
<th>$\log(D_P)/N_s$</th>
<th>$\log(D_N)/N_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>154.30</td>
<td>3.63</td>
<td>19.05</td>
<td>175.5</td>
<td>142.37</td>
<td>9.21</td>
<td>7.47</td>
</tr>
<tr>
<td>10</td>
<td>228.70</td>
<td>4.61</td>
<td>28.76</td>
<td>316.5</td>
<td>208.80</td>
<td>11.01</td>
<td>7.26</td>
</tr>
<tr>
<td>12</td>
<td>208.58</td>
<td>5.96</td>
<td>38.49</td>
<td>504.8</td>
<td>283.01</td>
<td>13.11</td>
<td>7.35</td>
</tr>
<tr>
<td>14</td>
<td>603.65</td>
<td>7.61</td>
<td>48.08</td>
<td>742.8</td>
<td>365.60</td>
<td>15.45</td>
<td>7.60</td>
</tr>
<tr>
<td>16</td>
<td>266.06</td>
<td>9.67</td>
<td>56.45</td>
<td>1030.1</td>
<td>454.50</td>
<td>18.25</td>
<td>8.05</td>
</tr>
<tr>
<td>18</td>
<td>293.43</td>
<td>11.96</td>
<td>64.98</td>
<td>1372.8</td>
<td>554.22</td>
<td>21.13</td>
<td>8.53</td>
</tr>
<tr>
<td>20</td>
<td>375.31</td>
<td>14.71</td>
<td>72.55</td>
<td>1766.8</td>
<td>666.02</td>
<td>24.35</td>
<td>9.18</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$N + 2$</th>
<th>$\kappa(H)$</th>
<th>$\kappa(HT)$</th>
<th>$N_s$</th>
<th>$\log(D_P)$</th>
<th>$\log(D_N)$</th>
<th>$\log(D_P)/N_s$</th>
<th>$\log(D_N)/N_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>10.66</td>
<td>3.58</td>
<td>12.31</td>
<td>231.2</td>
<td>83.77</td>
<td>9.21</td>
<td>7.47</td>
</tr>
<tr>
<td>10</td>
<td>13.16</td>
<td>4.50</td>
<td>18.18</td>
<td>386.4</td>
<td>126.20</td>
<td>11.01</td>
<td>7.26</td>
</tr>
<tr>
<td>12</td>
<td>16.01</td>
<td>5.62</td>
<td>24.68</td>
<td>588.1</td>
<td>178.01</td>
<td>13.11</td>
<td>7.35</td>
</tr>
<tr>
<td>14</td>
<td>18.65</td>
<td>6.95</td>
<td>30.95</td>
<td>839.6</td>
<td>235.22</td>
<td>15.45</td>
<td>7.60</td>
</tr>
<tr>
<td>16</td>
<td>21.62</td>
<td>8.48</td>
<td>37.45</td>
<td>1142.0</td>
<td>302.88</td>
<td>18.25</td>
<td>8.05</td>
</tr>
<tr>
<td>18</td>
<td>24.46</td>
<td>10.18</td>
<td>43.20</td>
<td>1497.0</td>
<td>375.42</td>
<td>21.13</td>
<td>8.53</td>
</tr>
<tr>
<td>20</td>
<td>26.59</td>
<td>12.01</td>
<td>48.79</td>
<td>1907.5</td>
<td>454.41</td>
<td>24.35</td>
<td>9.18</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$N + 4$</th>
<th>$\kappa(H)$</th>
<th>$\kappa(HT)$</th>
<th>$N_s$</th>
<th>$\log(D_P)$</th>
<th>$\log(D_N)$</th>
<th>$\log(D_P)/N_s$</th>
<th>$\log(D_N)/N_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>6.52</td>
<td>3.49</td>
<td>10.08</td>
<td>266.7</td>
<td>64.62</td>
<td>9.21</td>
<td>7.47</td>
</tr>
<tr>
<td>10</td>
<td>8.00</td>
<td>4.33</td>
<td>14.93</td>
<td>433.9</td>
<td>99.42</td>
<td>11.02</td>
<td>7.21</td>
</tr>
<tr>
<td>12</td>
<td>9.47</td>
<td>5.31</td>
<td>19.96</td>
<td>647.4</td>
<td>140.45</td>
<td>12.46</td>
<td>7.04</td>
</tr>
<tr>
<td>14</td>
<td>11.02</td>
<td>6.43</td>
<td>25.40</td>
<td>910.4</td>
<td>188.72</td>
<td>14.02</td>
<td>7.74</td>
</tr>
<tr>
<td>16</td>
<td>12.46</td>
<td>7.65</td>
<td>30.77</td>
<td>1225.3</td>
<td>242.87</td>
<td>15.56</td>
<td>7.89</td>
</tr>
<tr>
<td>18</td>
<td>14.02</td>
<td>9.00</td>
<td>35.63</td>
<td>1592.0</td>
<td>303.91</td>
<td>18.25</td>
<td>8.53</td>
</tr>
<tr>
<td>20</td>
<td>15.56</td>
<td>10.44</td>
<td>40.66</td>
<td>2015.0</td>
<td>373.26</td>
<td>24.35</td>
<td>9.18</td>
</tr>
</tbody>
</table>

Table 3 lists the data for various column sizes $N$ with $M = \{N, N + 2, N + 4\} - \kappa(H), \kappa(HT), N_s, \log(D_P),$ and $\log(D_N)$ where $T$ is obtained by using LLL algorithm and $N_s$ is the average numbers of two-reduction steps performed when the LLL algorithm is applied to $H$. Here, $\log(D_P)$ is the upper bound of the number of two-reduction steps as defined in Lemma 1 and $\log(D_N)$ is a newly derived upper bound in Section 3. From the condition number, we note that the alternative of using techniques to generate random unimodular matrices in [11–13] may be utilized, but in this paper, to consider the application of LLL to the engineering field (especially, wireless communications), Gaussian random distribution is adopted.
can find that $\kappa(HT)$ is lower than $\kappa(H)$, regardless of matrix size. That is, through the LLL algorithm, we can find the better-conditioned or shorter lattice basis set (or, matrix) than the original lattice basis set. In comparison of the average numbers of the two-reduction steps ($N_s$), the case for $M = N$ requires more two-reduction steps than those for $M = \{N+2, N+4\}$. However, the conventional upper bound ($\log(D_P)$) for $N = M$ has the smallest values compared to the other cases ($M = \{N+2, N+4\}$). That is, it does not catch that the number of two-reduction steps decreases when $M - N$ increases. Interestingly, the proposed new upper bound ($\log(D_N)$) decreases as $M - N$ increases, which coincides with the observation that the average numbers of the two-reduction steps also decreases for increasing $M - N$. Accordingly, from the proposed upper bound, we can infer the behavior of the computational complexity of lattice reduction according to the change of the matrix structure.

Table 4 shows the data for various column sizes $N$ for $H_a$, where $H_a = [h^T_r; H]$ ($H \in \mathbb{R}^{M \times N}$, $M = N$, and $h^T_r \in \mathbb{R}^{1 \times N}$). We evaluate $\kappa(H_a)$, $\kappa(H_aT_1)$, $\kappa(H_aT_2)$ where $T_1$ and $T_2$ are respectively obtained by using the updating algorithm and by using the conventional LLL algorithm. We also compare the average numbers of two-reduction steps $- N_u$ indicates that the updating algorithm in Section 4.1 is applied to $H_a$ and $N_t$ indicates that the conventional LLL algorithm is applied to $H_a$. From the condition numbers in Table 4, both LLL algorithm and the updating algorithm exhibit similar condition numbers. In comparison of the average numbers of the two-reduction steps, the updating algorithm performs much less two-reduction steps compared to the case when the LLL algorithm is applied to $H_a$ without updating algorithm. That is, the proposed updating scheme removes the redundant computational complexities.

### Table 4. Comparison of the average condition numbers and the average number of two-reduction steps for the row-wise updating algorithm when $M = N$.  

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\kappa(H_a)$</th>
<th>$\kappa(H_aT_1)$</th>
<th>$\kappa(H_aT_2)$</th>
<th>$N_u$</th>
<th>$N_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>17.96</td>
<td>3.59</td>
<td>3.60</td>
<td>4.75</td>
<td>14.46</td>
</tr>
<tr>
<td>10</td>
<td>23.01</td>
<td>4.50</td>
<td>4.56</td>
<td>6.37</td>
<td>21.53</td>
</tr>
<tr>
<td>12</td>
<td>28.26</td>
<td>5.70</td>
<td>5.76</td>
<td>8.18</td>
<td>29.22</td>
</tr>
<tr>
<td>14</td>
<td>33.46</td>
<td>7.13</td>
<td>7.26</td>
<td>9.55</td>
<td>36.60</td>
</tr>
<tr>
<td>16</td>
<td>37.16</td>
<td>8.84</td>
<td>9.01</td>
<td>10.97</td>
<td>43.27</td>
</tr>
<tr>
<td>18</td>
<td>41.10</td>
<td>10.81</td>
<td>10.97</td>
<td>12.32</td>
<td>50.12</td>
</tr>
<tr>
<td>20</td>
<td>47.76</td>
<td>12.94</td>
<td>13.10</td>
<td>13.39</td>
<td>56.03</td>
</tr>
</tbody>
</table>

Table 5 shows the data for the same simulation environments as in Table 4 except $M = N + 3$. We can find the similar observation that the updating algorithm requires less two-reduction steps indicating the less computational complexities. Note that the case for $M = N + 3$ requires less two-reduction steps than that for $M = N$, which is consistent with the discussion in Remark 4.
Table 5. Comparison of the average condition numbers and the average number of two-reduction steps for the row-wise updating algorithm when \( M = N + 3 \).

<table>
<thead>
<tr>
<th>( N )</th>
<th>( \kappa(H_b) )</th>
<th>( \kappa(H_bT_1) )</th>
<th>( \kappa(H_bT_2) )</th>
<th>( N_u )</th>
<th>( N_t )</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>6.55</td>
<td>3.46</td>
<td>3.48</td>
<td>1.46</td>
<td>10.10</td>
</tr>
<tr>
<td>10</td>
<td>7.99</td>
<td>4.28</td>
<td>4.32</td>
<td>2.15</td>
<td>14.98</td>
</tr>
<tr>
<td>12</td>
<td>9.48</td>
<td>5.26</td>
<td>5.31</td>
<td>2.87</td>
<td>20.08</td>
</tr>
<tr>
<td>14</td>
<td>10.99</td>
<td>6.36</td>
<td>6.41</td>
<td>3.63</td>
<td>25.40</td>
</tr>
<tr>
<td>16</td>
<td>12.51</td>
<td>7.59</td>
<td>7.64</td>
<td>4.53</td>
<td>30.81</td>
</tr>
<tr>
<td>18</td>
<td>13.95</td>
<td>8.93</td>
<td>8.98</td>
<td>5.24</td>
<td>35.53</td>
</tr>
<tr>
<td>20</td>
<td>15.52</td>
<td>10.38</td>
<td>10.44</td>
<td>6.12</td>
<td>40.28</td>
</tr>
</tbody>
</table>

Table 6 and Table 7 show the condition numbers and the numbers of two-reduction steps for the outputs with and without the row-wise downdating method for the submatrix \( H_b \) of \( H \), where \( H = [h_1^T; H_b] \), \( H \in \mathbb{R}^{M \times N} \), and \( h_1^T \in \mathbb{R}^{1 \times N} \). Again, we compare \( \kappa(H_b) \), \( \kappa(H_bT_1) \), \( \kappa(H_bT_2) \) where \( T_1 \) and \( T_2 \) are respectively obtained by using the downdating algorithm and by using the conventional LLL algorithm. In Table 6, we set \( M = N + 1 \) and can also find that both \( \kappa(H_bT_1) \) and \( \kappa(H_bT_2) \) have lower values than \( \kappa(H_b) \). Moreover, from the results for \( M = N + 3 \) in Table 7, as the difference \( M - N \) becomes higher, the less computational complexities are required.

Table 6. Comparison of the average condition numbers and the average number of two-reduction steps for the row-wise downdating algorithm when \( M = N + 1 \).

<table>
<thead>
<tr>
<th>( N )</th>
<th>( \kappa(H_b) )</th>
<th>( \kappa(H_bT_1) )</th>
<th>( \kappa(H_bT_2) )</th>
<th>( N_d )</th>
<th>( N_t )</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>136.59</td>
<td>3.60</td>
<td>3.61</td>
<td>7.13</td>
<td>19.25</td>
</tr>
<tr>
<td>10</td>
<td>299.28</td>
<td>4.60</td>
<td>4.63</td>
<td>10.41</td>
<td>28.80</td>
</tr>
<tr>
<td>12</td>
<td>183.53</td>
<td>5.90</td>
<td>5.94</td>
<td>13.55</td>
<td>38.57</td>
</tr>
<tr>
<td>14</td>
<td>2205.13</td>
<td>7.55</td>
<td>7.62</td>
<td>16.59</td>
<td>48.22</td>
</tr>
<tr>
<td>16</td>
<td>379.73</td>
<td>9.53</td>
<td>9.63</td>
<td>19.21</td>
<td>57.12</td>
</tr>
<tr>
<td>18</td>
<td>327.57</td>
<td>11.89</td>
<td>12.00</td>
<td>21.47</td>
<td>65.01</td>
</tr>
<tr>
<td>20</td>
<td>518.40</td>
<td>14.49</td>
<td>14.64</td>
<td>23.48</td>
<td>72.16</td>
</tr>
</tbody>
</table>

6. Conclusion

In this paper, we derive a new upper bound for the number of two-reduction steps in LLL algorithm, which is a critical parameter in evaluating the computational complexity of LLL algorithm. Because the newly derived upper bound is expressed by the determinant and the condition number of the given lattice basis matrix relating with its matrix structure, the proposed
TABLE 7. Comparison of the average condition numbers and the average number of two-reduction steps for the row-wise downdating algorithm when $M = N + 4$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\kappa(H_b)$</th>
<th>$\kappa(H_bT_1)$</th>
<th>$\kappa(H_bT_2)$</th>
<th>$N_d$</th>
<th>$N_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>7.98</td>
<td>3.51</td>
<td>3.53</td>
<td>2.20</td>
<td>10.98</td>
</tr>
<tr>
<td>10</td>
<td>9.84</td>
<td>4.37</td>
<td>4.41</td>
<td>3.20</td>
<td>16.36</td>
</tr>
<tr>
<td>12</td>
<td>11.81</td>
<td>5.43</td>
<td>5.46</td>
<td>4.38</td>
<td>22.01</td>
</tr>
<tr>
<td>14</td>
<td>13.63</td>
<td>6.62</td>
<td>6.68</td>
<td>5.60</td>
<td>27.70</td>
</tr>
<tr>
<td>16</td>
<td>15.61</td>
<td>7.98</td>
<td>8.02</td>
<td>6.59</td>
<td>33.56</td>
</tr>
<tr>
<td>18</td>
<td>17.65</td>
<td>9.47</td>
<td>9.57</td>
<td>7.84</td>
<td>38.79</td>
</tr>
<tr>
<td>20</td>
<td>19.28</td>
<td>11.08</td>
<td>11.14</td>
<td>8.89</td>
<td>43.93</td>
</tr>
</tbody>
</table>

upper bound gives an insight into the effects of the lattice structure on the complexities of the LR algorithm. From the numerical results, while the conventional upper bound does not catch that the average number of two-reduction steps decreases when the number of rows increases, the proposed new upper bound decreases as the number of rows increases, which agrees with the changes in the average number of two-reduction steps. That is, from the proposed upper bound, we can infer the behavior of the computational complexity of lattice reduction according to the change of the matrix structure. We have also analyzed the row-wise LLL updating and downdating algorithms when a new basis row is added or when an existing basis row is removed in a given lattice basis matrix. Through the updating and downdating schemes we can eliminate the redundant complexities in finding newly updated preconditioning (unimodular) matrix. Especially, by the new upper bound, we can expect that the required computational complexities would be diminished when the row size becomes large, which is demonstrated by the simulation results.

ACKNOWLEDGMENTS

This work was supported by a Research Grant of Pukyong National University (2015 year).

REFERENCES

ARITHMETIC AVERAGE ASIAN OPTIONS WITH STOCHASTIC ELASTICITY OF VARIANCE

KYU-HWAN JANG 1 MIN-KU LEE2,†

1Hanjin Logistics Institute, Namdaeun-ro 63, Jung-Gu, Seoul, Korea
E-mail address: khjang@hli.re.kr

2Department of Mathematics, Kunsan National University, Kunsan 54150, Korea
E-mail address: mgcorea@kunsan.ac.kr

ABSTRACT. This article deals with the pricing of Asian options under a constant elasticity of variance (CEV) model as well as a stochastic elasticity of variance (SEV) model. The CEV and SEV models are underlying asset price models proposed to overcome shortcomings of the constant volatility model. In particular, the SEV model is attractive because it can characterize the feature of volatility in risky situation such as the global financial crisis both quantitatively and qualitatively. We use an asymptotic expansion method to approximate the no-arbitrage price of an arithmetic average Asian option under both CEV and SEV models. Subsequently, the zero and non-zero constant leverage effects as well as stochastic leverage effects are compared with each other. Lastly, we investigate the SEV correction effects to the CEV model for the price of Asian options.

1. INTRODUCTION

Since the seminal achievement (Black & Scholes [1]) of Black and Scholes on European vanilla options, pricing methods for a lot of complex exotic options also have been developed. This study particularly concerns with the pricing of Asian options among those exotic options. Asian options have a payoff frame more complex than the original European vanilla options because Asian options have a strongly path-dependent feature. The name of Asian is known to originate from the fact that two founders of the first pricing formula used to belong to Asia (Tokyo, Japan) (cf. Wilmott [2]). Due to their averaging property, Asian options could diminish the risk of financial market manipulation of underlying risky assets at the expiration day.

Like many other exotic derivatives, there is no analytic closed form formula for the price of Asian options. So, many researchers have been devoted to develop how to price these options approximately or numerically. For example, Geman and Yor [3], and Linetsky [4] used the
approximation method. Kemna and Vorst [5] used the Monte Carlo method. Ingersoll [6], Rogers and Shi [7], and Večer [8] used the partial differential equation (PDE) method.

On the other hand, the Black–Scholes model (Black & Scholes [1]) corresponding to constant volatility has been extended to fulfill requirements given by practical market phenomena including volatility skew/smile, fat-tailed and asymmetricity of returns distributions, mean-reversion of volatility and etc. For instance, constant elasticity of variance (CEV) model suggested by Cox [9], stochastic volatility models by Heston [10] or Fouque et al. [11], and a Levy model by Carr et al. [12] are among those popular ones. So, it is desirable to price Asian options also based upon these advanced models. In fact, there are quite a number of studies published along the lines of the extension. For example, there are Peng and Peng [13] for the CEV model, Fouque and Han [14] for a mean-reverting stochastic volatility model, and Lemmens et al. [15] for a Levy type model.

This paper chooses not only the CEV model but also the stochastic elasticity of variance (SEV) model of Kim et al. [16] and compares the models by studying the price of an arithmetic average Asian option. In the SEV model, stochastic leverage effect is an important feature of the SEV model. It’s shown that the SEV model gives results of desirable correction to the price under the Black–Scholes model in regard to both dynamics and geometry of the resultant implied volatilities. The model has been utilized to study a perpetual American option (Yoon et al. [17]) and an asset allocation problem (Yang et al. [18]). Further, the model can characterize the feature of volatility during the peak period of the 2007-2008 global financial crisis both quantitatively and qualitatively as shown in Kim et al. [19]. So, it would be quite interesting to derive the value of the Asian option under this model.

This paper is organized as follows. In Section 2, we use the CEV model to characterize the price of an arithmetic average Asian option. Section 3 is devoted to use the SEV model to obtain a dynamic law of the price of the Asian option. In Section 4, we obtain the option price in expanded form. Section 5 investigates the implications of the solution. Section 6 provides concluding remarks.

2. Pricing under CEV model

In this section, we use a PDE method for the price of an Asian floating-strike option based on the CEV model. In fact, Peng and Peng [13] have already studied the pricing of arithmetic Asian options under the CEV model. They used a binomial tree method. In this paper, however, we utilize Večer’s dimension reduction skill given by Večer [8] and some details in Fouque and Han [14] to derive the corresponding option price.

Under a risk-neutral measure \( P^* \), the price \( S_t \) of an underlying asset at time \( t \) follows the CEV model following stochastic differential equation (SDE) given by

\[
dS_t = rS_t dt + \sigma S_t^\theta dW_t^*, \quad t < T, \tag{2.1}
\]

where \( r \) is an interest rate, \( \sigma \) is a volatility coefficient, \( \theta \) is an elasticity parameter, \( W_t^* \) is a Brownian motion. In this paper, we truncate smoothly the value of \( S_t \) in such a way that the
resultant price is bounded and bounded away from zero in the interval \([0, T]\) almost surely. However, we are going to use the same notation \(S_t\) for the resultant price still.

A payoff function for arithmetic average Asian options is defined by

\[
h \left( \frac{1}{T} \int_0^T S_t dt - K_1 S_T - K_2 \right)
\]

for some constants \(K_1\) and \(K_2\). Here, \(h\) is a homogeneous function with the property

\[
h(\alpha x) = \alpha h(x).
\]

Here, if \(K_1 = 0\), (2.2) becomes a payoff for fixed strike Asian options, while, if \(K_2 = 0\), it is a payoff for floating strike Asian options.

Now, we define the Asian option price at \(t = 0\) by

\[
P(0, s; T, K_1, K_2) = \mathbb{E}^* \left[ e^{-rT} h \left( \frac{1}{T} \int_0^T S_u du - K_1 S_T - K_2 \right) | S_0 = s \right],
\]

where \(\mathbb{E}^*\) means the expectation under the risk-neutral measure \(P^*\).

As a preliminary, we compose a portfolio \((\alpha_t, \beta_t)\) whose value is given by

\[
X_t = \alpha_t S_t + \beta_t e^{rt}
\]

to replicate an averaged process \(\frac{1}{T} \int_0^T S_u du\), where \(\alpha_t\) and \(\beta_t\) are numbers of the risky asset and risk-free asset at time \(t\) so that they are to be determined later. Here, \(\alpha_t\) is assumed to be a non-random function. Applying the self-financing strategy, we obtain that from (2.1) and (2.4)

\[
dX_t = \alpha_t dS_t + \beta_t r e^{rt} dt
\]

Since the function \(\alpha_t\) is a non-randomness, we can obtain

\[
d(e^{r(T-t)}\alpha_t S_t) = e^{r(T-t)} \alpha_t (dS_t - r S_t dt) + e^{r(T-t)} S_t d\alpha_t
\]

Then, using equation (2.5) and (2.6), one can have

\[
d(e^{r(T-t)} X_t) = -re^{r(T-t)} X_t dt + e^{r(T-t)} dX_t
\]

Thus, by integrating equation (2.7), one can obtain

\[
X_T = e^{rT} X_0 + \alpha_T S_T - \alpha_0 e^{rT} S_0 - \int_0^T e^{r(T-t)} S_t d\alpha_t
\]
If we choose the trading strategy \( \alpha_t \) and the portfolio value \( X_0 \) at \( t = 0 \) as
\[
\alpha_t = \frac{1 - e^{-r(T-t)}}{rT}, \quad X_0 = x = \frac{1 - e^{-rT}}{rT} S_0 - K_2 e^{-rT},
\]
respectively, then the portfolio value \( X_T \) at \( t = T \) becomes \( \frac{1}{T} \int_0^T S_i \, dt - K_2 \) and so the payoff function (2.2) becomes \( h(X_T - K_1 S_T) \). Refer to Večer [8], and Fouque and Han [14]. Thus the price at \( t = 0 \) can be represented by
\[
P(0, s; T, K_1, K_2) = E^s \left[ e^{-rT} h(X_T - K_1 S_T) \mid S_0 = s \right].
\]
with the portfolio process \( X_t \).

Now, we obtain a dynamic law of the arithmetic Asian option price in a PDE form as shown in the next theorem.

**Theorem 2.1.** By the change of numeraire \( \psi_t := \frac{X_t}{S_t} \), the arithmetic Asian option price at \( t = 0 \) under the CEV model (1) is given by
\[
P(0, s; T, K_1, K_2) = su(0, \psi; T, K_1, K_2),
\]
where \( \psi = \frac{1}{s} = \frac{1 - e^{-rT}}{rT} - \frac{K_2 e^{-rT}}{s} \) and \( u(t, \psi; T, K_1, K_2) \) is the solution of the PDE
\[
u_t + \frac{1}{2}(\psi - \alpha_t)^2 \sigma^2 S^{\theta-2} u_{\psi\psi} = 0
\]
with the final condition \( u(T, \psi; T, K_1, K_2) = h(\psi - K_1) \). Further, the price at arbitrary \( t > 0 \) satisfies
\[
P(t, s; T, K_1, K_2) = \frac{T-t}{T} su(0, \psi; T, K_1, K_2).
\]

**Proof.** From the Itô formula and the SDE (2.1), we obtain the SDEs
\[
d(S_t^{-1}) = S_t^{-1} \left[ (\sigma^2 S_t^{\theta-2} - r) dt - \sigma S_t^{\theta-1} dW_t^* \right], \quad dX_t = S_t \left[ r \psi_t dt + \alpha_t \sigma S_t^{\theta-1} dW_t^* \right].
\]
Then, using the Itô product rule, we have
\[
d \psi_t = X_t d(S_t^{-1}) + S_t^{-1} dX_t + dX_t d(S_t^{-1})
\]
\[
= \psi_t \left[ (\sigma^2 S_t^{\theta-2} - r) dt - \sigma S_t^{\theta-1} dW_t^* \right] + r \psi_t dt + \alpha_t \sigma S_t^{\theta-1} dW_t^* - \alpha_t \sigma^2 S_t^{\theta-2} dt
\]
\[
= \psi_t \left[ \sigma^2 S_t^{\theta-2} dt - \sigma S_t^{\theta-1} dW_t^* \right] - \alpha_t \left[ \sigma S_t^{\theta-2} dt - \sigma S_t^{\theta-1} dW_t^* \right] \quad (2.8)
\]
\[
= \sigma S_t^{\theta-1} (\alpha_t - \psi_t) (dW_t - \sigma S_t^{\theta-1} dt)
\]
\[
= \sigma S_t^{\theta-1} (\alpha_t - \psi_t) dW_t^*,
\]

where $\tilde{W}_t^*$ is given by $d\tilde{W}_t^* = dW_t^* - \sigma S_t^g dt$. From the Girsanov theorem (cf. Oksendal [20]), we change the probability measure $P^*$ into a measure $\tilde{P}^*$ through

$$
\frac{d\tilde{P}^*}{dP^*} = e^{-rT S_T \frac{S_T}{S_0}} = \exp \left[ \int_0^T \sigma S_t^g d\tilde{W}_t^* - \frac{1}{2} \int_0^T \sigma^2 S_t^g \gamma_t^2 dt \right].
$$

(2.9)

Then, from (2.3) and (2.9), we have

$$
P(0, s; T, K_1, K_2) = \tilde{E} \left[ e^{-rT h(X_T - K_1 S_T)} \mid S_0 = s \right]
= s \tilde{E} \left[ e^{-rT \frac{S_T}{S_0} h(\psi_T - K_1)} \mid \psi_0 = \psi \right]
= s \tilde{E} \left[ h(\psi_T - K_1) \mid \psi_0 = \psi \right],
$$

where $\tilde{E}$ denotes expectation with respect to the probability measure $\tilde{P}^*$.

Now, we introduce a function $u$ defined by

$$
u(t, \psi; T, K_1, K_2) = \tilde{E} [h(\psi_T - K_1) \mid \psi_t = \psi].
$$

Then the option price $P(0, s; T, K_1, K_2)$ at $t = 0$ satisfies $P(0, s; T, K_1, K_2) = su(0, \psi; T, K_1, K_2)$.

Here, from the Feynman-Kac formula (cf. Oksendal [20]) and (2.8), $u(t, \psi; T, K_1, K_2)$ satisfies the PDE

$$
u_t + \frac{1}{2} (\psi - \alpha_t) \sigma^2 \gamma_t^2 u_{\psi\psi} = 0
$$

with a final condition given by $u(T, \psi; T, K_1, K_2) = h(\psi - K_1)$. If the solution $u$ is substituted into $P(0, s; T, K_1, K_2) = su(0, \psi; T, K_1, K_2)$, then the option price $P(t, s; T, K_1, K_2)$ at any time $t$ can be immediately determined by the aid of the following identity whose derivation can be found at Fouque and Han [14].

$$
P(t, s; T, K_1, K_2) = \frac{T - t}{T} P(0, s; T, K_1, K_2).
$$

(2.10)

3. Pricing under SEV model

In this section, the constant elasticity of variance is randomized and subsequently a dynamic law of the price of Asian floating-strike option is obtained.

3.1. SEV Formulation. In the stochastic elasticity of variance (SEV) model of Kim et al. [19], the elasticity of variance is assumed by a stochastic process in such a way that

$$
ds_t = r S_t dt + \sigma S_t^\gamma dW_t^*,
\gamma_t = \frac{\theta}{2} + \sqrt{\epsilon} f(Y_t),
\nu = \frac{1}{\epsilon} (m - Y_t) - \frac{\nu \sqrt{2}}{\sqrt{\epsilon}} \Lambda(Y_t)
$$

(3.1)
under a risk-neutral probability measure $P^*$, where $r$, $\sigma$, $\theta$, $m$ and $\nu$ are positive constants, and $W^*_t$ and $\hat{Z}^*_t$ are Brownian motions correlated by $d\langle W^*_t, \hat{Z}^*_t \rangle = \rho dt$, and $\Lambda$ represents the market price of elasticity risk. When $\rho$ is positive, $Y_t$ may explode to infinite, and so $S_t$ may fail to be a true martingale. Refer to Andersen & Piterbarg [24]. Also, based on the analysis of the most financial data excluding some commodity markets, it’s observed that the correlation between stock price and its volatility is negative. So, $\rho$ is assumed to be negative. Furthermore, it is considered as a constant for simplicity. For avoiding the non-existence of moments of $S_t$, we assume that the function $f$ have the condition that $0 < c_1 \leq f \leq c_2 < \infty$ for some constants $c_1$ and $c_2$.

As the Ornstein-Uhlenbeck process (OU) $Y_t$ is an ergodic process with an invariant distribution given by $N(m, \nu^2)$, for the later sections, we denote the average $\langle \cdot \rangle$ with respect to this invariant distribution by

$$\langle g \rangle = \frac{1}{\sqrt{2\pi\nu^2}} \int_{-\infty}^{+\infty} g(y)e^{-\frac{(y-m)^2}{2\nu^2}}dy,$$

for any function $g$.

Now, we suppose that $0 < \epsilon \ll 1$ so that the process $Y_t$ reverts fast a mean. The introduction of this process to finance by Fouque et al. [11] was motivated by an empirical analysis of financial data (example, S&P 500 index). The process gives a quite useful analytic tool to deal with the problems related to the valuation of financial derivatives. Refer to Fouque et al. [21]. Kim et al. [16] used it as a process for the elasticity of variance.

### 3.2. Price Dynamics

In this section, we extend Večer’s dimension reduction technique introduced by Fouque and Han [14] to obtain the option price in the form of two space dimensional PDE. Since the option price under the SEV model depends on the stochastic processes $S_t$ and $Y_t$ (differently from the CEV case), the payoff function for the option has a generalized form given by

$$P(0, s, y; T, K_1, K_2) = E^*[e^{-r(T-t)}h\left(\frac{1}{T} \int_0^T S_u du - K_1 S_T - K_2\right) | S_0 = s, Y_0 = y]$$

(3.2)

under a risk-neutral measure $P^*$, where $h$ has the condition (2.3).

Similarly with the previous section, we want to replicate an averaged process $\frac{1}{T} \int_0^T S_u du$ with a portfolio $X_t = \alpha_t S_t + \beta_t e^{rt}$. Here, $\alpha_t$ is also supposed to be a non-random function. Then applying the self-financing strategy to $X_t$ yields

$$dX_t = rX_t dt + \alpha_t \sigma S_t^{\theta + \sqrt{j(Y_t)}} dW_t^*.$$ 

(3.3)

Let us choose $\alpha_t$ and $X_0$ as $\alpha_t = \frac{1 - e^{-r(T-t)}}{rT}$ and $X_0 = x = \alpha_0 S_0 + e^{-rT} K_2$, respectively. Then $X_T = \frac{1}{T} \int_0^T S_u dt - K_2$, and so the payoff function (3.2) becomes

$$P(0, s, y; T, K_1, K_2) = E^*[e^{-rT}h(X_T - K_1 S_T) | S_0 = s, Y_0 = y]$$

(3.4)

at $t = 0$. 

We define a probability measure $\tilde{P}^*$ by
\[
\frac{d\tilde{P}^*}{dP^*} = e^{-rT} \frac{S_T}{S_0} = \exp \left[ \int_0^T \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} d\tilde{W}_t^* - \frac{1}{2} \int_0^T \sigma^2 S_t^{\theta-2+2\sqrt{t}/(Y_i)} dt \right].
\] (3.5)

Now, we derive the option price as follows:

**Theorem 3.1.** By the change of numeraire $\psi_t := \frac{S_t}{X_t}$, the arithmetic Asian option price at $t = 0$ under the SEV model is given by
\[
P(0, s, y; T, K_1, K_2) = su(0, \psi; y; T, K_1, K_2),
\]
where $\psi = \frac{z}{y} = \frac{1}{y}e^{-rT} - \frac{K_2}{y}e^{-rT}$ and $u(t, \psi, y; T, K_1, K_2)$ satisfies the PDE
\[
u \frac{\partial u}{\partial t} + \frac{\partial u}{\partial y} + \left( \frac{1}{\epsilon}(m - y) - \frac{\nu \sqrt{2}}{\sqrt{\epsilon}}(\Lambda(y) - \rho \sigma \epsilon \psi s^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)}) \right) u_y + \frac{\nu^2}{\epsilon} u_{yy} = 0
\]
with the final condition $u(T, \psi, y; T, K_1, K_2) = h(\psi - K_1)$. Furthermore, the price at arbitrary $t > 0$ satisfies
\[
P(t, s, y; T, K_1, K_2) = \frac{T-t}{T} su(0, s, y; T, K_1, K_2).
\]

**Proof.** By the Itô formula and (3.3), we have the SDEs
\[
d(S_t^{-1}) = S_t^{-1} \left[ (\sigma^2 S_t^{\theta-2+2\sqrt{t}/(Y_i)} - r)dt - \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dW_t^* \right],
\]
\[
dX_t = S_t \left[ r \psi_t dt + \alpha_t \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dW_t^* \right].
\]

Then, we obtain
\[
d\psi_t = X_t d(S_t^{-1}) + S_t^{-1} dX_t + dX_t d(S_t^{-1})
\]
\[
= \psi_t \left[ (\sigma^2 S_t^{\theta-2+2\sqrt{t}/(Y_i)} - r)dt - \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dW_t^* \right] + r \psi_t dt + \alpha_t \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dW_t^* - \alpha_t \sigma^2(Y_t) S_t^{\theta-2+2\sqrt{t}/(Y_i)} dt
\]
\[
= \psi_t \left[ \sigma S_t^{\theta-2+2\sqrt{t}/(Y_i)} dt - \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dW_t^* \right] - \alpha_t \left[ \sigma S_t^{\theta-2+2\sqrt{t}/(Y_i)} dt - \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dW_t^* \right]
\]
\[
= \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} (\alpha_t - \psi_t) dW_t^* - \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} dt
\]
\[
= \sigma S_t^{\frac{1}{2}(\theta-2)+\sqrt{t}/(Y_i)} (\alpha_t - \psi_t) d\tilde{W}_t^*.
\] (3.6)
where $\tilde{W}_t$ is given by $d\tilde{W}_t = dW_t - \sigma S_t^{\frac{1}{2}(\theta - 2) + \sqrt{\psi}(Y_t)} dt$ and it is a Brownian motion under the measure $\tilde{P}$. Since $d\langle W_t, \hat{Z}_t \rangle = \rho dt$, $\hat{Z}_t = \rho W_t + \sqrt{1 - \rho^2}Z_t$ for some independent Brownian motion $Z_t$ with $W_t$. Then the process $Y_t$ satisfies the SDE

$$dY_t = \left[\frac{1}{\epsilon}(m - Y_t) - \frac{\nu \sqrt{2}}{\sqrt{\epsilon}} \left(\Lambda(Y_t) - \rho \sigma S_t^{\frac{1}{2}(\theta - 2) + \sqrt{\psi}(Y_t)} \right)\right] dt$$

$$+ \frac{\nu \sqrt{2}}{\sqrt{\epsilon}} (\rho d\tilde{W}_t + \sqrt{1 - \rho^2}dZ_t).$$

(3.7)

From (2.3), (3.4) and (3.5), we have

$$P(0, s, y; T, K_1, K_2) = \mathbb{E}^*[e^{-rT}h(X_T - K_1 S_T) \mid S_0 = s, Y_0 = y]$$

$$= s \mathbb{E}^*[e^{-rT}T_{S_0}h(\psi_T - K_1) \mid \psi_0 = \psi, Y_0 = y]$$

$$= s \tilde{\mathbb{E}}^*[h(\psi_T - K_1) \mid \psi_0 = \psi, Y_0 = y].$$

If we define a function $u$ by

$$u(t, \psi, y; T, K_1, K_2) := \tilde{\mathbb{E}}^*[h(\psi_T - K_1) \mid \psi_t = \psi, Y_t = y].$$

(3.8)

Then the option price at $t = 0$ can be represented by $P(0, s, y; T, K_1, K_2) = su(0, \psi, y; T, K_1, K_2)$.

Applying the Feynman-Kac formula to (3.6) and (3.7), we obtain a PDE for $u$ of (3.8) as given in Theorem 3.1. Furthermore, the option price at $t > 0$ satisfies

$$P(t, s, y; T, K_1, K_2) = \frac{T - t}{T} P(0, s, y; T, K_1, K_2) = \frac{T - t}{T} su(0, s, y; T, K_1, K_2)$$

by the work of Fouque and Han [14].

4. ASYMPTOTIC EXPANSION

It is difficult to solve the PDE obtained by Theorem 3.1. In this section, we use an asymptotic expansion method to obtain PDEs whose numerical solutions can be computed easily. So, we suppose the solution of the form $u = \sum_{i=0}^{\infty} \epsilon^i u_i$ for solving the PDE problem in Theorem 3.1.

First, the PDE problem for $u$ can be rewritten as

$$u_t + \frac{1}{2}(\psi - \alpha_t)^2 \sigma_t^2 s^{\theta - 2 + 2\sqrt{\psi}(y)} u_{\psi\psi}$$

$$+ \frac{1}{\sqrt{\epsilon}} \left(\rho \nu \sqrt{2}(\alpha_t - \psi)\sigma_t s^{\frac{1}{2}(\theta - 2) + \sqrt{\psi}(y)} u_{\psi y} - \nu \sqrt{2}(\Lambda(y) - \rho \sigma s^{\frac{1}{2}(\theta - 2) + \sqrt{\psi}(y)}) u_y\right)$$

$$+ \frac{1}{\epsilon} \left((m - y)u_y + \nu^2 u_{yy}\right) = 0$$

(4.1)
with the final condition \( u(T, \psi, y; T, K_1, K_2) = h(\psi - K_1) \). Applying the Taylor series expansion (cf. regular perturbation) to the PDE (4.1), we obtain
\[
\left(\frac{1}{\epsilon}L_{00} + \frac{1}{\epsilon^2}L_{10} + (L_{11} + L_{20}) + \sqrt{\epsilon}(L_{12} + L_{21}) + \epsilon(L_{13} + L_{22})\right)u(t, \psi, y) = 0,
\]
where
\[
L_{00} := (m - y)\partial_y + \nu^2\partial_{yy},
\]
\[
L_{11} := \rho\sigma\sqrt{2\nu}2^{\frac{1}{2}(\theta - 2)}\left(\log sf(y)\right)^{\frac{i}{i!}}((\alpha_i - \psi)\partial_{\psi y} - \partial_y) - \delta_i\nu\sqrt{2}\Lambda(y)\partial_y, \quad i = 0, 1, 2, \ldots
\]
\[
L_{20} := \partial_t + \frac{1}{2}(\psi - \alpha_t)^2\sigma^2s^{\theta - 2}\partial_{\psi\psi},
\]
\[
L_{21} := \frac{1}{2}(\psi - \alpha_t)^2\sigma^2s^{\theta - 2}\left(\frac{2\log sf(y)}{i!}\right)^{i}\partial_{\psi\psi}, \quad i = 1, 2, \ldots
\]
where \( \delta_0 = 1 \) and \( \delta_i = 0 \) for \( i = 1, 2, \ldots \) Putting the expansion \( u = \sum_{i=0}^{\infty} \epsilon^\frac{i}{2} u_i \) into the PDE (4.2) and the final condition, we obtain
\[
\frac{1}{\epsilon}L_{00}u_0 + \frac{1}{\epsilon^2}(L_{00}u_1 + L_{10}u_0) + (L_{00}u_2 + L_{10}u_1 + L_{11}u_0 + L_{20}u_0) + \sqrt{\epsilon}(L_{00}u_3 + L_{10}u_2 + L_{11}u_1 + L_{20}u_1 + L_{21}u_0) = 0
\]
with the final condition \( \sum_{i=0}^{\infty} \epsilon^\frac{i}{2} u_i(T, \psi, y; T, K_1, K_2) = h(\psi - K_1) \).

Now, we obtain desirable PDEs for \( u_0 \) and \( u_1 \).

**Theorem 4.1.** Suppose that \( u_i \), \( i = 0, 1, 2, \ldots \), does not increase as much as \( u_i \sim e^{\frac{y^2}{2}} \) as \( y \) goes to infinity. Then \( u_0 \) is a \( y \)-independent function and satisfies the PDE
\[
L_{20}u_0(t, \psi; T, K_1, K_2) = 0, \quad t < T,
\]
\[
u_0(T, \psi; T, K_1, K_2) = h(\psi - K_1).
\]

**Proof.** The \( \frac{1}{2} \) term of (4.3) yields the following equation
\[
L_{00}u_0 = 0.
\]
The assumed growth condition on \( u_0 \) leads that the solution \( u_0 \) of this ODE doesn’t depend on \( y \), i.e., \( u_0 = u_0(t, \psi) \). The \( \frac{1}{\sqrt{\epsilon}} \) term of (4.3) gives
\[
L_{00}u_1 + L_{10}u_0 = 0.
\]
Since the operator \( L_{10} \) has the partial derivative with respect to \( y \) in its every terms and \( u_0 \) does not depend on the variable \( y \), we have \( L_{00}u_1 = 0 \) and so \( u_1 \) is independent of the \( y \) variable. By the \( y \)-independence of \( u_0 \) and \( u_1 \), the \( \mathcal{O}(1) \) term of (4.3) becomes the PDE
\[
L_{00}u_2 + L_{20}u_0 = 0.
\]
From the solvability condition (cf. Ramm [22]) of this Poisson equation, we have \( L_{20}u_0 = 0 \). Therefore, Theorem 4.1 is proved. \( \square \)
Continuously, we derive a PDE for the correction term $u_1$ from the following theorem.

**Theorem 4.2.** Suppose that $u_i$, $(i = 0, 1, 2, ...)$, does not increase as much as $u_i \sim e^{\frac{t^2}{2}}$ as $t$ goes to infinity. Then $u_1$ is a $y$-independent function and satisfies the PDE

$$L_{20}u_1(t, \psi; T, K_1, K_2) = -\langle L_{21} \rangle u_0(t, \psi; T, K_1, K_2), \quad t < T,$$

$$u_1(T, \psi; T, K_1, K_2) = 0,$$

where $u_0$ is obtained by Theorem 4.1.

**Proof.** We have already found that $u_0$ and $u_1$ are independent on the variable $y$ from the proof of Theorem 4.1. Also, $u_2$ is independent of $y$ since the proof process of Theorem 4.1 gives $L_{00}u_2 = 0$. Then, from the $\sqrt{\tau}$ term of (4.3), we obtain $L_{00}u_3 + L_{20}u_1 + L_{21}u_0 = 0$ and the solvability condition of this Poisson equation for $u_3$ yields $L_{20}u_1 + \langle L_{21} \rangle u_0 = 0$. Hence, we obtain Theorem 4.2. $\qquad \square$

### 5. Approximate Option Price

This section investigates the influence of the stochastic elasticity of variance on the constant elasticity of variance by using a numerical experiment.

In Section 4, we derived the first order approximation for $u$ given by $u(t, \psi; y; T, K_1, K_2) \approx u_0(t, \psi; T, K_1, K_2) + \sqrt{\tau}u_1(t, \psi; T, K_1, K_2)$. We define the leading order term $P_0$ and the first order term $P_1$ by

$$P_0(t, s; T, K_1, K_2) = \frac{T - t}{T}u_0(0, \psi; T, K_1, K_2),$$

$$P_1(t, s; T, K_1, K_2) = \frac{T - t}{T}u_1(0, \psi; T, K_1, K_2),$$

so that the option price $P(t, s; y; T, K_1, K_2)$ at time $t$ has the approximation

$$P(t, s; y; T, K_1, K_2) \approx P_0(t, s; T, K_1, K_2) + \sqrt{\tau}P_1(t, s; T, K_1, K_2).$$

The accuracy of this approximation depends on the property of the payoff $h$. If the payoff is sufficiently smooth, it is straightforward to find an approximation error in the pointwise convergent sense. Otherwise, it needs a regularization of the payoff function as the case of European vanilla options in Fouque et al. [23]. This paper checks a numerical error instead of the theoretical proof of accuracy. See Figure 1 (c).

Now, we solve the PDEs for the leading order $P_0$ and the correction $\tilde{P}_1 := \sqrt{\tau}P_1$ by using the finite difference method (the Crank-Nicolson method). The solution has the truncation error given by $O((\Delta t)^2) + O((\Delta \psi)^2)$ with $\Delta = 0.005$ and $\psi = 0.0104$. Here, the parameters are given by $r = 0.06$, $\sigma = 0.5$, $T = 1$, $K_1 = 0$, $K_2 = 2$, $\epsilon = 0.01$, and $f(y) = e^y$ is chosen. Based on the observed financial data (such as S&P 500 index), we choose the value of parameter $\theta$ by three values $\theta = 1.8$, $\theta = 2$, and $\theta = 2.2$.

Figure 1 (a) shows the Asian option price $P_0$ underlying the CEV model and Figure 1 (b) shows the approximate price $P_0 + \tilde{P}_1$ under the SEV model. The CEV price also increases as the elasticity $\theta$ becomes larger but the stochastic elasticity of variance tends to lower the
increase. In fact, Figure 1 (c) shows the correction term $\tilde{P}_1$ which is negative and has a hump shape in every case of $\theta$. Here, the sign of $\tilde{P}_1$ is determined by the choice of $f$, i.e., $\tilde{P}_1$ has the opposite sign to the sign of $\langle f \rangle$. Note that the lowering effect has a maximum value near the strike price $K_2$ and it is more pronounced as $\theta$ becomes larger.

Figure 2 shows the sensitivity of the correction term $\tilde{P}_1$ to the asymptotic parameters $\bar{f}$. One can observe that the correction term $\tilde{P}_1$ decrease as the asymptotic parameter $\bar{f}$ increases and the slope is more large when the parameter $\theta$ is large. So one can figure out that the correction term $\tilde{P}_1$ is more sensitive to the asymptotic parameter as the parameter $\theta$ increases.
The sensitivity of $\tilde{P}_1$ to the asymptotic parameter $\tilde{f}$; $S_0 = 2.5$, $r = 0.06$, $\sigma = 0.5$, $T = 1$, $K_1 = 0$, and $K_2 = 1.6$.

6. Conclusion

The SEV model has been devised based upon a direct observation on the market elasticity of variance and successfully contributed to making up for the limitation of the CEV model for the European vanilla option (path-independent option) price. From the standpoint of this success, we price an Asian option (one of typical path-dependent options) under the SEV model by using a dimension reduction technique and a singular-regular perturbation method. This study finds that the CEV option price is somewhat over priced regardless of the elasticity parameter $\theta$. The degree of over valuation has a maximum value near the strike price $K_2$. This may give a remarkable feature of the SEV model in that Gamma, which is important due to the fact that it corrects for the convexity of option value, is greatest approximately at-the-money.

References

ARITHMETIC ASIAN OPTIONS WITH SEV


ROBUST A POSTERIORI ERROR ESTIMATOR FOR LOWEST-ORDER FINITE ELEMENT METHODS OF INTERFACE PROBLEMS

KWANG-YEON KIM

DEPARTMENT OF MATHEMATICS, KANGWON NATIONAL UNIVERSITY, CHUNCHEON, SOUTH KOREA
E-mail address: eulerkim@kangwon.ac.kr

ABSTRACT. In this paper we analyze an a posteriori error estimator based on flux recovery for lowest-order finite element discretizations of elliptic interface problems. The flux recovery considered here is based on averaging the discrete normal fluxes and/or tangential derivatives at midpoints of edges with weight factors adapted to discontinuous coefficients. It is shown that the error estimator based on this flux recovery is equivalent to the error estimator of Bernardi and Verfürth based on the standard edge residuals uniformly with respect to jumps of the coefficient between subdomains. Moreover, as a byproduct, we obtain slightly modified weight factors in the edge residual estimator which are expected to produce more accurate results.

1. INTRODUCTION

Let $\Omega \subset \mathbb{R}^2$ be an open bounded polygonal domain with the boundary $\partial \Omega = \Gamma_D \cup \Gamma_N$, $\Gamma_D \cap \Gamma_N = \emptyset$. We consider the second-order elliptic equation

$$-\nabla \cdot (\alpha \nabla u) = f \quad \text{in } \Omega$$

subject to the boundary conditions

$$u = u_D \quad \text{on } \Gamma_D, \quad (\alpha \nabla u) \cdot n = g_N \quad \text{on } \Gamma_N,$$

where the diffusion coefficient $\alpha$ is a positive scalar-valued function and $n$ denotes the outward unit normal vector on $\partial \Omega$.

The variational formulation of the problem (1.1)–(1.2) is to find $u \in H^1(\Omega)$ such that $u = u_D$ on $\Gamma_D$ and

$$\int_{\Omega} \alpha \nabla u \cdot \nabla v \, dx = \int_{\Omega} f v \, dx + \int_{\Gamma_N} g_N v \, ds \quad \forall v \in H^1_D(\Omega),$$

where

$$H^1_D(\Omega) = \{ v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_D \}.$$

We assume that the coefficient $\alpha$ is piecewise constant with respect to a partition of the domain $\Omega$ into disjoint polygonal subdomains $\{\Omega_i\}_{i=1}^n$ and may have large jumps between
subdomains. In this case the problem (1.1)–(1.2) is often referred to as an interface problem, and it is important to observe that the following continuity conditions hold across the edges of the triangulation:

(NC) The normal flux $\alpha \nabla u \cdot n$ is continuous across the edges, i.e.,
$$\alpha \nabla u \in H(\text{div}; \Omega) = \{ \tau \in (L^2(\Omega))^2 : \nabla \cdot \tau \in L^2(\Omega) \}.$$

(TC) The tangential derivative $\nabla u \cdot t$ is continuous across the edges, i.e.,
$$\nabla u \in H(\text{curl}; \Omega) = \{ \tau \in (L^2(\Omega))^2 : \text{curl} \tau \in L^2(\Omega) \}.$$

Notice also that neither $\nabla u \cdot n$ nor $\alpha \nabla u \cdot t$ is continuous across the edges where $\alpha$ has a jump. This implies that the global regularity of the solution $u$ is at best $H^{3/2-\epsilon}(\Omega)$ for any $\epsilon > 0$.

In this paper we deal with three lowest-order finite element methods, namely, the $P_1$ conforming/nonconforming FEMs and the lowest-order Raviart–Thomas ($RT_0$) mixed FEM over triangular meshes. One popular tool for efficient implementation of finite element methods is adaptive mesh refinement guided by an a posteriori error estimator which can provide quantitative and/or qualitative information on the numerical error. An error estimator for the interface problem (1.1)–(1.2) consisting of standard element and edge residuals was first proposed by Bernardi and Verfürth [1] for the $P_1$ conforming FEM; see also [2]. The key result is that appropriate weight factors depending on $\alpha$ should be incorporated into the residuals in order for the error estimator to be robust with respect to the jumps of $\alpha$ between subdomains. Moreover, following the technique of [3] for the Poisson equation, Cai and Zhang [4, 5] showed that element residuals are redundant, i.e., edge residuals dominate even for the interface problem (1.1)–(1.2).

Since the pioneering work of Zienkiewicz and Zhu [6], the error estimator based on gradient ($\nabla u$) or flux ($\alpha \nabla u$) recovery has been widely used and well investigated. The most popular recovery procedure in the lowest-order FEMs is based on averaging either the flux or gradient at vertices of the triangulation (cf. [6, 7]). Gradient recovery by averaging at midpoints of edges was discussed in [8, 9] for the $P_1$ conforming FEM, in Brandts [10] for the $RT_0$ mixed FEM and then in [11] for the $P_1$ nonconforming FEM. In either way the recovered flux or gradient is a piecewise linear polynomial and is superconvergent under favorable conditions (such as uniform triangulations and smooth solutions), in which case we get the asymptotically exact error estimator. But the error estimator is not well-suited for the interface problem, as illustrated by the numerical results of [4, 5]. On the other hand, in consideration of the two continuity conditions (NC) and (TC) stated above, Cai and Zhang [4, 5] recently proposed an error estimator using flux recovery in $H(\text{div}; \Omega)$ and/or gradient recovery in $H(\text{curl}; \Omega)$ which is achieved either globally (and implicitly) by the $L^2$ projection or locally (and explicitly) by averaging on edges with weight factors depending on the coefficient $\alpha$. Numerical results in [4, 5] show that the error estimator using the explicit recovery is fairly accurate but not asymptotically exact even under the most favorable conditions.

In this paper we adapt the recovery procedure based on averaging at midpoints of edges to the interface problem (1.1)–(1.2), inspired by the works of Cai and Zhang [4, 5]. The adapted
procedure recovers the flux variable, but it also involves gradient recovery to reflect the continuity of the tangential derivative. More specifically, we combine flux averaging in the normal direction and gradient averaging in the tangential direction at the midpoint of each edge to construct a piecewise linear flux approximation (but not in $H(\text{div}; \Omega)$). This flux recovery is applicable to the $P1$ conforming/nonconforming FEMs and the $RT_0$ mixed FEM in a unified way, and we show that the error estimator based on the recovered flux is equivalent to the edge residual error estimator of Bernardi and Verfürth uniformly with respect to the jumps of the coefficient $\alpha$ between subdomains. In fact, it turns out that our recovery-based error estimator is nothing but the edge residual error estimator with slightly modified weight factors. Thus this error estimator is not only robust with respect to the jumps of $\alpha$ between subdomains (under the monotonicity condition of Bernardi and Verfürth), but also expected to be as inexpensive as and more accurate than the edge residual error estimator.

The remainder of the paper is organized as follows. In Section 2 we introduce some notation to be used throughout the paper and define the three lowest-order FEMs. In Section 3 we review the edge residual error estimators for the interface problem. In Section 4 we present the flux recovery based on averaging at midpoints of edges and the corresponding error estimator which is shown to be uniformly equivalent to the edge residual error estimator. Finally, in Section 5, some numerical results are provided to illustrate the performance of our error estimator.

2. Notation and Finite Element Methods

Let $\mathcal{T}_h = \{T\}$ be a shape-regular triangulation of $\Omega$ into triangular elements such that $\overline{\Omega} = \bigcup_{T \in \mathcal{T}_h} T$, where $h_T = \text{diam}(T)$ and $h = \max_{T \in \mathcal{T}_h} h_T$. It is assumed that the subdomain interface $\Gamma_s := \left( \bigcup_{i=1}^n \partial \Omega_i \right) \setminus \partial \Omega$ is aligned with the triangulation $\mathcal{T}_h$, i.e., $\Gamma_s$ does not pass through the interior of any element $T \in \mathcal{T}_h$.

We denote the collection of all vertices of $\mathcal{T}_h$ by $\mathcal{N}_h = \{z\}$ and the collection of all edges of $\mathcal{T}_h$ by $\mathcal{E}_h = \{e\}$, where $h_e = \text{diam}(e)$. For a vertex $z \in \mathcal{N}_h$, let $\omega_z$ be the union of all elements of $\mathcal{T}_h$ sharing $z$. For an edge $e \in \mathcal{E}_h$, let $\omega_e$ be the union of at most two elements of $\mathcal{T}_h$ sharing $e$. We set $\omega_e = T^+_e \cup T^-_e$ for an interior edge $e$ and $\omega_e = T_e$ for a boundary edge $e$.

With each edge $e \in \mathcal{E}_h$, we associate a fixed unit normal vector $n_e = (n_1, n_2)$ and a unit tangent vector $t_e = (-n_2, n_1)$ such that $n_e$ is oriented from $T^+_e$ to $T^-_e$ for an interior edge $e$ and is outward to $T_e$ for a boundary edge $e$. For a piecewise smooth function $v$, we define the jump of $v$ across an interior edge $e$ as

$$\llbracket v \rrbracket_e = v\big|_{T^+_e} - v\big|_{T^-_e},$$

where the index $e$ is suppressed whenever there is no confusion.

The restriction of $\alpha$ to an element $T \in \mathcal{T}_h$ is denoted by $\alpha_T \equiv \alpha|_T$. Let $\tilde{f}_S := \frac{1}{|S|} \int_S f \, dx$, where $|S|$ is the area of a domain $S \subset \mathbb{R}^2$, and let $\tilde{f}_h$ be the piecewise constant function such that $\tilde{f}_h|_T \equiv \tilde{f}_T$ for all $T \in \mathcal{T}_h$. The notation $\nabla h$ represents the gradient operator taken piecewise over the triangulation $\mathcal{T}_h$.

Now we define the lowest-order FEMs for the problem (1.1)–(1.2). Let $P_k(T)$ be the space of all polynomials of degree at most $k$ on an element $T$. Based on the variational formulation
(1.3), the $P1$ conforming FEM is to find $u_h^{c} \in S_h^{c}$ such that $u_h^{c}(z) = u_D(z)$ at each vertex $z \in \mathcal{N}_h \cap \Gamma_D$ and
\[ \int_{\Omega} \alpha \nabla u_h^{c} \cdot \nabla v_h \, dx = \int_{\Omega} f v_h \, dx + \int_{\Gamma_N} g_N v_h \, ds \quad \forall v_h \in S_h^{c}, \quad (2.1) \]
where
\[ S_h^{c} = \left\{ v_h \in H^1(\Omega) : v_h|_T \in \mathbb{P}_1(T) \quad \forall T \in \mathcal{T}_h \right\}, \]
\[ S_h^{c,D} = \left\{ v_h \in S_h^{c} : v_h|_{\Gamma_D} = 0 \right\}. \]
Similarly, the $P1$ nonconforming FEM is to find $u_h^{nc} \in S_h^{nc}$ such that \[
\int_{\Omega} \alpha \nabla u_h^{nc} \cdot \nabla v_h \, dx = \int_{\Omega} f v_h \, dx + \int_{\Gamma_N} g_N v_h \, ds
\forall v_h \in S_h^{nc}, \quad (2.2)\]
where
\[ S_h^{nc} = \left\{ v_h \in L^2(\Omega) : v_h|_T \in \mathbb{P}_1(T) \quad \forall T \in \mathcal{T}_h \right\}, \]
\[ S_h^{nc,D} = \left\{ v_h \in S_h^{nc} : \int_{e} v_h \, ds = 0 \quad \forall e \subset \Gamma_D \right\}. \]

On the other hand, the RT$^0$ mixed FEM is based on the first-order mixed system of the problem (1.1)–(1.2) having the form
\[ \sigma = \alpha \nabla u, \quad \nabla \cdot \sigma = -f \quad \text{in} \ \Omega, \quad u|_{\Gamma_D} = u_D, \quad \sigma \cdot n|_{\Gamma_N} = g_N, \]
and is given as follows: find $(\sigma_h^{m}, u_h^{m}) \in RT_h \times W_h$ such that $\int_{e} \sigma_h^{m} \cdot n_e \, ds = \int_{e} g_N \, ds$ for each edge $e \subset \Gamma_D$ and
\[ \begin{cases} \int_{\Omega} \alpha^{-1} \sigma_h^{m} \cdot \tau_h \, dx + \int_{\Omega} \nabla \cdot \tau_h u_h^{m} \, dx = \int_{\Gamma_D} u_D \tau_h \cdot n \, ds & \forall \tau_h \in RT_{h,N}, \\ \int_{\Omega} \nabla \cdot \sigma_h^{m} w_h \, dx = -\int_{\Omega} f w_h & \forall w_h \in W_h, \end{cases} \quad (2.3) \]
where
\[ RT_h = \{ \tau_h \in H(\text{div}; \Omega) : \tau_h|_T \in (\mathbb{P}_0(T))^2 + x \mathbb{P}_0(T) \quad \forall T \in \mathcal{T}_h \}, \]
\[ RT_{h,N} = \{ \tau_h \in RT_h : \tau_h \cdot n|_{\Gamma_N} = 0 \}, \]
\[ W_h = \{ w_h \in L^2(\Omega) : w_h|_T \in \mathbb{P}_0(T) \quad \forall T \in \mathcal{T}_h \}. \]

Finally, $C$ will stand for a positive generic constant which is independent of the jumps of $\alpha$ between subdomains as well as the mesh size $h$. 

3. Edge Residual Error Estimators

From now on we assume for simplicity that $u_D$ is continuous and piecewise linear over $\Gamma_D$ and $g_N$ is piecewise constant over $\Gamma_N$.

Suppose that the flux approximation $\sigma_h \approx \sigma = \alpha \nabla u$ has been obtained from one of the lowest-order FEMs as follows:

(a) $\sigma_h = \alpha \nabla u_h^c$, where $u_h^c$ is the solution of the $P1$ conforming FEM (2.1) such that $u_h^c|_{\Gamma_D} = u_D$. Note that the discrete tangential derivative $\alpha^{-1} \sigma_h \cdot t_e$ is continuous across interior edges of $T_h$, but the discrete normal flux $\sigma_h \cdot n_e$ is not.

(b) $\sigma_h = \alpha \nabla_h u_h^{nc}$, where $u_h^{nc}$ is the solution of the $P1$ nonconforming FEM (2.2). Note that neither the discrete tangential derivative $\alpha^{-1} \sigma_h \cdot t_e$ nor the discrete normal flux $\sigma_h \cdot n_e$ is continuous across interior edges of $T_h$.

(c) $\sigma_h = \sigma_h^m$, where $\sigma_h^m$ is the vector solution of the $RT_0$ mixed FEM (2.3) such that $\sigma_h^m \cdot n|_{\Gamma_N} = g_N$. Note that the discrete normal flux $\sigma_h \cdot n_e$ is continuous across interior edges of $T_h$, but the discrete tangential derivative $\alpha^{-1} \sigma_h \cdot t_e$ is not.

Using the flux approximation $\sigma_h$ obtained as above, we define the following error estimator based on edge residuals (cf. [1, 2, 3, 4, 5])

$$\eta_E = \left( \sum_{e \in \mathcal{E}_h} \eta_e^2 \right)^{1/2}$$

with

$$\eta_e^2 = \begin{cases} \frac{2}{\alpha_{T_e} + \alpha_{T_e}^{-1}} h_e \| \sigma_h \cdot n_e \|_{0,e}^2 + \frac{2 \alpha_{T_e}^+ \alpha_{T_e}^{-1}}{\alpha_{T_e}^+ + \alpha_{T_e}^{-1}} h_e \| (\alpha^{-1} \sigma_h \cdot t_e) \|_{0,e}^2 & \text{for } e \subset \Omega, \\
\alpha_{T_e} h_e \| (\alpha^{-1} \sigma_h)_{T_e} \cdot t_e - \frac{d u_D}{d s} \|_{0,e}^2 & \text{for } e \subset \Gamma_D, \\
\alpha_{T_e}^{-1} h_e \| \sigma_h|_{T_e} \cdot n_e - g_N \|_{0,e}^2 & \text{for } e \subset \Gamma_N, \end{cases}$$

where $\frac{du}{ds}|_e$ denotes the directional derivative of $w|_e$ in the direction of $t_e$. We remark that the local contribution $\eta_e$ vanishes for all $e \subset \Gamma_D$ when $\sigma_h$ is computed from the $P1$ conforming FEM and vanishes for all $e \subset \Gamma_N$ when $\sigma_h$ is computed from the $RT_0$ mixed FEM. (It does not vanish but becomes a higher-order term if $u_D$ is not piecewise linear and/or $g_N$ is not piecewise constant.)

The weight factor $\frac{2}{\alpha_{T_e} + \alpha_{T_e}^{-1}}$ (which is equivalent to $\frac{1}{\max\{\alpha_{T_e}^+, \alpha_{T_e}^{-1}\}}$) for the normal jump $h_e \| \sigma_h \cdot n_e \|_{0,e}^2$ was first presented in [1] for the $P1$ conforming FEM and the weight factor $\frac{2 \alpha_{T_e}^+ \alpha_{T_e}^{-1}}{\alpha_{T_e}^+ + \alpha_{T_e}^{-1}}$ (which is equivalent to $\min\{\alpha_{T_e}^+, \alpha_{T_e}^{-1}\}$) for the tangential jump $h_e \| (\alpha^{-1} \sigma_h \cdot t_e) \|_{0,e}^2$ was adopted in [12] to control the jump $h_e^{-1} \| u_h^{nc} \|_{0,e}^2$ for the $P1$ nonconforming FEM. With these weight factors, the error estimator $\eta_E$ is reliable and efficient independently of the jumps of the coefficient $\alpha$ between subdomains under the following condition stated in [1].
Monotonicity Condition: For any two different subdomains $\Omega_i$ and $\Omega_j$, which share at least one point, there is a connected path from $\Omega_i$ to $\Omega_j$ through adjacent subdomains such that the coefficient $\alpha$ is monotone along this path.

When this condition is satisfied, we can obtain the robust global upper bound (extending the result of [4, 5] to the nonhomogeneous case $u_D \neq 0$, $g_N \neq 0$)

$$\|\alpha^{-1/2}(\sigma - \sigma_h)\|_{0,\Omega} \leq C(\eta_E + H_f)$$

with the extra term $H_f$ given by

$$H_f = \left( \sum_{z \in \bar{\Gamma}_h \cap (\Gamma_p \cup \Gamma_D)} \sum_{T \subset \omega_z} \alpha_T^{-1} h_T^2 \|f\|_{0,T}^2 + \sum_{z \in \bar{\Gamma}_h \setminus (\Gamma_p \cup \Gamma_D)} \sum_{T \subset \omega_z} \alpha_T^{-1} h_T^2 \|f - \bar{f}_\omega\|_{0,T}^2 \right)^{1/2}$$

(3.1)

for the $P1$ conforming/nonconforming FEMs (the second term is a higher-order term for $f \in L^2(\Omega)$ and so is the first term for $f \in L^p(\Omega)$ with $p > 2$; see [3]) and

$$H_f = \left( \sum_{T \in \bar{T}_h} \alpha_T^{-1} h_T^2 \|f - \bar{f}_T\|_{0,T}^2 \right)^{1/2}$$

(3.2)

for the $RT_0$ mixed FEM.

Furthermore, without any assumption on the distribution of $\alpha$, the following robust lower bound was established in [1, 2, 5]

$$\eta_E \leq C(\|\alpha^{-1/2}(\sigma - \sigma_h)\|_{0,\omega_e} + H_f)$$

(3.3)

with the extra term $H_f$ given by

$$H_f = \left( \sum_{T \subset \omega_e} \alpha_T^{-1} h_T^2 \|f - \bar{f}_T\|_{0,T}^2 \right)^{1/2}$$

for the $P1$ conforming/nonconforming FEMs and

$$H_f = 0$$

for the $RT_0$ mixed FEM.

While the discrete normal flux $\sigma_h \cdot n_e$ is constant on each edge $e$ for all choices of $\sigma_h$ given above, the discrete tangential derivative $\alpha^{-1}\sigma_h \cdot t_e$ is not constant on $e$ for the $RT_0$ mixed FEM. Fortunately, in place of $\alpha^{-1}\sigma_h \cdot t_e$, one may use the nodal value of $\alpha^{-1}\sigma_h \cdot t_e$ at the midpoint $m_e$ of the edge $e$ and modify $\eta_E$ as

$$\eta_E' = \left( \sum_{e \in \mathcal{E}_h} \eta_e'^2 \right)^{1/2}$$

(3.4)
with the local contributions
\[
\tilde{\eta}_e^2 = \begin{cases} 
\frac{2}{\alpha_{T_e^+} + \alpha_{T_e^-}} h_e \| \sigma_h \cdot n_e \|_{0,e}^2 + \frac{2 \alpha_{T_e^+} \alpha_{T_e^-}}{\alpha_{T_e^+} + \alpha_{T_e^-}} h_e \| (\alpha^{-1} \sigma_h(m_e) \cdot t_e) \|_{0,e}^2 & \text{for } e \subset \Omega, \\
\alpha_{T_e} h_e \| (\alpha^{-1} \sigma_h)|_{T_e}(m_e) \cdot t_e - \frac{du_D}{ds} \|_{0,e}^2 & \text{for } e \subset \Gamma_D, \\
\alpha_{T_e}^{-1} h_e \| \sigma_h|_{T_e} \cdot n_e - g_N \|_{0,e}^2 & \text{for } e \subset \Gamma_N.
\end{cases}
\]

Such a modification only entails a higher order perturbation and thus does not affect the global upper bound (3.1) and the local lower bound (3.4), which is stated in the following theorem.

**Theorem 3.1.** Let \( \sigma_h = \sigma_h^\text{m} \) be the vector solution of the RT0 mixed FEM (2.3). Then we have \( \tilde{\eta}_e \leq \eta_e \) for all \( e \in \mathcal{E}_h \) and
\[
\eta_E \leq \tilde{\eta}_E + C \tilde{H}_f,
\]
where the extra term \( \tilde{H}_f \) is the sum of (3.2) and (3.3).

**Proof.** Following the proof of [13, Theorem 7.1], we can see that
\[
h_e \| \alpha^{-1} \sigma_h \cdot t_e \|_{0,e}^2 = h_e \| (\alpha^{-1} \sigma_h(m_e) \cdot t_e) \|_{0,e}^2 + \frac{1}{48} h_e^3 \| (\alpha^{-1} \tilde{f}_h) \|_{0,e}^2
\]
for every edge \( e \subset \Omega \) and
\[
h_e \| \alpha^{-1} \sigma_h \cdot t - \frac{du_D}{ds} \|_{0,e}^2 = h_e \| (\alpha^{-1} \sigma_h)|_{T_e}(m_e) \cdot t_e - \frac{du_D}{ds} \|_{0,e}^2 + \frac{1}{48} h_e^3 \| (\alpha^{-1} \tilde{f}_T) \|_{0,e}^2
\]
for every edge \( e \subset \Gamma_D \). This immediately gives \( \tilde{\eta}_e \leq \eta_e \) for all \( e \in \mathcal{E}_h \).

To prove the second result, it is sufficient to show that
\[
\sum_{e \subset \Omega} \frac{\alpha_{T_e^+} \alpha_{T_e^-}}{\alpha_{T_e^+} + \alpha_{T_e^-}} h_e^2 \| (\alpha^{-1} \tilde{f}_h) \|_{0,e}^2 + \sum_{e \subset \Gamma_D} \alpha_{T_e} h_e^3 \| (\alpha^{-1} \tilde{f}_T) \|_{0,e}^2 \leq C \tilde{H}_f^2.
\]
By using the inequality \( \frac{\alpha_{T_e^+} \alpha_{T_e^-}}{\alpha_{T_e^+} + \alpha_{T_e^-}} \leq \min\{\alpha_{T_e^+}, \alpha_{T_e^-}\} \), we easily obtain for the edges \( e \subset \Gamma_s \cup \Gamma_D \)
\[
\sum_{e \subset \Gamma_s} \frac{\alpha_{T_e^+} \alpha_{T_e^-}}{\alpha_{T_e^+} + \alpha_{T_e^-}} h_e^2 \| (\alpha^{-1} \tilde{f}_h) \|_{0,e}^2 + \sum_{e \subset \Gamma_D} \alpha_{T_e} h_e^3 \| (\alpha^{-1} \tilde{f}_T) \|_{0,e}^2 \leq C \sum_{e \subset \Gamma_s \cup \Gamma_D} \sum_{T \subset \omega_e} \alpha_{T_e} h_e^3 \| (\alpha^{-1} \tilde{f}_T) \|_{0,T}^2. \quad (3.6)
\]
For the remaining edges \( e \subset \Omega \setminus \Gamma_s \), we have \( \alpha_{T_e^+} = \alpha_{T_e^-} \) and thus it follows that
\[
\sum_{e \subset \Omega \setminus \Gamma_s} \frac{\alpha_{T_e^+} \alpha_{T_e^-}}{\alpha_{T_e^+} + \alpha_{T_e^-}} h_e^3 \| (\alpha^{-1} \tilde{f}_h) \|_{0,e}^2 = \frac{1}{2} \sum_{e \subset \Omega \setminus \Gamma_s} \alpha_{T_e^-} h_e^3 \| (\tilde{f}_h) \|_{0,e}^2
\]
\[
\leq C \sum_{e \subset \Omega \setminus \Gamma_s} \sum_{T \subset \omega_e} \alpha_{T_e^-} h_e^3 \| (\tilde{f}_T) \|_{0,T}^2. \quad (3.6)
\]
For each vertex $z \in N_h \setminus (\Gamma_s \cup \Gamma_D)$, we further obtain
\[
\sum_{e \in \partial \omega_z \setminus \partial \omega_s} \alpha_{T_e}^{-1} h_e^3 \| \tilde{f}_e \|_{0,e}^2 = \sum_{e \in \partial \omega_z \setminus \partial \omega_s} \alpha_{T_e}^{-1} h_e^3 \| [\tilde{f}_e - \tilde{f}_n] \|_{0,e}^2 \\
\leq C \sum_{T \in \mathcal{O}} \alpha_T^{-1} h_T^2 \| \tilde{f}_e - \tilde{f}_n \|_{0,T}^2,
\]
which yields
\[
\sum_{e \in \Omega \setminus \Gamma_s} \alpha_{T_e}^{+} \alpha_{T_e}^{-} h_e^3 \| [\alpha^{-1} f_e] \|_{0,e}^2 \leq C \sum_{z \in N_h \setminus (\Gamma_s \cup \Gamma_D)} \sum_{T \in \mathcal{O}} \alpha_T^{-1} h_T^2 \| f - \tilde{f}_n \|_{0,T}^2 + C \sum_{T \in \mathcal{T}_h} \alpha_T^{-1} h_T^2 \| f - \tilde{f}_n \|_{0,T}^2.
\]

The proof is completed by combining (3.6) and (3.7). \hfill \Box

4. Recovery-based Error Estimator

In this section we present and analyze an error estimator based on flux recovery which is applicable to the $P1$ conforming/nonconforming FEMs and the $RT_0$ mixed FEM in a unified way for the interface problem (1.1)–(1.2). The flux recovery procedure given below is an extension of the gradient recovery by averaging at midpoints of edges discussed in [8, 9, 10, 11] to the case of piecewise constant diffusion coefficients.

Suppose that the flux approximation $\sigma_h \approx \sigma = \alpha \nabla u$ has been obtained from one of the three lowest-order FEMs as given in the previous section. Then we recover a weakly continuous piecewise linear polynomial $R(\sigma_h) \approx \sigma$ by interpolating the following nodal values at midpoints of edges of $\mathcal{T}_h$.

**Definition 4.1.** For an interior edge $e \subset \Omega$, the normal component of $R(\sigma_h)$ at the midpoint $m_e$ of $e$ is uniquely determined by
\[
R(\sigma_h)(m_e) \cdot n_e = \gamma_e^+ (\sigma_h|_{T_e^+} \cdot n_e) + \gamma_e^- (\sigma_h|_{T_e^-} \cdot n_e)
\]
and the tangential component of $R(\sigma_h)$ at the midpoint $m_e$ is double-valued by
\[
R(\sigma_h)|_{T_e^\pm}(m_e) \cdot t_e = \alpha_{T_e^\pm} \left\{ \gamma_e^- (\alpha^{-1} \sigma_h)|_{T_e^\pm}(m_e) \cdot t_e + \gamma_e^+ (\alpha^{-1} \sigma_h)|_{T_e^\pm}(m_e) \cdot t_e \right\},
\]
where the weight factors $\gamma_e^\pm$ are given by
\[
\gamma_e^+ = \frac{\alpha_{T_e^+}^{1/2}}{\alpha_{T_e^+}^{1/2} + \alpha_{T_e^-}^{1/2}}, \quad \gamma_e^- = 1 - \gamma_e^+ = \frac{\alpha_{T_e^-}^{1/2}}{\alpha_{T_e^+}^{1/2} + \alpha_{T_e^-}^{1/2}}.
\]

For a boundary edge $e \subset \partial \Omega$, we set
\[
R(\sigma_h)(m_e) \cdot n_e = \begin{cases} 
\sigma_h|_{T_e} \cdot n_e & \text{for } e \subset \Gamma_D, \\
g_N & \text{for } e \subset \Gamma_N,
\end{cases}
\]
and

\[ R(\sigma_h)(m_e) \cdot t_e = \begin{cases} \alpha T_e \frac{d u_D}{d s} & \text{for } e \subset \Gamma_D, \\ \sigma_h|_{T_e}(m_e) \cdot t_e & \text{for } e \subset \Gamma_N. \end{cases} \]

Finally, we find \( R(\sigma_h)|_T \in (P_1(T))^2 \) locally on each element \( T \in \mathcal{T}_h \) which interpolates the normal and tangential components defined above.

In other words, the normal component of \( R(\sigma_h) \) is recovered at the midpoint \( m_e \) of an interior edge \( e \) by averaging the discrete normal fluxes \( \sigma_h|_{T_e}|_e \cdot n_e \), whereas the tangential component of \( R(\sigma_h) \) is recovered at \( m_e \) by first averaging the discrete tangential derivatives \( (\alpha^{-1}\sigma_h)|_{T_e}|_e \cdot t_e \) at \( m_e \) and then multiplying the result by the coefficient \( \alpha \). This ensures that both the normal component of \( R(\sigma_h) \) and the tangential component of \( \alpha^{-1}R(\sigma_h) \) are continuous at the midpoints of edges of \( \mathcal{T}_h \). But, in general, it holds that \( R(\sigma_h) \notin H(\text{div}; \Omega) \) and \( \alpha^{-1}R(\sigma_h) \notin H(\text{curl}; \Omega) \).

**Remark 4.2.** For a smooth coefficient \( \alpha \), the weight factors \( \gamma^+_e \) are equal to \( \frac{1}{2} \) and we get a single formula

\[ R(\sigma_h)(m_e) = \frac{1}{2} \left\{ \sigma_h|_{T_e}(m_e) + \sigma_h|_{T_e}(m_e) \right\} \]

for every interior edge \( e \). This recovery formula (with some modifications for boundary edges) was discussed in [8, 9] for the \( P_1 \) conforming FEM, in [10] for the \( RT_0 \) mixed FEM, and recently in [11] for the \( P_1 \) nonconforming FEM. In particular, it was shown to be superconvergent if the triangulations are uniform and the exact solution \( u \) is smooth. Numerical results reported in the next section suggest that superconvergence of \( R(\sigma_h) \) may be retained even when \( \alpha \) is discontinuous.

**Remark 4.3.** The idea of averaging discrete normal fluxes on edges for flux recovery and averaging discrete tangential derivatives on edges for gradient recovery with the same weights (4.1) was also used by Cai and Zhang [4, 5], but their recovery procedure is different from ours. They considered discontinuities of discrete normal fluxes and discrete tangential derivatives on edges separately by recovering two vector fields

\[ \tilde{\sigma}_h \approx \sigma = \alpha \nabla u \in H(\text{div}; \Omega) \quad \text{and} \quad \tilde{\rho}_h \approx \nabla u \in H(\text{curl}; \Omega) \]

in conforming finite elements of \( H(\text{div}; \Omega) \) and \( H(\text{curl}; \Omega) \), respectively. Actually, one can set \( \tilde{\rho}_h = \nabla u^n_h \) for the \( P_1 \) conforming FEM and \( \tilde{\sigma}_h = \sigma^n_h \) for the \( RT_0 \) mixed FEM so that only one recovery is necessary for these two FEMs.

**Definition 4.4.** The error estimator based on the flux recovery \( R(\sigma_h) \) is defined as

\[ \eta_R = \| \alpha^{-1/2}(\sigma_h - R(\sigma_h)) \|_{0, \Omega}. \]

The following theorem reveals that the recovery-based error estimator \( \eta_R \) is in fact the edge residual error estimator (3.5) with slightly modified weight factors.
For every element $T \in T_h$, we have
\[
\|\alpha^{-1/2} (\sigma_h - R(\sigma_h))\|_{0,T} = \left( \frac{|T|}{3} \sum_{e \subset \partial T} \hat{\eta}_e^2 \right)^{1/2},
\]
where
\[
\hat{\eta}_e^2 = \begin{cases} 
\frac{1}{(\alpha_{T_e}^{1/2} + \alpha_{T_e}^{1/2})^2} \|\sigma_h \cdot n_e\|_e^2 + \frac{\alpha_{T_e}^{1/2} \alpha_{T_e}^{-1}}{(\alpha_{T_e}^{1/2} + \alpha_{T_e}^{1/2})^2} \|\alpha^{-1} \sigma_h (m_e) \cdot t_e\|_e^2 & \text{for } e \subset \Omega, \\
\alpha_{T_e}^{-1} |\sigma_h|_{T_e} (m_e) \cdot t_e - \frac{du_D}{ds} |^{2} & \text{for } e \subset \Gamma_D, \\
\alpha_{T_e}^{-1} |\sigma_h|_{T_e} \cdot n_e - gN |^{2} & \text{for } e \subset \Gamma_N.
\end{cases}
\]

Proof. Fix an edge $e \subset \partial T$ and set
\[
\sigma_h|_T (m_e) - R(\sigma_h)|_T (m_e) = A_e n_e + B_e t_e,
\]
where
\[
A_e = \sigma_h|_T \cdot n_e - R(\sigma_h)|_T (m_e) \cdot n_e, \\
B_e = \sigma_h|_T (m_e) \cdot t_e - R(\sigma_h)|_T (m_e) \cdot t_e.
\]
Since $\sigma_h - R(\sigma_h)$ is a linear polynomial on $T$, we obtain
\[
\|\alpha^{-1/2} (\sigma_h - R(\sigma_h))\|_{0,T}^2 = \frac{|T|}{3} \sum_{e \subset \partial T} \alpha_{T_e}^{-1} |\sigma_h|_{T_e} (m_e) - R(\sigma_h)|_{T_e} (m_e) |^2 \\
= \frac{|T|}{3} \sum_{e \subset \partial T} \alpha_{T_e}^{-1} (|A_e|^2 + |B_e|^2). 
\]
If $e$ is an interior edge and $T = T_e^+$, then it follows by Definition 4.1 that
\[
A_e = \gamma_e^- \|\sigma_h \cdot n_e\|_e, \\
B_e = \alpha_{T_e}^+ \gamma_e^+ \|\alpha^{-1} \sigma_h (m_e) \cdot t_e\|_e,
\]
which leads to
\[
\alpha_{T_e}^{-1} (|A_e|^2 + |B_e|^2) = \alpha_{T_e}^{-1} (\gamma_e^-)^2 \|\sigma_h \cdot n_e\|_e^2 + \alpha_{T_e}^+ (\gamma_e^+)^2 \|\alpha^{-1} \sigma_h (m_e) \cdot t_e\|_e^2 \\
= \frac{1}{(\alpha_{T_e}^{1/2} + \alpha_{T_e}^{1/2})^2} \|\sigma_h \cdot n_e\|_e^2 + \frac{\alpha_{T_e}^{1/2} \alpha_{T_e}^{-1}}{(\alpha_{T_e}^{1/2} + \alpha_{T_e}^{1/2})^2} \|\alpha^{-1} \sigma_h (m_e) \cdot t_e\|_e^2.
\]

We also arrive at the same result when $T = T_e^-$ by exchanging the roles of $T_e^+$ and $T_e^-$. Besides, Definition 4.1 immediately gives
\[
A_e = 0, \quad B_e = \sigma_h|_{T_e} (m_e) \cdot t_e - \alpha_{T_e} \frac{du_D}{ds} \quad \forall e \subset \Gamma_D
\]
and
\[
A_e = \sigma_h|_{T_e} \cdot n_e - gN, \quad B_e = 0 \quad \forall e \subset \Gamma_N.
\]
Collecting the above results, we conclude that for every edge $e \subset \partial T$,
\[
\alpha_T^{-1}(|A_e|^2 + |B_e|^2) = \tilde{\eta}_e^2,
\]
which completes the proof by (4.2).

It is trivial to see that $\frac{|\mathcal{T}|}{3} \tilde{\eta}_e^2$ is equivalent to $\tilde{\eta}_e^2$ independently of the mesh size $h$ and the jumps of $\alpha$ between subdomains. Therefore the recovery-based error estimator $\eta_R$ is robust with respect to the jumps of $\alpha$ between subdomains under the monotonicity condition stated in Section 3. By Theorem 4.5 it is also as inexpensive as the edge error residual estimators $\eta_E$ and $\tilde{\eta}_E$. Moreover, we expect that $\eta_R$ will yield more accurate estimates of the numerical error $\|\alpha^{-1/2}(\sigma - \sigma_h)\|_{0,\Omega}$ than $\eta_E$ and $\tilde{\eta}_E$, due to the superconvergence property of the recovered flux $R(\sigma_h)$ for smooth $\alpha$.

5. Numerical Results

In this section we report some numerical results which illustrate the performance of the recovery-based error estimator $\eta_R$ computed as in Theorem 4.5. The test problem is chosen to be the well-known benchmark problem with two intersecting interfaces and $f \equiv 0$ which was given by Kellogg [14]. Numerical experiments are carried out for the $P1$ conforming and nonconforming FEMs, because we have $\alpha \nabla_{h} u_{h}^{nc} = \sigma_{h}^{m}$ when $f \equiv 0$ and $\alpha$ is piecewise constant over $T_{h}$.

![Figure 1. Subdomains (left), initial mesh (middle), and adapted mesh with 2484 elements for $\beta = 0.1$ (right).](image)
Notice that $\alpha$ is highly discontinuous between subdomains when $\beta$ is close to 0 or 2. The exact solution has the form $u(r, \theta) = r^\beta \psi(\theta)$ in polar coordinates, where

$$\psi(\theta) = \begin{cases} 
\cos((\theta - \rho)\beta)/\cos(\rho\beta) & \text{if } 0 \leq \theta \leq \frac{\pi}{2}, \\
-\sin((\theta - 3\rho)\beta)/\sin(\rho\beta) & \text{if } \frac{\pi}{2} \leq \theta \leq \pi, \\
-\cos((\theta - 5\rho)\beta)/\cos(\rho\beta) & \text{if } \pi \leq \theta \leq \frac{3\pi}{2}, \\
\sin((\theta - 7\rho)\beta)/\sin(\rho\beta) & \text{if } \frac{3\pi}{2} \leq \theta \leq 2\pi.
\end{cases}$$

The Dirichlet condition determined by $u(r, \theta)$ is imposed on the whole boundary $\partial \Omega$. The solution $u$ and the normal flux $\sigma \cdot n$ are continuous across subdomain interfaces. Moreover, it holds that $u|_{\Omega_i} \in H^{1+\beta-\epsilon}(\Omega_i)$ for any $\epsilon > 0$ on each subdomain $\Omega_i$.

We consider two values $\beta = 1.9$ and $\beta = 0.1$, which correspond to $R = 0.006194 \cdots$ and $R = 161.4476 \cdots$, respectively. The coefficient $\alpha$ has the same jumps between subdomains, but the solution $u$ is regular for $\beta = 1.9$, while it is singular around the origin for $\beta = 0.1$. Hence we perform uniform mesh refinement for $\beta = 1.9$ and adaptive mesh refinement for $\beta = 0.1$, starting with the coarse mesh of 32 congruent elements shown in the middle of Figure 1.

For $\beta = 0.1$, the sequence of adapted meshes is generated by applying the red-green-blue refinement for the $P1$ nonconforming FEM, where we mark the element $T \in T_h$ for refinement if

$$\eta_T > 0.5 \max_{T' \in T_h} \eta_{T'}, \quad \eta_T = \|\alpha^{1/2}(\sigma_h - R(\sigma_h))\|_{0,T}.$$ 

Numerical results for the $P1$ conforming FEM are obtained on the same sequence of adapted meshes. In the right of Figure 1, we show the adapted mesh with 2484 elements. It is clearly seen that the mesh is highly refined around the singularity of $u$.

In Tables 1–2, we present numerical errors and effectivity indices with respect to the horizontal edge length $h_{el}$ for uniformly refined meshes and the number of elements $N_{el}$ for adaptively refined meshes. The effectivity index $EI$ is the ratio of the numerical error to the estimated error $\eta_R$. The numerical error is computed by the formula

$$\|\alpha^{1/2} \nabla(u - u_h^c)\|^2_{0, \Omega} = \int_{\partial \Omega} \alpha \nabla u \cdot n (u - 2u_h^c) \, ds + \|\alpha^{1/2} \nabla u_h^c\|^2_{0, \Omega}$$

for the $P1$ conforming FEM and

$$\|\alpha^{1/2} \nabla_h(u - u_h^{nc})\|^2_{0, \Omega} = \int_{\partial \Omega} \alpha \nabla(u - 2u_h^{nc}) \cdot n u \, ds + \|\alpha^{1/2} \nabla_h u_h^{nc}\|^2_{0, \Omega}$$

for the $P1$ nonconforming FEM, where a high-order quadrature is used to approximate the boundary integral over each edge $e \subset \partial \Omega$. These error formulas are readily derived using integration by parts and the equality $\nabla \cdot (\alpha \nabla u) = \nabla \cdot (\alpha \nabla_h u_h^{nc}) = 0$.

From Table 1 we see that the recovery-based error estimator $\eta_R$ is asymptotically exact, even though the coefficient $\alpha$ has large jumps between subdomains. (Recall that the triangulations are uniform and the solution $u$ is regular for $\beta = 1.9$.) On the other hand, Table 2 shows that $\eta_R$ produce rather accurate estimates of the numerical error even when the solution $u$ is
TABLE 1. Numerical errors and effectivity indices for $\beta = 1.9$ on uniformly refined meshes, where $h_{el}$ is the horizontal edge length.

<table>
<thead>
<tr>
<th>$1/h_{el}$</th>
<th>$P1$ conforming FEM</th>
<th>$P1$ nonconforming FEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>$8.916063e+1$</td>
<td>$8.563063e+1$</td>
</tr>
<tr>
<td>8</td>
<td>$4.460283e+1$</td>
<td>$4.404250e+1$</td>
</tr>
<tr>
<td>16</td>
<td>$2.230530e+1$</td>
<td>$2.222061e+1$</td>
</tr>
<tr>
<td>32</td>
<td>$1.115329e+1$</td>
<td>$1.114087e+1$</td>
</tr>
<tr>
<td>64</td>
<td>$5.576746e+1$</td>
<td>$5.574964e+1$</td>
</tr>
<tr>
<td>128</td>
<td>$2.788389e+1$</td>
<td>$2.788137e+1$</td>
</tr>
<tr>
<td>256</td>
<td>$1.394197e+1$</td>
<td>$1.394162e+1$</td>
</tr>
<tr>
<td>512</td>
<td>$6.970988e+1$</td>
<td>$6.970949e+1$</td>
</tr>
</tbody>
</table>

TABLE 2. Numerical errors and effectivity indices for $\beta = 0.1$ on adaptively refined meshes, where $N_{el}$ is the number of elements.

<table>
<thead>
<tr>
<th>$N_{el}$</th>
<th>$P1$ conforming FEM</th>
<th>$P1$ nonconforming FEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>$1.306996e+1$</td>
<td>$6.287810e+0$</td>
</tr>
<tr>
<td>512</td>
<td>$3.465470e+0$</td>
<td>$3.169759e+0$</td>
</tr>
<tr>
<td>1088</td>
<td>$1.929723e+0$</td>
<td>$1.800292e+0$</td>
</tr>
<tr>
<td>2484</td>
<td>$1.204094e+0$</td>
<td>$1.097007e+0$</td>
</tr>
<tr>
<td>6260</td>
<td>$7.355309e+0$</td>
<td>$6.627787e+0$</td>
</tr>
<tr>
<td>16484</td>
<td>$4.456168e+0$</td>
<td>$4.069451e+0$</td>
</tr>
<tr>
<td>45792</td>
<td>$2.638758e+0$</td>
<td>$2.470452e+0$</td>
</tr>
<tr>
<td>129060</td>
<td>$1.558867e+0$</td>
<td>$1.483348e+0$</td>
</tr>
<tr>
<td>339124</td>
<td>$9.543845e+0$</td>
<td>$9.214557e+0$</td>
</tr>
</tbody>
</table>

very singular and highly non-uniform meshes are used. These numerical results illustrate the robustness of $\eta_R$ and its superiority over the edge residual error estimators $\eta_E$ and $\tilde{\eta}_E$. 
ACKNOWLEDGMENT

This study was supported by 2014 Research Grant from Kangwon National University(No. C1011748-01-01).

REFERENCES

APPROXIMATION ORDER OF $C^3$ QUARTIC B-SPLINE APPROXIMATION OF CIRCULAR ARC

SUNG CHUL BAE and YOUNG JOON AHN

1DEPARTMENT OF MATHEMATICS EDUCATION, KOREA UNIVERSITY, SEOUL, 136-701, KOREA
2DEPARTMENT OF MATHEMATICS EDUCATION, CHOSUN UNIVERSITY, GWANGJU, 501–759, KOREA
E-mail address: ahn@chosun.ac.kr

ABSTRACT. In this paper, we present a $C^3$ quartic B-spline approximation of circular arcs. The Hausdorff distance between the $C^3$ quartic B-spline curve and the circular arc is obtained in closed form. Using this error analysis, we show that the approximation order of our approximation method is six. For a given circular arc and error tolerance we find the $C^3$ quartic B-spline curve having the minimum number of control points within the tolerance. The algorithm yielding the $C^3$ quartic B-spline approximation of a circular arc is also presented.

1. INTRODUCTION

The circle approximation is one of the most simple and challenging problems in the field of CAGD(Computer Aided Geometric Design). If a circular arc is subdivided with the same length, then all subdivided arcs are congruent. Therefore, if one arc is approximated by a Bézier curve, then all arcs can be approximated by the same method. This is the reason why the circle approximation is simple. However, a reduction in the error and an increase in the continuity of the approximation curve remain as problems to be solved. In the last thirty years, the focus of the circle approximation has been to find the spline approximation that has the highest possible orders of approximation and continuity.

Since de Boor [1] showed the existence of the $G^2$ cubic spline approximation of a planar curve with approximation order six, many studies have been carried out on the circle approximation using a Bézier or spline curve with a higher order of approximation or continuity. The methods for the circle or ellipse approximation by a $G^1$ quadratic [2] or $G^k$ cubic spline curve [3, 4, 5] for $k = 1, 2$ have been presented with increasingly smaller error. Floater [6, 7] found a $G^2$ quadratic spline approximation of a conic with approximation order four and a $G^{n-1}$ spline approximation of odd degree $n$ of a conic with approximation order $2n$, which can be...
naturally applied to the circle approximation. The approximation methods of a circular arc by a quartic or quintic spline with approximation order eight or ten and with $G^k$-continuity for $k = 1, 2, 3$ have been developed [8, 9, 10]. The error bound of a quartic Bézier approximation of a circular arc has been further reduced [11, 12, 13, 14, 15]. Moreover, the circle approximation using LN(linear normal) Bézier curves has been used to obtain the offset approximation [16, 17, 18, 19, 20].

The spline approximation of a circular arc is obtained by merging the Bézier approximation of a segment of the circular arc and its rotations. If the approximate spline is $G^k$-continuous at the junction point of two consecutive Bézier segments, the number of control points of the spline can be reduced by $k$. Thus, the continuity order of the approximate spline curve is an important factor. The geometric continuity $G^k$ cannot imply the continuity $C^k$ for $k \geq 2$ in general. Yang and Ye [21] presented a $C^2$ cubic spline approximation of a circular arc, but there is no $C^3$ spline approximation whose Bézier segments are all congruent in previous works on circle approximations. This is the motivation of our work. Since any $C^3$ spline curve composed of two or more Bézier segments should have a degree of at least four, we present a $C^3$ quartic B-spline approximation of a circular arc in this paper. The exact form of the Hausdorff distance between the circular arc and the $C^3$ quartic B-spline approximation is obtained. Using this Hausdorff distance, we present an algorithm yielding the $C^3$ quartic B-spline approximation with the smallest number of control points and an error less than a given tolerance.

The $C^1$ quadratic spline approximation of a circular arc can be easily obtained, and it has approximation order four [2]. Yang and Ye [21] showed that the $C^2$ cubic B-spline approximation of a circular arc has approximation order four as well. Thus, we are interested in the approximation order of our $C^3$ quartic B-spline approximation of a circular arc. In this paper, we show that its approximation order is six, which is a very interesting result.

Our manuscript is organized as follows. In section 2, we find the $C^3$ quartic uniform B-spline curve approximation of a circular arc, which is obtained by merging the quartic Bézier approximation and its rotations. In section 3, we present the Hausdorff distance between the $C^3$ quartic B-spline approximation and the circular arc in closed form, and prove that it has the approximation order six. In section 4, the algorithm yielding the $C^3$ quartic B-spline approximation with the smallest number of control points within the tolerance is obtained. We summarize our work in section 5.

2. $C^3$ QUARTIC UNIFORM B-SPLINE CURVE APPROXIMATION OF CIRCULAR ARCS

In this section, we find the approximation of a circular arc by a $C^3$ quartic uniform B-spline curve whose Bézier segments are all congruent.

Let $c$ be the unit circular arc of angle $0 < \alpha < \pi$ expressed by

$$c(\theta) = (\cos \theta, \sin \theta) \quad \text{for } \theta \in [0, \alpha]$$
and \( \mathbf{p} \) be the quartic Bézier approximation of the circular arc with the control points \( \mathbf{p}_0, \mathbf{p}_1, \cdots, \mathbf{p}_4 \) expressed by

\[
\mathbf{p}(t) = \sum_{i=0}^{4} \mathbf{p}_i B_i^4(t),
\]

where \( B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i} \) for \( i = 0, 1, \cdots, n \) is the Bernstein polynomial of degree \( n \) [22, 23]. Since the circular arc is symmetric, the quartic Bézier curve is restricted to be symmetric. Let \( b \) be the spline curve constructed by \( \mathbf{p} \) and its rotations \( R \mathbf{p}, R^2 \mathbf{p}, \cdots \), where \( R \) is the rotation operator by the angle \( \alpha \), and \( R_i \mathbf{p} \) is the rotated curve of \( \mathbf{p} \) by the angle \( i\alpha \).

Then, the \( C^3 \)-continuity of \( b \) at least implies that \( \mathbf{p} \) is the \( G^1 \) endpoint interpolation of the circular arc. Thus, the control points of \( \mathbf{p} \) are

\[
\begin{align*}
\mathbf{p}_0 &= (1, 0) \\
\mathbf{p}_1 &= (1, h) \\
\mathbf{p}_2 &= r(\cos \frac{\alpha}{2}, \sin \frac{\alpha}{2}) \\
\mathbf{p}_3 &= (\cos \alpha, \sin \alpha) + h(\sin \alpha, -\cos \alpha) \\
\mathbf{p}_4 &= (\cos \alpha, \sin \alpha).
\end{align*}
\] (2.1)

For a given circular arc \( c \) of angle \( \phi \) and a given positive integer \( m \), we will construct the quartic B-spline approximation \( b \) by the composition of the quartic Bézier curves \( \mathbf{p}, R \mathbf{p}, R^2 \mathbf{p}, \cdots, R^{m-1} \mathbf{p} \), where \( \alpha = \phi/m \). Consider the continuity of \( b \) at each junction point. Two quartic Bézier curves \( \mathbf{p} \) and \( R \mathbf{p} \) meet at \( \mathbf{p}_4 = R \mathbf{p}_0 \) and are symmetric with respect to the line \( L \) passing through the origin and \( \mathbf{p}_4 \), as shown in Fig. 1. The curve \( b \) is \( C^2 \)-continuous at \( \mathbf{p}_4 \) if and only if \( \mathbf{p}''(1) = (R \mathbf{p})''(0) \) or

\[
\Delta \mathbf{p}_3 - \Delta \mathbf{p}_2 = R(\Delta \mathbf{p}_2 - \Delta \mathbf{p}_1),
\]

where \( \Delta \mathbf{p}_i = \mathbf{p}_{i+1} - \mathbf{p}_i \). Its geometric meaning is that the ratio of the distances from \( \mathbf{p}_3 \) and \( \mathbf{p}_2 \) to the line \( L \) is 1 : 2, as shown in Fig. 1, which is equivalent to

\[
r \sin \frac{\alpha}{2} = 2h.
\] (2.2)

The curve \( b \) is \( C^3 \)-continuous at \( \mathbf{p}_4 \) if and only if \( \mathbf{p}'''(1) = (R \mathbf{p})'''(0) \) or

\[
\Delta \mathbf{p}_3 - 2\Delta \mathbf{p}_2 + \Delta \mathbf{p}_1 = R(\Delta \mathbf{p}_2 - 2\Delta \mathbf{p}_1 + \Delta \mathbf{p}_0).
\]

Its geometric meaning is that the perpendicular foot of \( \mathbf{p}_2 \) to the line \( L \) is the internally dividing point of the two perpendicular feet of \( \mathbf{p}_1 \) and \( \mathbf{p}_3 \) to the line \( L \) in the proportion of 1 : 2, as shown in Fig. 1, which is equivalent to

\[
\cos \alpha + h \sin \alpha = 3r \cos \frac{\alpha}{2} - 2.
\] (2.3)

Solving the linear system of two equations in Eqs. (2.2)–(2.3), we have

\[
r = \frac{2 + \cos \alpha}{\cos \frac{\alpha}{2}(2 + \cos^2 \frac{\alpha}{2})} \quad \text{and} \quad h = \frac{\tan \frac{\alpha}{2}(2 + \cos \alpha)}{2(2 + \cos^2 \frac{\alpha}{2})},
\] (2.4)
as shown in Fig. 2 and the curve composed of $p$ and $Rp$ can be a $C^3$-continuous curve.

**Proposition 2.1.** If $p(t)$, $t \in [0, 1]$ is the quartic Bézier curve with the control points $p_0, p_1, \cdots, p_4$ satisfying Eq. (2.4), then for any positive integer $m$, the curve $b(t)$, $0 \leq t \leq m$, defined by

$$
b(t) = \begin{cases} 
p(t) & \text{for } t \in [0, 1] \\
R(t)\{p(t - \lfloor t \rfloor) & \text{for } t \in (1, m]
\end{cases}
$$

is $C^3$-continuous, where $\lfloor x \rfloor$ is the greatest integer less than $x$.

For $i = 0, 1, \cdots, m - 1$, the control points of the quartic Bézier curve $R^i p$ are $R^i p_j$ for $j = 0, 1, \cdots, 4$. Thus, the $C^3$ curve $b(t)$, $t \in [0, m]$ can be represented in the form of a quartic
B-spline curve

\[ b(t) = \sum_{i=0}^{4m} b_i N_i^4(t) \]

with the control points

\[ p_0, p_1, p_2, p_3, p_4, Rp_1, Rp_2, \ldots, R^{m-1}p_3, R^{m-1}p_4 \]

and the knot vector \( t = (t_i)_{i=0}^{4m+5} \) satisfying \( t_0 = 0 \),

\[ t_{4i+j} = i \quad \text{for } i = 0, 1, \ldots, m \text{ and } j = 1, 2, 3, 4, \]

and \( t_{4m+5} = m \), where the B-spline basis functions of degree \( j \) are defined by

\[ N_i^0(t) = \begin{cases} 1 & \text{if } t_i \leq t < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \]

for \( i = 0, 1, \ldots \) and

\[ N_i^j(t) = \frac{t - t_i}{t_{i+j} - t_i} N_i^{j-1}(t) + \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}} N_{i+1}^{j-1}(t) \]

for \( j = 1, 2, \cdots \) recursively [24, 25]. The domain interval \([0, m]\) and the knot vector \( t \) can be transformed into any interval and a new knot vector by a linear functional.

Since \( b \) is \( C^3 \)-continuous and has multiple knots in the domain interior, it can be knot-removable. Using the reverse process of knot insertion algorithm [26, 27, 28, 29], we can obtain the \( C^3 \) quartic uniform B-spline approximation

\[ b(t) = \sum_{i=0}^{m+3} b_i N_i^4(t) \]

**Figure 3.** The quartic Bezier approximation (magenta) \( p \) of a quarter circle (green) and its control polygon (blue) satisfying Eqs. (2.1) and (2.4).
For $\phi = 1.2\pi$ and $m = 4$, the de Boor points $b_0, b_1, b_2, \ldots, b_{m+3}$ (green) and the Bézier points $p_0, p_1, \ldots, p_{4m}$ (blue).

with the new control points

\[
\begin{align*}
    b_0 &= p_0 \\
    b_1 &= p_1 \\
    b_2 &= 2p_2 - p_1 \\
    b_i &= \frac{3}{2 + \cos \alpha} R^{i-2} p_2 \quad \text{for } i = 3, \ldots, m \\
    b_{m+1} &= R^{-1}(2p_2 - p_3) \\
    b_{m+2} &= R^{-1}p_3 \\
    b_{m+3} &= R^{-1}p_4
\end{align*}
\]

and the new knot vector $t = (t_i)_{i=0}^{m+8}$ satisfying $t_0 = \cdots = t_4 = 0,

\[
t_i = i - 4 \quad \text{for } i = 5, \cdots, m + 3,
\]

and $t_{m+4} = \cdots = t_{m+8} = m$.

3. Approximation Order of $C^3$ Quartic B-spline Approximation of Circular Arc

In this section, we find the Hausdorff distance $d_H(c, b)$ between the circular arc $c$ of angle $\alpha$ and the quartic Bézier approximation $b$ having the control points in Eqs. (2.1) and (2.4), which is the maximum of

\[
\psi(t) = ||p(t)|| - 1 \quad \text{for } t \in [0, 1]
\]

Using this error function, we obtain the following error analysis.
Figure 5. Error functions $\psi(t)$, $t \in [0, 1]$ for $\alpha$ from $\frac{3}{10} \pi$ (bottom) to $\frac{\pi}{2}$ (top).

Proposition 3.1. The Hausdorff distance $d_H(c, b)$ between the circular arc $c$ of angle $\alpha$ and the quartic Bézier approximation $b$ having the control points in Eqs. (2.1) and (2.4) is

$$d_H(c, b) = \frac{(5 - \cos \frac{\alpha}{2})(1 - \cos \frac{\alpha}{2})^3}{8(\cos \frac{\alpha}{2})(2 + \cos^2 \frac{\alpha}{2})},$$

(3.1)

and its approximation order is six.

Proof. Let $\psi_1(t) = \|p(t)\|^2 - 1$. Then, by Eqs. (2.1) and (2.4), we have

$$\psi_1(t) = \frac{4 \sin^6 \frac{\alpha}{2}}{\cos^2 \frac{\alpha}{2}(2 + \cos^2 \frac{\alpha}{2})^2}(t^2 - t)^2\{\sin^2 \frac{\alpha}{2}(t^2 - t)^2 - 2(t^2 - t) + 1\}$$

which is a polynomial of degree eight. Since $\psi_1(t) \geq 0$ for $t \in [0, 1]$, we obtain $\psi(t) = \sqrt{\psi_1(t)} + 1 - 1$. It follows from $\psi'(t) = \frac{1}{2\sqrt{\psi_1(t+1)}} \cdot \psi'_1(t)$ that $\psi(t)$ and $\psi_1(t)$ have the same critical points. Since

$$\psi'_1(t) = \frac{8 \sin^6 \frac{\alpha}{2}}{\cos^2 \frac{\alpha}{2}(2 + \cos^2 \frac{\alpha}{2})^2}(2t - 1)(t^2 - t)\{2 \sin^2 \frac{\alpha}{2}(t^2 - t)^2 - 3(t^2 - t) + 1\}$$

has seven zeros, $0, \frac{1}{2}, 1$, and

$$\frac{1}{2} \pm \frac{1}{2} \sqrt{1 + \frac{3 \pm \sqrt{9 - 8 \sin^2 \frac{\alpha}{2}}}{\sin^2 \frac{\alpha}{2}}}$$

and the zeros other than $\frac{1}{2}$ are not contained in the open interval $(0, 1)$, $\psi(t)$ has a unique maximum at $\frac{1}{2}$, as shown in Fig. 5, and

$$d_H(c, b) = \psi\left(\frac{1}{2}\right) = \frac{(5 - \cos \frac{\alpha}{2})(1 - \cos \frac{\alpha}{2})^3}{8(\cos \frac{\alpha}{2})(2 + \cos^2 \frac{\alpha}{2})}.$$
By series extension, we have

\[ d_H(c, b) = \frac{1}{3} \cdot \frac{1}{2^1} \alpha^6 + \mathcal{O}(\alpha^8), \]

which means that the approximation order is six.

The Hausdorff distance \( d_H(c, b) \) in Eq. (3.1) is dependent only on the angle \( \alpha \) of the circular arc, as shown in Fig. 6; therefore, we denote it by \( \varepsilon(\alpha) \).

4. APPROXIMATION ALGORITHM AND NUMERICAL EXAMPLE

For a given circular arc \( c \) of angle \( \phi \) and a given tolerance \( TOL \), if \( d_H(c, b) = \varepsilon(\alpha) \) is greater than \( TOL \), then the circular arc can be approximated by the quartic B-spline curve, which consists of at least two Bézier segments, i.e., \( m \geq 2 \). Using the error analysis in Eq. (3.1), we can find the smallest integer \( m \) satisfying

\[ \varepsilon \left( \frac{\phi}{m} \right) < TOL. \]

Letting \( \alpha = \phi/m \), the quartic uniform B-spline curve \( b \) satisfying Eq. (2.5) is a \( C^3 \) approximation of the circular arc \( c \) having an error less than \( TOL \). Now, we present an algorithm yielding the \( C^3 \) quartic uniform B-spline approximation \( b \) of the circular arc of the angle \( \phi \).
Figure 7. For a full circle (\( \phi = 2\pi \)) and \( TOL = 0.005 \), our algorithm yields \( m = 5 \) and the \( C^3 \) quartic B-spline approximation (magenta) \( b \) with the control polygon (green) \( b_0, b_1, \cdots, b_8 \) within the error tolerance.

having the minimum number of control points within the error tolerance \( TOL \) as follows.

\begin{algorithm}
- input: the angle \( \phi \) and tolerance \( TOL \)
- find the smallest integer \( m \) satisfying \( \epsilon(\frac{\phi}{m}) < TOL \)
- put \( \alpha = \frac{\phi}{m} \)
- find the control points \( b_0, b_1, \cdots, b_{m+3} \) by Eq. (2.5)
  and the knot vector \( t = (t_i)_{i=0}^{m+8} \) by Eq. (2.6)
- output: \( m, b_0, b_1, \cdots, b_{m+3}, \) and \( t = (t_i)_{i=0}^{m+8} \)
\end{algorithm}

For example, if a full circle and \( TOL = 0.005 \) are given, then this algorithm yields \( m = 5 \) and the control points \( b_0, b_1, \cdots, b_8 \) of the \( C^3 \) quartic uniform B-spline approximation \( b \), as shown in Fig. 7. The exact Hausdorff distance is \( d_H(c, b) = 0.0017 \). Using the algorithm, Table 1 summarizes the required minimum number of control points of the \( C^3 \) quartic uniform B-spline approximation of the full circle within the error tolerance for \( TOL = 10^{-1}, 10^{-2}, \cdots, 10^{-5} \).

5. Conclusion

In this paper, we presented the \( C^3 \) quartic uniform B-spline approximation of a circular arc. Since the order of continuity is three, it can reduce the number of control points of the quartic B-spline curve obtained by merging of the quartic Bézier approximate curve and its rotations, which is an advantage of our approximation method. Another advantage is that the Hausdorff distance between any circular arc and its \( C^3 \) quartic B-spline approximation is obtained in closed form. Using this closed form, we proved that our quartic B-spline approximation has...
TABLE 1. The required minimum number of control points of the $C^3$ quartic B-spline approximation of the full circle ($\phi = 2\pi$) within the given tolerance $TOL$.

<table>
<thead>
<tr>
<th>$TOL$</th>
<th>number of control points</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}$</td>
<td>7</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>8</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>10</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>12</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>16</td>
</tr>
</tbody>
</table>

the approximation order six. Moreover, we obtained the $C^3$ quartic B-spline approximation having the minimum number of control points within the error tolerance and the algorithm yielding the $C^3$ quartic B-spline approximation.

ACKNOWLEDGMENT

This study was supported by research funds from Chosun University, 2015, and by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2011-0010147).

REFERENCES


PATH AVERAGED OPTION VALUE CRITERIA FOR SELECTING BETTER OPTIONS

JUNSEOK KIM\textsuperscript{1}, MINHYUN YOO\textsuperscript{2}, HYEJU SON\textsuperscript{3}, SEUNGGYU LEE\textsuperscript{1}, MYEONG-HYEON KIM\textsuperscript{4}, YONGHO CHOI\textsuperscript{1}, DARAE JEONG\textsuperscript{1}, AND YOUNG ROCK KIM\textsuperscript{5}

\textsuperscript{1}Department of Mathematics, Korea University, Seoul 136-713, Korea
\textsuperscript{2}Department of Financial Engineering, Korea University, Seoul 136-701, Korea
\textsuperscript{3}Department of Economics, Sookmyung Women’s University, Seoul 140-742, Korea
\textsuperscript{4}Business School, Korea University, Seoul 136-701, Korea
\textsuperscript{5}Major in Mathematics Education, Hankuk University of Foreign Studies, Seoul 130-791, Korea

E-mail address: rocky777@hufs.ac.kr

ABSTRACT. In this paper, we propose an optimal choice scheme to determine the best option among comparable options whose current expectations are all the same under the condition that an investor has a confidence in the future value realization of underlying assets. For this purpose, we use a path-averaged option as our base instrument in which we calculate the time discounted value along the path and divide it by the number of time steps for a given expected path. First, we consider three European call options such as vanilla, cash-or-nothing, and asset-or-nothing as our comparable set of choice schemes. Next, we perform the experiments using historical data to prove the usefulness of our proposed scheme. The test suggests that the path-averaged option value is a good guideline to choose an optimal option.

1. INTRODUCTION

When it comes to investment, selecting appropriate financial products and adjusting the ratio of those selected items, i.e., maximizing expected return of the portfolio and minimizing the risk of the investment at the same time, should be one’s top priority. There have been many theories concerning standards for compromising the ‘ideal’ portfolio. ‘Portfolio Theory’, by Sharpe, is one of the competing theories. To be specific, Sharpe insisted that, to maximize the benefit, investors should find the set of mean and variance of portfolios and select the one that provides the greatest expected utility. Similarly, the idea of comparing the set of mean of different types of options is an underlying base in this paper. When we choose one option
under the condition that all the options’ current expectations are the same, we have to make a
decision as to the best for the investment. In this paper, we present a method of an appropriate
judgement for a better option choice.

Until now, there have been various attempts to evaluate option prices in financial markets.
Beginning in 1973, it was described that a mathematical framework for finding the fair price of
a European option by Black and Scholes [1, 2], several numerical methods have been presented
for the cases where analytic solutions are neither available nor easily computable. See more
details about numerical methods such as finite difference method (FDM) [3, 4, 5, 6, 7, 8, 9,
10, 11, 12, 13], finite element method [14, 15, 16], finite volume method [17, 18, 19], and
a fast Fourier transform [20, 21, 22, 23, 24]. For convenience, we use the closed-form of the
Black–Scholes equation in this work. Next, we will describe the proposed method for choosing
the better option in next section. In the proposed method, we compute various path-averaged
option values and then choose the option which gives the maximum value.

The rest of the paper is organized as follows. Section 2 describes the proposed method to
determine the best option among comparable options whose current expectations are all the
same. We present the numerical experiments in Section 3. Finally, conclusions are drawn in
Section 4.

2. PROPOSED METHOD

For simplicity of exposition, we consider European call options. Let $x$ denote the value
of underlying asset, $t$ be time, $\sigma$ be the volatility of return on the underlying asset, and $r$
be the risk-free interest rate. $T$ and $E$ represent the maturity and predetermined exercise price of
option, respectively. Let $C_1$, $C_2$, and $C_3$ be the prices of vanilla European, cash-or-nothing,
and asset-or-nothing call options, respectively. Equations (2.1)–(2.3) are the payoff functions
for those options. In Fig. 1(a), (b), and (c), the solid and dash lines are the payoff at maturity
and the value of options at time $t = 0$.

$$\Lambda_1 (x) = \max (x - E, 0), \quad (2.1)$$

$$\Lambda_2 (x) = \begin{cases} 0 & \text{if } x < E, \\ K & \text{otherwise}, \end{cases} \quad (2.2)$$

$$\Lambda_3 (x) = \begin{cases} 0 & \text{if } x < E, \\ x & \text{otherwise}. \end{cases} \quad (2.3)$$

Similarly, let $P_1$, $P_2$, and $P_3$ be the prices of vanilla European, cash-or-nothing, and asset-
or-nothing put options, respectively. Equations (2.4)–(2.6) are also the payoff functions for
European put options. In Fig. 2(a), (b), and (c), the solid and dash lines are the payoff at
Figure 1. Payoffs of (a) European vanilla, (b) cash-or-nothing, and (c) asset-or-nothing call options. The solid and dash lines are the payoff at maturity and the value of options at time $t = 0$.

$\Lambda_1(x) = \max (E - x, 0)$, 

(2.4) 

$\Lambda_2(x) = \begin{cases} 
K & \text{if } x < E, \\
0 & \text{otherwise}, 
\end{cases}$

(2.5) 

$\Lambda_3(x) = \begin{cases} 
x & \text{if } x < E, \\
0 & \text{otherwise}. 
\end{cases}$

(2.6) 

Figure 2. Payoffs of (a) European vanilla, (b) cash-or-nothing, and (c) asset-or-nothing put options. The solid and dash lines are the payoff at maturity and the value of options at time $t = 0$. 

$\Lambda_1(x) = \max (x - E, 0)$, 

(2.7) 

$\Lambda_2(x) = \begin{cases} 
K & \text{if } x < E, \\
0 & \text{otherwise}, 
\end{cases}$

(2.8) 

$\Lambda_3(x) = \begin{cases} 
x & \text{if } x < E, \\
0 & \text{otherwise}. 
\end{cases}$

(2.9)
The closed-form solutions \([25, 26]\) for the European call and put options with payoffs (2.1)–(2.6) are given as follows.

\[
C_1(x, t) = xN(d_1) - \text{e}^{-r(T-t)}N(d_2), \quad (2.7)
\]
\[
C_2(x, t) = K\text{e}^{-r(T-t)}N(d_2), \quad (2.8)
\]
\[
C_3(x, t) = \text{e}^{-r(T-t)}N(d_1), \quad (2.9)
\]
\[
P_1(x, t) = \text{e}^{-r(T-t)}N(-d_2) - xN(-d_1), \quad (2.10)
\]
\[
P_2(x, t) = K\text{e}^{-r(T-t)}N(-d_2), \quad (2.11)
\]
\[
P_3(x, t) = x\text{e}^{-r(T-t)}N(-d_1), \quad (2.12)
\]

where \(N(d) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d} e^{-0.5x^2} dx\) is the cumulative distribution function for the standard normal distribution,

\[
d_1 = \frac{\log(x/E) + (r + 0.5\sigma^2)(T - t)}{\sigma\sqrt{T-t}}, \quad d_2 = d_1 - \sigma\sqrt{T-t}.
\]

In the proposed method, we use the path-averaged option value (PAOV) which is the average of discounted value of option from each time steps to the present along the path as

\[
\text{PAOV} = \frac{1}{N} \sum_{i=1}^{N} V(x, t_i)\text{e}^{-r t}, \quad (2.13)
\]

where \(V(x, t)\) is the value of the option at time \(t\) and \(N\) is the number of days. Here, the time \(t_i\) represents a discrete observation time, with uniform time steps. Thus, \(t_i - t_{i-1} = h\). Since we assume constant slope on the value of the underlying asset, the PAOV is obtained by the average of discounted option price on day by day.

3. Numerical experiments

In the previous section, we have introduced the concept of PAOV, path averaged option value, to calculate expected value of the option in the aspect of future tendencies of underlying asset. Now we will actually apply this concept both theoretically and practically. In the subsection 3.1, we set certain linear direction of future tendencies of underlying asset. To be specific, we calculate PAOV of 6 options (Vanilla call/put, Cash-or-nothing call/put, and Asset-or-nothing call/put), and compare the theoretical value. In the subsection 3.2, on the other hand, we apply PAOV with real stock data, KOSPI200, in more practical way. We find specific periods of stock data movements that have increase and decrease tendency, and calculate PAOV of them. In this process, we first calculate PAOV with real data of the real date, and then we fit a linear line for each tendencies and calculated PAOV with those fitted data. In addition, since the majority of option market in the real world is consisted with vanilla option, we perform our test only with European vanilla call and put options.
3.1. **Stock path with the drift.** In this section, we present the numerical tests to compare the performance of options with respect to the direction of underlying asset. Equations (2.7)–(2.12) are used to value an European option on a non-dividend stock. The value of underlying asset at present value \( x(0) \) and the exercise price \( E \) are 100, respectively. The maturity is 1 year, and the number of the time steps is 365. The riskless interest rate \( r \) is 3%. We firstly compare PAOV of call options given the condition that the underlying asset has a certain tendency, and then we compare PAOV of put options. The procedure is explained in the next paragraph.

Table 1 represents the call option price at time \( t = 0 \) and a ratio of the price of the asset-or-nothing to the price of vanilla call option, which means the option buyer takes a long position with an amount of ratio of vanilla call and cash-or-nothing. Moreover, we have set the present value of cash-or-nothing same with that of vanilla call option by adjusting designated cash payoff \( K \). Calculating the ratio and adjusting the present value are to make sure that different kinds of options have same condition.

<table>
<thead>
<tr>
<th>Types</th>
<th>Vanilla call</th>
<th>Cash-or-nothing</th>
<th>Asset-or-nothing</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>13.2833084</td>
<td>13.2833084</td>
<td>58.1011880</td>
<td>4.374</td>
</tr>
</tbody>
</table>

Table 2 shows the path-averaged call option values with respect to the direction of underlying asset. The obtained ratio is reflected in the calculation of asset-or-nothing option values, i.e., we divide the value of asset-or-nothing by the ratio for the fair comparison. We test every cases with linear increase or decrease at certain percentage which makes linear direction of underlying asset in daily market.

<table>
<thead>
<tr>
<th>Types</th>
<th>-0.1%</th>
<th>-0.05%</th>
<th>0%</th>
<th>+0.05%</th>
<th>+0.1%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vanilla call</td>
<td>16.107322</td>
<td>22.950863</td>
<td>37.393584</td>
<td>65.434432</td>
<td>99.379309</td>
</tr>
<tr>
<td>Cash-or-nothing</td>
<td>22.685429</td>
<td>33.689913</td>
<td>58.743554</td>
<td>82.868066</td>
<td>93.373250</td>
</tr>
<tr>
<td>Asset-or-nothing</td>
<td>21.381259</td>
<td>31.622485</td>
<td>54.799640</td>
<td>80.333005</td>
<td>96.336642</td>
</tr>
</tbody>
</table>

From Table 2, we can observe the fact that PAOVs of cash-or-nothing and asset-or-nothing tend to move together. To be specific, the variation of PAOV between vanilla call and either cash-or-nothing or asset-or-nothing is bigger than that of between cash-or-nothing and asset-or-nothing. This result comes from the different payoff structures of each option. When we check the payoffs of Fig. 3, we can easily notice that both cash-or-nothing and asset-or-nothing have discontinuous payoff at the maturity. Consequently, the values of the option jump up at the time near maturity, resulting in higher PAOV of cash-or-nothing and asset-or-nothing. However, we can also observe that PAOV of vanilla call option is the largest when the underlying asset
The value of European call options with respect to the increasing path of underlying asset. (a) Vanilla European, (b) cash-or-nothing, and (c) asset-or-nothing options.

Increasing by 0.1%. Table 2 suggests that we should choose options which give the highest PAOV depending on the expected path direction.

Similarly, Table 3 shows the put option price at time $t = 0$ and a ratio of the price of the asset-or-nothing to the price of vanilla put option, which means the option buyer takes a long position with an amount of ratio of vanilla put and cash-or-nothing.

<table>
<thead>
<tr>
<th>Types</th>
<th>Vanilla put</th>
<th>Cash-or-nothing</th>
<th>Asset-or-nothing</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.3278618</td>
<td>10.3278618</td>
<td>38.9433654</td>
<td>3.771</td>
</tr>
</tbody>
</table>

Also, the present value of vanilla put option is set to be same with cash-or-nothing by adjusting the designated cash payoff $K$, and again, this is our intention to make same comparison condition between different option types. Table 4 shows the path-averaged put option values with respect to the direction of underlying asset.

<table>
<thead>
<tr>
<th>Types</th>
<th>-0.1%</th>
<th>-0.05%</th>
<th>0%</th>
<th>+0.05%</th>
<th>+0.1%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vanilla put</td>
<td>75.797600</td>
<td>47.969067</td>
<td>26.691586</td>
<td>17.136683</td>
<td>12.671502</td>
</tr>
<tr>
<td>Cash-or-nothing</td>
<td>60.863853</td>
<td>54.053611</td>
<td>38.344239</td>
<td>23.619190</td>
<td>17.117946</td>
</tr>
<tr>
<td>Asset-or-nothing</td>
<td>57.955672</td>
<td>56.570245</td>
<td>41.983720</td>
<td>25.571325</td>
<td>18.423489</td>
</tr>
</tbody>
</table>

From Table 4, we can observe the fact that PAOV of cash-or-nothing and asset-or-nothing tend to move together again, and vice versa. When we check the payoffs of Fig. 4, we can easily notice that both cash-or-nothing and asset-or-nothing have discontinuous payoff at the maturity. Consequently, the values of the option jumps up at the time near maturity, resulting in higher PAOV of cash-or-nothing and asset-or-nothing. However, we can also observe that
PAOV of vanilla put option is the largest when the underlying asset decreases by 0.1%. To sum up, all these differences are due to different structures of options, and since those structures are easily changed by many factors, we can only calculate and judge the options by PAOV.

![Table 5. PAOV of call options with respect to the direction with random perturbation of underlying asset.](image)

![Table 6. PAOV of put options with respect to the direction with random perturbation of underlying asset.](image)

**FIGURE 4.** The value of European put options with respect to the increasing path of underlying asset. (a) Vanilla European, (b) cash-or-nothing, and (c) asset-or-nothing options.

Next, we consider a PAOV of the path we tested with random perturbation for sensitivity analysis. We generate the path with normal distributed perturbation of mean $\mu = 0$ and standard deviation $\sigma = 0.1$.

The result is almost similar with the experiment without perturbations. See Tables 5 and 6.
3.2. **Stock path with real data.** In this subsection, we apply the PAOV with real stock data. Here the stock data, KOSPI200 index announced by Korea Securities and Future exchange (KRX), are used [27]. KOSPI200 has existed since 1964, which consists of 200 representative constituents. These constituents were chosen based on the largeness of capital stocks. We can easily understand it as a national index such as S&P 500 of U.S.A. Figure 5 shows the prices of KOSPI200 for 2 years, from August 5, 2013 to August 6, 2015.

![Figure 5](image)

**Figure 5.** The rough variation of KOSPI200 for 2 years, from August 5, 2013 to August 6, 2015.

As a first step, we intend to apply PAOV with the random movement of underlying asset. We select the data from March 2, 2015 to June 1, 2015, that shows random movements, and manipulate them for efficient implementation. Then we calculate PAOV of the vanilla call and put option for KOSPI200 index at different strike prices, using the pricing formula derived by Black and Scholes (see Eqs. (2.7) and (2.10)). A schematic for the evaluating procedure is shown in Figure 6. We perform the test in a similar way on previous test with parameters as follows.

![Figure 6](image)

**Figure 6.** Schematic for vanilla (a) call and (b) put options on KOSPI200 data.
The underlying asset is KOSPI200, the maturity is 0.25 with 63 operation days from March 2, 2015 to June 1, 2015. The riskless interest rate $r$ is 1.65%, and the volatility on return of underlying asset $\sigma$ is 0.3. The result is shown in Table 7. As we can infer from the different payoff structures of vanilla call and put options, PAOV of vanilla call option increases when strike price decreases. On the other hand, PAOV of vanilla put option increases as strike price increases.

**Table 7. PAOVs with respect to different strike price.**

<table>
<thead>
<tr>
<th>Strike price</th>
<th>245</th>
<th>247.5</th>
<th>250</th>
<th>252.5</th>
<th>255</th>
<th>257.5</th>
<th>260</th>
</tr>
</thead>
<tbody>
<tr>
<td>put</td>
<td>4.5226</td>
<td>5.1813</td>
<td>5.9165</td>
<td>6.7353</td>
<td>7.6457</td>
<td>8.6563</td>
<td>9.7755</td>
</tr>
</tbody>
</table>

Next, we consider the windows for specific dates in real data to reflect a sort of tendency. The criteria for choosing certain tendencies are based on the concept called moving average. Here, moving average is one of the most frequently used indicators of stock market, which reflects long and short term tendency of stock index movements. For instance, if someone tries to calculate 20 days of moving average on a specified date, he or she has to gather data of past 20 operation days and simply divide the sum of data by 20. In the same way, we can easily calculate moving average of a period with specific lengths. 20 and 60 days of moving averages are the typically used periods, which generally imply short and long term of the stock tendency, respectively.

Practically, if 20 days of moving average increases, overtaking 60 days of moving average, this implies that the current stock indexes are on the rise. Thus, this tendency is interpreted as a signal of purchasing stocks. With the decreasing tendency of 20 days of moving average, on the other hand, which passes down through 60 days of moving average, the shareholders are encouraged to sell the stocks. These concepts are known as golden cross and dead cross.

**Figure 7. Moving average of 20 and 60 days from August 5, 2013 to August 6, 2015.**
In the Fig. 7, we calculate both 20 days and 60 days of moving average of KOSPI200, from August 5, 2013 to August 5, 2015, with 492 days of operation. By calculating the proportion of 20 days of moving average divided by that of 60’s, we select 2 periods that represents decreasing (from December 2, 2013 to February 2, 2014) and increasing (from February 2, 2015 to April 30, 2015) tendency of KOSPI200, namely, dead cross and golden cross. These periods are shaded in Fig. 8 below. Figure 8 represents some of the dates which we focus on.

\[\text{FIGURE 8. Shaded areas which represent the decreasing (left side, the first period) and increasing (right side, the second period) tendency.}\]

Tables 8 and 9 show PAOV from the first period and the second period of KOSPI200 data, respectively.

\[\text{TABLE 8. PAOV in the first period (November 1, 2013 ~ January 29, 2014) from Fig 8.}\]

<table>
<thead>
<tr>
<th>Strike</th>
<th>245</th>
<th>247.5</th>
<th>250</th>
<th>252.5</th>
<th>255</th>
<th>257.5</th>
<th>260</th>
</tr>
</thead>
</table>

\[\text{TABLE 9. PAOV in the second period (February 2, 2015 ~ April 30, 2015) from Fig 8.}\]

<table>
<thead>
<tr>
<th>Strike</th>
<th>245</th>
<th>247.5</th>
<th>250</th>
<th>252.5</th>
<th>255</th>
<th>257.5</th>
<th>260</th>
</tr>
</thead>
</table>

First, we obtain the ratios of each strike on the basis of the present option price of median strike (252.5) which is meant to manage the share, making same present option price. Next, we obtain the PAOV and multiply them by the ratios.

Tables 10 and 11 show PAOV on linear tendency from data of the first and second period.
TABLE 10. PAOV on linear decreasing tendency from the first period (November 1, 2013 ~ January 29, 2014) from Fig 8.

<table>
<thead>
<tr>
<th>Strike</th>
<th>245</th>
<th>247.5</th>
<th>250</th>
<th>252.5</th>
<th>255</th>
<th>257.5</th>
<th>260</th>
</tr>
</thead>
</table>

TABLE 11. PAOV on linear increasing tendency from the second period (February 2, 2015 ~ April 30, 2015) from Fig 8.

<table>
<thead>
<tr>
<th>Strike</th>
<th>245</th>
<th>247.5</th>
<th>250</th>
<th>252.5</th>
<th>255</th>
<th>257.5</th>
<th>260</th>
</tr>
</thead>
</table>

In this part, we simply draw a linear line by connecting the first and last KOSPI200 indexes in each periods respectively, and again calculated PAOV of them. We conclude that the results in the Tables 10 and 11 show no significance here, since the variances of PAOV have not changed.

4. CONCLUSION

The simple and useful method was proposed for choosing a better option among options under the conditions that all the options’ current expectations are the same and one has a confidence in predicting future tendency of underlying assets. To choose a better option, we compute the PAOV. As test problems, we considered three European call and put options such as vanilla, cash-or-nothing, and asset-or-nothing since there exist closed-form solutions. The proposed method is general. Therefore it can be applied to other options. If there is no closed-form solution is available, then we can use numerical approximations such as finite difference method, finite element method, and Monte Carlo simulation. The test results suggested that the PAOV is a good guideline to choose a better option.

ACKNOWLEDGMENT

The corresponding author (Young Rock Kim) was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education(NRF-2015R1D1A1A01059643) and supported by the National Institute of Mathematics Sciences(NIMS).

REFERENCES


Instruction for Authors

Manuscript Preparation
Beginning January 1, 2008, Journal of the Korean Society for Industrial and Applied Mathematics (J.KSIAM) uses a new paper format. The templates for the new paper format (MS word template and LaTeX template) can be downloaded from the KSIAM website (URL: http://www.ksiam.org).
To keep the review time as short as possible, J.KSIAM requests all authors to submit their manuscripts online via the journal Online Submission System, except for very special circumstances that J.KSIAM recognizes. All manuscripts submitted to J.KSIAM should follow the instructions for authors and use the template files. Manuscripts should be written in English and typed on A4-sized papers. The first page of the manuscript must include (1) the title which should be short, descriptive and informative, (2) the name(s) and address(es) of the author(s) along with e-mail address(es), (3) the abstract which should summarize the manuscript and be at least one complete sentence and at most 300 words. The manuscript should also include as a footnote the 2000 Mathematics Subject Classification by the American Mathematical Society, followed by a list of key words and phrases.

Final Manuscript Submission
To facilitate prompt turnarounds of author proofs and early placements of the accepted manuscripts in the future issues, J.KSIAM requests all authors of the accepted manuscripts to submit the electronic text/graphics file(s). The final manuscript must be submitted in either the LaTeX format or the MS word format. We recommend that all authors use the template files which can be downloaded at http://www.ksiam.org/jksiam. In the case that the corresponding author does not submit all relevant electronic text/graphics, further producing process regarding his/her manuscript would be stopped. The final manuscripts for publication should be sent to an appropriate managing editor via the journal Online Submission System.
Please note that the submission of the final manuscript implies that the author(s) are agreeing to be bound by the KSIAM Provisions on Copyright. Author check list and Copyright transfer can be found during the submission process via homepage.

Offprints
J. KSIAM supplies a final version of published paper in PDF form to a corresponding author. Authors are allowed to reproduce their own papers from the PDF file. Reprints of published paper can be also supplied at the approximate cost of production on request. For details, please contact one of managing editors.

Peer Review
Suitability for publication in J.KSIAM is judged by the editorial policy of the Editorial Board. A submitted paper is allocated to one of the Managing Editors, who has full directorship for the reviewing process of the submission. The managing editor selects one of Associate Editors, and the selected Associate Editor chooses two qualified referees to review the paper. The Associate Editor him/herself may act as a referee. The managing Editor handles all correspondence with the author(s). The anonymity of the reviewers is always preserved. The reviewers should examine the paper and return it with their reports to the Associate Editor within 4 weeks from the date of the initial review request. The reviewers recommend acceptance, rejection or revision through a report. Based on the reports, the Associate Editor makes a recommendation to the managing Editor whether the paper should be accepted, rejected or needs to be returned to the author(s) for revision.
Papers needing revision will be returned to the author(s), and the author(s) must submit a revised manuscript to the online system within one month from the date of the revision request; otherwise it will be assumed that the paper has been withdrawn. The revision is assigned for review to the managing Editor who handled its initial submission. The managing Editor sends the revised manuscript to the original Associate Editor to check if the manuscript is revised as suggested by the reviewers in the previous review. The Associate Editor makes a recommendation for the revision to the managing Editor within 3 weeks from the date of the Associate Editor assignment.
Final decision by the managing Editor is usually made within 3 months from the time of initial submission. The length of time from initial submission to final decision may vary, depending on the time spent for review and revision. A letter announcing a publication date is sent to author(s) after a manuscript has been accepted by the managing Editor.

**Research and Publication Ethics**
Research published in Journal of KSIAM must have followed institutional, national and international guidelines. For the policies on the research and publication ethics that are not stated in these instructions, the Guidelines on Good Publication (http://www.publicationethics.org.uk/guidelines) can be applied.

**Publication Charge**
The publication fee is US $200, and extra charge could be necessary for color printings. The publication cost is subjected to change according to the society's financial situation.

**Forms of Publication**
- Original papers: this form of publication represents original research articles on research findings.
- Review article: this form do not cover original research but rather accumulate the results of many different articles on a particular topic
- Erratum/Revision/Retraction: these kinds of editorial notice may be published.

**Format of References**
The Vancouver style system is the preferred reference system for stipulations not otherwise described below: References should be listed at the end of the paper and conforming to current Journal style. Corresponding bracketed numbers are used to cite references in the text [1]. For multiple citations, separate reference numbers with commas [2, 3], or use a dash to show a range [4-7]. The DOI (Digital Object Identifier) should be incorporated in every reference for which it is available (see the last reference example).

- **Journal**

- **Book**

- **Proceedings**
Copyright Transfer Agreement

Copyright in the unpublished and original article, including the abstract, entitled -

________________________________________ (Title of Article)

Submitted by the following author(s)

________________________________________ (Names of Authors)

is hereby assigned and transferred to Korean Society for Industrial and Applied Mathematics(KSIAM) for the full term thereof throughout the world, subject to the term of this Agreement and to acceptance of the Article for publication in Journal of Korean Society for Industrial and Applied Mathematics.

KSIAM shall have the right to publish the Article in any medium or form, or by any means, now known or later developed.

KSIAM shall have the right to register copyright to the Article in its name as claimant whether separately or as part of the journal issue or other medium in which the Article is included. The Author(s) reserve all proprietary right other than copyright, such as patent rights. The Author(s) represent and warrant:

1. that the Article is original with them;
2. that the Article does not infringe any copyright or other rights in any other work, or violate any other rights;
3. that the Author(s) own the copyright in the Article or are authorized to transfer it;
4. that all copies of the Article the Author(s) make or authorize will include a proper notice of copyright in KSIAM’s name;

If each Author’s signature does not appear below, the signing Author(s) represent that they sign this Agreement as authorized agents for and on behalf of all the Authors, and that this Agreement and authorization is made on behalf of all the Authors.

________________________________________

Signature_________________________ Date __________________________
Author’s checklist

☐ This manuscript has never been submitted to or published in other journals.

☐ All citation references are correct and comply with the submission rule.

☐ I will supply a manuscript in either TeX or MS-Word format as well as all artworks once the article is accepted.

☐ I filled out and signed the proper copyright transfer agreement.

☐ Research published in the Journal has followed institutional, national, and international guidelines of ethnics.

☐ I checked whether the manuscript contained all authors’ names, affiliation, full address, e-mail. And also whether it contained an abstract and keywords.

Paper title__________________________________________________________

Authors____________________________________________________________

Signature__________________________ Date____________________________
<table>
<thead>
<tr>
<th>Title</th>
<th>Authors</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANALYSIS OF THE UPPER BOUND ON THE COMPLEXITY OF LLL ALGORITHM</td>
<td>YUNJU PARK AND JAEHYUN PARK</td>
<td>107</td>
</tr>
<tr>
<td>ARITHMETIC AVERAGE ASIAN OPTIONS WITH STOCHASTIC ELASTICITY OF VARIANCE</td>
<td>KYU-HWAN JANG  MIN-KU LEE</td>
<td>123</td>
</tr>
<tr>
<td>ROBUST A POSTERIORI ERROR ESTIMATOR FOR LOWEST-ORDER FINITE ELEMENT METHODS OF INTERFACE PROBLEMS</td>
<td>KWANG-YEON KIM</td>
<td>137</td>
</tr>
<tr>
<td>APPROXIMATION ORDER OF $C^3$ QUARTIC B-SPLINE APPROXIMATION OF CIRCULAR ARC</td>
<td>SUNG CHUL BAE AND YOUNG JOON AHN</td>
<td>151</td>
</tr>
<tr>
<td>PATH AVERAGED OPTION VALUE CRITERIA FOR SELECTING BETTER OPTIONS</td>
<td>JUNSEOK KIM, MINHYUN YOO, HYEJU SON, SEUNGGYU LEE, MYEONG-HYEON KIM, YONGHO CHOI, DARAE JEONG, AND YOUNG ROCK KIM</td>
<td>163</td>
</tr>
</tbody>
</table>