

การจัดตั้งความน่าจะเป็นเพื่อการขยายสหสัมพันธ์  
ในการตั้งราคา CDO



นางสาวจรรยา ศิริอารยะพันธ์

ศูนย์วิทยทรัพยากร

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาการเงิน ภาควิชาการเงินและการธนาคาร

คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2552

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

**PROBABILITY BUCKETING FOR CORRELATION EXPANSION  
IN CDO PRICING**

**MISS CHANYA SIRIARAYAPHAN**

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

**A Thesis Submitted in Partial Fulfillment of Requirements**

**for the Degree of Master of Science Program in Finance**

**Department of Banking and Finance**

**Faculty of Commerce and Accountancy**

**Chulalongkorn University**

**Academic Year 2009**

**Copyright of Chulalongkorn University**

**521896**

Thesis Title           **PROBABILITY BUCKETING IN CORRELATION  
EXPANSION FOR CDO PRICING**

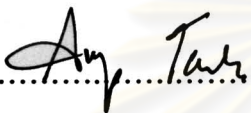
By                       **Miss Chanya Siriarayaphan**

Field of Study       **Finance**


Thesis Advisor       **Sira Suchintabandid, Ph.D.**


---

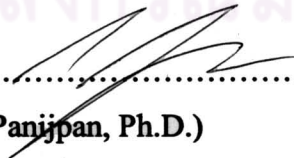
Accepted by the Faculty of Commerce and Accountancy, Chulalongkorn  
University in Partial Fulfillment of the Requirements for the Master's degree

  
..... Dean of the Faculty of  
Commerce and Accountancy  
(Associate Professor Annop Tanlamai, Ph.D.)

**THESIS COMMITTEE**

  
..... Chairman  
(Thaisiri Watewai, Ph.D.)

  
..... Thesis Advisor  
(Sira Suchintabandid, Ph.D.)

  
..... External Examiner  
(Kris Panijpan, Ph.D.)

จรรยา ศิริอารยะพันธ์ : การจัดตั้งความน่าจะเป็นเพื่อการขยายสหสัมพันธ์ในการตั้งราคา CDO.  
(PROBABILITY BUCKETING FOR CORRELATION EXPANSION IN CDO  
PRICING) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ดร.สิระ สุจินตะบัณฑิต, 36 หน้า.

งานของเราคือการรวมสองเทคนิคซึ่งก็คือเทคนิคการขยายสหสัมพันธ์และเทคนิคตั้งความน่าจะเป็น เทคนิคการขยายสหสัมพันธ์แสดงให้เห็นว่า CDO ที่ผู้ให้กู้ยืมแต่ละรายขึ้นแก่กัน สามารถที่จะแสดงในรูปแบบชุดของราคาและผู้ให้กู้ยืมแต่ละรายเป็นอิสระกัน ดังนั้นวิธีนี้จึงมีความซับซ้อนน้อยในการตั้งราคา CDO และตราสารอนุพันธ์ อีกเทคนิคหนึ่งคือเทคนิคตั้งความน่าจะเป็นซึ่งเป็นเทคนิคที่สามารถสร้างการแจกแจงความเสียหายได้ วัตถุประสงค์ของเราคือพัฒนาตัวแบบในการประมาณราคาชั้นของ CDO และตราสารอนุพันธ์ซึ่งมีความซับซ้อนน้อยและเพิ่มความเร็วในการคำนวณผล เรายังสนใจวิธีสร้างการแจกแจงความเสียหายซึ่งเป็นส่วนสำคัญส่วนหนึ่งในการตั้งราคาตราสารอนุพันธ์ เราพบว่าวิธีการที่รวมเทคนิคทั้งสองนี้เข้าด้วยกันให้ผลลัพธ์ที่ยอมรับได้

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา.....การเงินและการธนาคาร.....

ลายมือชื่อนิติศ.....จรรยา ศิริอารยะพันธ์.....

สาขาวิชา.....การเงิน.....

ลายมือชื่ออ.ที่ปรึกษาวิทยานิพนธ์หลัก.....*S. S.*.....

ปีการศึกษา.....2552.....

## 518 20721 26 : MAJOR FINANCE

KEYWORDS : Collateralize Debt Obligation/ Probability Bucketing/Correlation  
Expansion/Loss distribution/pricing.

CHANYA SIRIARAYAPHAN : PROBABILITY BUCKETING FOR  
CORRELATION EXPANSION IN CDO PRICING. THESIS ADVISOR : SIRA  
SUCHINTABANDID, Ph.D., 36 pp.

Our work is to combine two techniques which are Correlation Expansion technique and Probability Bucketing technique. Correlation Expansion technique shows that CDOs, whose obligors are dependent, can be expressed as a series of prices in independent obligor models. So, this is much less complicated to price CDOs and credit basket derivatives. Probability Bucketing technique is the technique that can create loss distributions. Our purpose is to develop a method of approximating CDOs tranche price and credit basket derivatives, which is less complicated in the computation of the model's output and improves the calculation speed. We also focus on the way to contribute loss distributions which is one of the important parts for pricing credit derivatives. We find out that this combined method gives us acceptable outputs.

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

Department : Banking and Finance

Student's Signature ชรรษา ศิริธารชฌ์

Field of Study : Finance

Advisor's Signature ศิริ

Academic Year : 2009

## ACKNOWLEDGEMENTS

This work was supported by Faculty of Commerce and Accountancy Chulalongkorn University. The author would like to thank Sira Suchintabandid (Ph.D.), who is my advisor, for his invaluable support, encouragement, supervision and helpful suggestions throughout this study. Sincerely thanks to Thaisiri Watwai (Ph.D.) and Kris Panijpan (Ph.D.) for their comments and suggestions. I also thank to my friends who shared their love and experiences with me. Moreover, I would like to deeply thank to my parents for their love and support. Without all these people's care and consideration, this paper would not be completed.



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## CONTENTS

	PAGE
Abstract in Thai.....	iv
Abstract in English.....	v
Acknowledgements.....	vi
Contents.....	vii
List of Tables .....	viii
List of Figures .....	ix
Chapter	
Chapter I : Introduction.....	1
Chapter II : Literature Review.....	5
Chapter III : Methodology.....	7
Example I.....	10
Example II.....	14
Example III.....	16
Chapter IV : Results.....	19
Chapter V : Conclusion.....	20
References.....	21
Appendices	
Appendix A : Models	
Correlation Expansion.....	25
Probability Bucketing.....	28
Appendix B : Program Code.....	32
Biography.....	36

**LIST OF TABLES**

	<b>PAGE</b>
Table 1.....	11
Table 2.....	11
Table 3.....	12
Table 4.....	14
Table 5.....	14
Table 6.....	16
Table 7.....	16
Table 8.....	17
Table 9.....	17



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย



**LIST OF FIGURES**

	<b>PAGE</b>
Figure 1 : Illustrate the Probability Bucketing Technique.....	3
Figure 2 : Cumulative distribution function for example I.....	13
Figure 3 : Cumulative distribution function for example II.....	15
Figure 4 : Cumulative distribution function for example III.....	18



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

# CHAPTER I

## INTRODUCTION

As the credit derivatives market had grown over the past decade, products that depended on default correlations had become more popular such as a collateralized debt obligation (CDOs) and other credit basket derivatives. There are several advantages of issuing credit risk securities to improve the liquidity or cash flow (Rajiv Bhatt, 2008), so we believe that the CDOs market and other credit basket derivatives will return after the CDO market was almost stopped since the end of 2007 because of the crisis. In this paper, we would like to create a better or more accurate pricing model for market participants when the CDOs market comebacks.

One of the major risks involved in the evaluation of credit derivatives is credit risk which is the distribution of financial loss caused by a broken financial agreement; for instance, a failure to pay interest or principal on a loan or bond. Approximating loss distribution that tells us the probabilities associated with different percentage losses in the portfolio is a part of the pricing process. The difficult task when we construct a loss distribution is the correlation among obligors since different correlation among obligors can make a loss distribution look completely different. In this paper, we presented the way to deal with this task through CDOs pricing framework.

Generally, a Monte Carlo Simulation, the market standard method, is applied to derive values for CDOs. However, a Monte Carlo Simulation is not very efficient for calculating price in real time because of the number of simulations that need to be run. For one sampling of  $Z$ , a set of risk factors, the computing time of Joint Gaussian probabilities requires  $2dM$  basic operations and  $M$  evaluation of the normal cdf where  $d$  refer to the number of risk factors and  $M$  is number of obligors. Since the simulation often requires many replications, the computing time will grow with the number of replication. This can take too much time for practitioners in times of strong market movements. For numerical methods, the more difficulty of pricing CDOs has arisen since we venture into the more realistic case of multifactor models, because traditional methods often require computing time that is exponential in the number of factors. (P.Glasserman and S.Suchintabandid, 2007).

We will focus on the problem of pricing CDOs and credit basket derivatives. Our purpose of this work is to develop a method of approximating CDOs tranche price that is less complicated in the computation of the model's output and improves the calculation speed. To determine joint default probabilities and develop a method of approximating CDOs tranche prices we combine two techniques:

1. The correlation expansion (P.Glasserman and S.Suchintabandit, 2007)
2. The probability bucketing approach (John Hull and Alan White, 2004)

For the first technique, the correlation expansion, is a method for approximating CDOs tranche prices by using a series of portfolios in which obligors are independent to approximate a portfolio whose obligors are correlated. The approximation is of the form:

$$E(L - y)^+ \approx \sum_J \text{const} \cdot \tilde{E}_J(L - y)^+$$

where, for each label  $J$ , the expectation  $\tilde{E}_J(L - y)^+$  is the tranche price of a credit portfolio whose obligors are independent.

The advantage of this method is

- Pricing in independent obligor models are easy to compute.

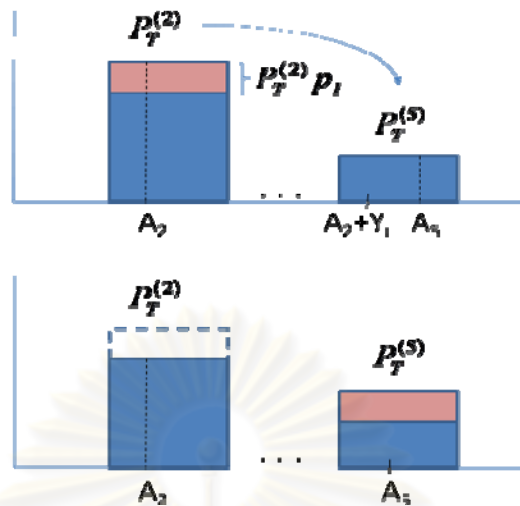
The disadvantage of this method is

- This technique requires the very exact value of  $\tilde{E}_J(L - y)^+$ .

The second technique is the probability bucketing approach. The purpose of this technique is to estimate the probability that the total loss lies in the  $k$ -th bucket for all  $k$  by time  $T$ , conditional on the values of the risk factors  $Z_1, Z_2, \dots, Z_d$ . Assume that the recovery rate is known and there are  $M$  obligors, then:

1. Divide potential losses into ranges:  $\{0, b_0\}, \{b_0, b_1\}, \dots, \{b_{K-1}, \infty\}$
2. Calculate the conditional probability that the loss by time  $T$  will be in the  $k$ -th bucket ( $P_T^{(k)}$ ) and the mean loss conditional that the loss is in  $k$ -th bucket ( $A_k$ ) by first assume that there are no obligors.
3. Built up one obligor at a time.

The only assumption in the iterative procedure is that we concentrate at the recent value of  $A_k$  for all the probability associated with bucket  $k$ .



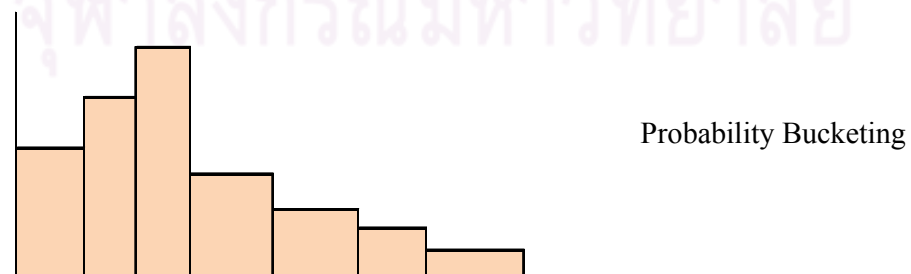
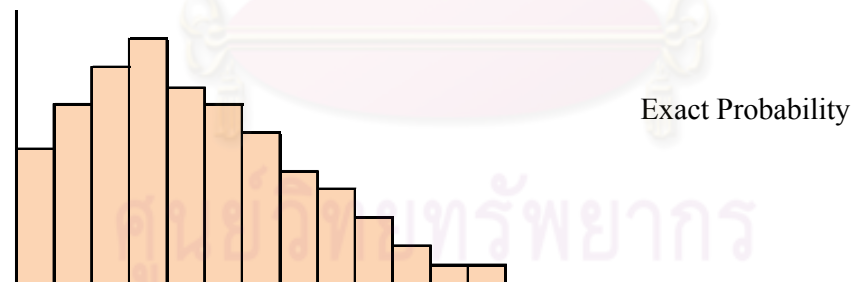
**Figure 1:** Illustrate the Probability Bucketing Technique

The advantages of this method are

- It gives us the advantage in term of computation speed.
- It gives reasonably accurate answers compared with Monte Carlo simulation.
- It is flexible to organize sizes of buckets.

The disadvantage of this method is

- It cannot deal with the correlation task.



Andersen, L., J. Sidenius, and S. Basu (2003) also show a similar technique. The loss unit was chosen by a ratio of losses which is a special case of probability bucketing approach.

So, we use probability bucketing technique to create independent total loss distributions and merge them with correlation expansion technique to help them create a dependent total loss distribution.

From the two techniques above, we find that using probability bucketing approach to estimate  $\tilde{E}_j(L-y)^+$  and substituting it into

$$E(L-y)^+ \approx \sum_j \text{const} \cdot \tilde{E}_j(L-y)^+$$

still give us the accurate value of  $E(L-y)^+$  in condition of the size of the buckets.



## **CHAPTER II**

### **LITERATURE REVIEW**

Currently, several numerical integrations were presented to deal with the difficulty of correlation among obligors in basket credit derivatives and CDOs pricing. Andersen, Sidenius, and Basu (2003) and Laurent and Gregory (2003) presented the method to deal with dependent defaults for multi-name credit derivatives and found that using semi-analytical methods and numerical integration could efficiently calculate CDO prices which were a single-factor model.

P.Glasserman and S.Suchintabandit (2007) developed numerical methods which were non-simulation based, and were less sensitive to the number of factors for evaluating credit risks and pricing basket credit derivatives and CDOs. They presented two methods which were Correlation Expansion and the method that based on the Laplace inversion method. For Correlation Expansion technique showed that CDO tranche prices could be expressed as a series of prices in independent obligor models and could be used to approximate joint probabilities of multivariate normal random variables. In another method, it presented fast and precise way of valuing CDO tranche especially with the high number of risk factors.

To predict the loss distribution for basket credit derivatives and CDOs is one of the important parts of the pricing process. Oldrich Alfons Vasicek (2002) offered a method to construct loss distributions that could be used to represent the loan loss behavior of large portfolios.

John Hull and Alan White (2004) offered a numerical procedure to create the loss distribution called "Probability Bucketing". Andersen, L., J. Sidenius, and S. Basu (2003) also showed the similar technique. The loss unit was chosen by a ratio of losses which was a special case of probability bucketing approach.

Other researchers for example Sudheer Chava, Catalina Stefanescu and Stuart Turnbull (2008) also focused on modeling and calculating the loss distribution. And found that the default probabilities and recovery rate had negative correlation. In addition, they found that the predicted loss distribution was affected by the default model significantly.

There also are a number of researchers who presented numerical methods to price credit basket derivative products. From literature review above, each method has both advantages and disadvantages. For our work, we decided to combine two techniques, Correlation Expansion and Probability Bucketing, to improve the method which can give us the advantage in term of speed and accuracy also in the less complicated way.



ศูนย์วิจัยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## CHAPTER III

### METHODOLOGY

The main contribution of this paper is to combine two techniques. First, the Correlation Expansion method which is a method for approximating CDOs tranche prices by using a series of portfolios in which obligors are independent to approximate a portfolio whose obligors are correlated. The approximation is of the form:

$$E(L - y)^+ \approx \sum_j \text{const} \tilde{E}_j(L - y)^+$$

The advantage of this method is pricing in independent obligor models is easy to compute. The second method is the Probability Bucketing method which approximates the probability distribution of the losses by time T. The loss distribution is built up one obligor at a time. This approach has an advantage in term of computation speed and always gives reasonably accurate answers compared with Monte Carlo simulation (John Hull and Alan White, 2004). The details of each method are explained in appendix A.

The key that makes the program (MATLAB) generate faster is the technique that we created sparse matrix of probabilities and updated probabilities in every bucket at the same time for every one obligor that is added in, instead of using loops to update probabilities which can update only one bucket a loop. Suppose we have N buckets and M obligors. If we do update the probability one time for one bucket, then for one obligor that is added in we have to calculate it N times and for M obligors we have to do it N×M times. This is very time consuming. So, we update probabilities in the form of matrix. This helps us to be able to update N buckets at one time. We also developed the program code to be more flexible in case that there are many probabilities that move to the same bucket. We can illustrate the matrix approach in Probability Bucketing as follows



**Buckets after add the  $i$ -th obligor**

		Bin 1	Bin 2	Bin 3	...	Bin N
Previous buckets before add the $i$ -th obligor	Bin 1	$P_1 * p(i)$	0	0		0
	Bin 2	0	0	$P_2 * p(i)$		0
	Bin 3	0	0	$P_3 * p(i)$		0
	⋮					
	Bin m-1	0	0	0		$P_{m-1} * p(i)$
	Bin m	0	0	0		$P_m * p(i)$
	<b>Sum</b>	$P_1 * p(i)$	0	$P_2 * p(i) + P_3 * p(i)$		$P_{m-1} * p(i) + P_m * p(i)$

	Bin 1	Bin 2	Bin 3	...	Bin N	
Bin 1	$P_1 * p(i) * (A_1 + Y_i)$	0	0		0	
Bin 2	0	0	$P_2 * p(i) * (A_3 + Y_i)$		0	
Bin 3	0	0	$P_3 * p(i) * (A_3 + Y_i)$		0	
⋮						
Bin m-1	0	0	0		$P_{m-1} * p(i) * (A_{m-1} + Y_i)$	
Bin m	0	0	0		$P_m * p(i) * (A_m + Y_i)$	
	<b>Sum</b>	$P_1 * p(i) * (A_1 + Y_i)$	0	$P_2 * p(i) * (A_3 + Y_i) + P_3 * p(i) * (A_3 + Y_i)$		$P_{m-1} * p(i) * (A_{m-1} + Y_i) + P_m * p(i) * (A_m + Y_i)$

The two figures above represent the sparse matrix of probabilities in each bucket after we added the  $i$ -th obligor. Next, sum them in column and find that which buckets are not zero. Then, we can update the mean loss and the probabilities following the formulas in appendix A by using the two matrixes above.

To combine these two methods, first, we need to create perturbed probabilities ( $\tilde{p}_i^{(j)}$ ). Then, substitute the perturbed probabilities  $\tilde{p}_i^{(j)}$  to contribute the loss distribution by using probability bucketing method. The loss distribution tells us about the probabilities associated with different percentage losses in a portfolio. Consequently, we easily obtain expectations  $\tilde{E}_j(L - y)^+$  whose obligors are independent. But to make it more realistic, we need to consider the correlation among obligors. Thus, we use the correlation expansion method to deal with this task. This method helps us obtain the expectation  $E_i(L - y)^+$  whose obligors are dependent from expectations  $\tilde{E}_j(L - y)^+$  whose obligors are independent.

We can summarize this through these steps:

1. Calculate perturbed probabilities ( $\tilde{p}_i^{(j)}$ ).
2. Create independent loss distributions of perturbed probabilities using probability bucketing technique.
3. Compute expectations  $\tilde{E}_j(L-y)^+$  whose obligors are independent.
4. Weight  $\tilde{E}_j(L-y)^+$  using correlation expansion technique. Then, we receive the expectation  $E_i(L-y)^+$  whose obligors are dependent.

To illustrate the computing time, P.Glasserman and S.Suchintabandid (2007) compare the computing time of Monte Carlo Simulation and Correlation Expansion method in the following table

	Plain Monte-Carlo With $N$ replications	Correlation Expansion
Number of operations		
Basic operations*	$2dMN$	$4n \binom{n+2d}{n} M$
Normal cdf $\Phi(\cdot)$	$MN$	-
Accuracy	$O\left(\frac{1}{\sqrt{N}}\right)$	$O(\rho^{n+1})$

\*Basic operations comprise addition and multiplication.

Where  $d$  is refer to the number of risk factors.  $M$  is number of obligors and  $n$  is the number of dimensions. This approximate time of Correlation Expansion does not include the evaluated time of  $\varphi(v_1), \dots, \varphi(v_M)$  in perturbation method (only additions and multiplications).

However, since we combine two techniques which are Correlation Expansion and Probability Bucketing, the approximation of computing time is changed. The advantage of Probability Bucketing helps to reduce computing time of Correlation Expansion around 1.5 times of bucket reducing times. For example, suppose that we have  $M$  obligors and 1,000 buckets. This takes time  $T$  with Correlation Expansion to calculate. If we rearrange the buckets to be 250 buckets, the approximate time of the combined method is  $T / (1.5*4)$ .

So, this combined method lets us obtain an acceptable value in such a short time compared to the traditional methods such as Monte Carlo simulation which is not very efficient for calculating price in real time because of the number of simulations that is needed to be run. We will show how this combined method works through three examples below.

For the first example, we show a case that the potential losses are small and a default probability of each obligor is equally low.

For the second example, we show a case that the potential losses are big and a default probability of each obligor is equally low.

And for the last example, we illustrate a case that a default probability of each obligor is different and high.

### **Example I**

Suppose we have a portfolio of 50 obligors. Each obligor has a default probability equal to 0.02. The given loss amount of obligor  $k$  is  $C_k = k$ . The perturbation parameter  $\theta$  that we use is 0.1. And loading matrix  $A$  is a sparse matrix size  $50 \times 5$ . For the first column, the elements from  $a(1,1)$  to  $a(12,1)$  are equal to 0.2. The next column, from  $a(9,2)$  to  $a(22,2)$  are 0.2, and elements  $a(19,3)$  to  $a(32,3)$  of the third column,  $a(29,4)$  to  $a(42,4)$  of the fourth column and  $a(39,5)$  to  $a(50,5)$  of the fifth column are equal to 0.2 as well. We want to calculate  $E_t(L - y)^+$  at  $t = 1$  for the given  $y = 50, 75, 100, 200$ . We will compare the results among Monte Carlo simulation, the combined method with the interval of buckets equal to 1 unit (A), and the combined method whose intervals of buckets are in different ranges. We divide the range of buckets to be equal to 1 unit from 0 to 110 and shift 3 units each bucket from 110 to 200. Next buckets are set at 700 and 1,275 (B).

**Case I:** compare between Monte Carlo simulation and (A). Then the value of  $E_t(L - y)^+$  at  $t = 1$  is

*Table 1*

<b>y</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>Result of combined method</b>	<b>Monte Carlo</b>	<b>95% interval</b>
<b>50</b>	4.4497	0.3861	0.046	4.8588	4.7847	4.5975 - 4.9719
<b>75</b>	1.5082	0.2818	0.0547	1.81735	1.773	1.5888 - 1.9602
<b>100</b>	0.4352	0.1458	0.0473	0.60465	0.5888	0.4016 - 0.776
<b>200</b>	0.0011	0.0016	0.0022	0.0038	0.004	-0.1832 - 0.1912

where all alphas are the coefficient of Kibble's series. We also compute values of confidence interval by first computing mean then variance from a set of outputs from Monte Carlo simulation. Finally, we set a level of significant and calculate the confidence interval.

*Case II:* compare between Monte Carlo simulation and (B). Then the value of  $E_t(L - y)^+$  at  $t = 1$  is

*Table 2*

<b>y</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>Result of combined method</b>	<b>Monte Carlo</b>	<b>95% interval</b>
<b>50</b>	4.4497	0.3861	0.046	4.8588	4.7847	4.5975 - 4.9719
<b>75</b>	1.5082	0.2818	0.055	1.8175	1.773	1.5888 - 1.9602
<b>100</b>	0.4352	0.1458	0.0473	0.6047	0.5888	0.4016 - 0.776
<b>200</b>	0.0011	0.0016	0.0022	0.0038	0.004	-0.1832 - 0.1912

We find out that even we divide intervals to be 1 unit each bucket or we split them into different ranges they give us accurate value compare to Monte Carlo simulation with a simulation number of 1,000,000 times at 95% confidence interval. And this method has an advantage in term of computing time. For Monte Carlo simulation, to receive the answer it takes around 2.5 hours for a simulation number of 1,000,000 times using MATLAB program on personal computer (Intel Core 2 Duo CPU 1.8 GHz). While it takes only about 140 seconds for (A) and 13 seconds for (B). This means, in case I, it was 60 times faster and 670 times faster in case II.

In table 3, we show the answers we get from the combined method in many ways of bucket arrangement.

Table 3

<b>y</b>	<b>Case*</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>E(L- y)+</b>	<b>Time(sec.)**</b>
<b>50</b>	1	4.4497	0.3861	0.046	4.8588	138.157
	2	4.5368	0.3864	0.3463	5.09635	67.56% faster
	3	4.4497	0.3866	-0.1997	4.73645	90.23% faster
	4	4.4497	0.3861	0.046	4.8588	90.39% faster
	5	4.4497	0.3861	0.046	4.8588	90.95% faster
<b>100</b>	1	0.4352	0.1458	0.0473	0.60465	136.781
	2	0.4457	0.1472	0.5662	0.876	68.63% faster
	3	0.4353	0.1462	-0.1984	0.4823	89.27% faster
	4	0.4352	0.1458	0.0472	0.6046	90.45% faster
	5	0.4352	0.1458	0.0473	0.60465	90.69% faster
<b>200</b>	1	0.0011	0.0016	0.0022	0.0038	139.578
	2	0.0012	0.0017	0.0106	0.0082	69.50% faster
	3	0.0011	0.0016	8.21E-05	0.0027	90.92% faster
	4	0.0011	0.0016	0.0021	0.00375	90.12% faster
	5	0.0011	0.0016	0.0022	0.0038	90.62% faster

\*Case 1: We divide interval of every buckets to be equal to 1 unit.

Case 2: We divide interval of every buckets to be equal to 2 units.

Case 3: We divide the range of buckets to be equal to 1 unit from 0 to 70. Then shift 2 units each bucket until 150. From 150 to 200, the interval is 5 units and for the rest buckets we increase 100 units bucket by bucket until reach 1,300.

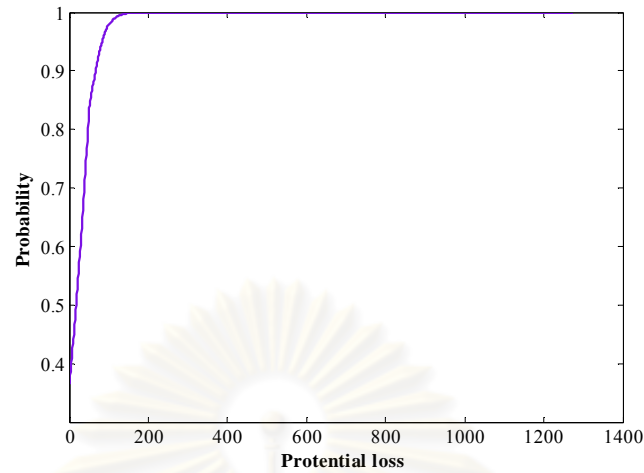
Case 4: We divide the range of buckets to be equal to 1 unit from 0 to 100 and shift 3 units each bucket from 100 to 199. Then, increase 10 units from 199 to 409. And the rest buckets, their intervals are equal to 100.

Case 5: We divide the range of buckets to be equal to 1 unit from 0 to 110 and shift 3 units each bucket from 110 to 200. Next buckets are set at 700 and 1,275.

\*\*For case 2 to 5, the computing times are the times compared to the computing time of case 1.

We see that if we divide buckets too wide the answer will be inaccurate as showed in case 2. In case 3, because we start to expand intervals too early so the answer is also imprecise. For case 4 and 5, we start to expand the buckets around a hundred-something and they give us acceptable accurate answers.

So, there is a procedure to set up the range of buckets. It is important to keep the range of early buckets to be narrow such as 1 unit each. We use a cumulative distribution function (cdf.) of total losses to find a point that we can start to expand a size of buckets.



**Figure 2:** *Cumulative distribution function for example I*

From figure 1 we see that the graph is sharply increased from the amount of potential loss of 0 to roughly 100. Then, it is gradually raised and stabled. So, we can choose a specific number around a hundred, and divide buckets to be quite narrow from 0 to that number and make interval wider and wider for the rest of buckets. It also works for the range with the decimal. Nevertheless, the answer will be inaccurate if we create intervals of buckets too wide.

The most advantage of using probability bucketing is the flexibility of bucket organization. Because we can set the number of buckets to be as small as possible, while obtain the acceptable accurate output. As table 3 shows us, if we well organize the buckets, we can obtain an acceptable accurate output in high speed of computing.

The running time of this combined method depends on the number of buckets and obligors. For this example with 50 obligors, the running time is about 120 seconds per 1,000 buckets.

**Example II**

This example the setting is the same as example I except the potential loss of obligor  $k$  which is equal to  $k^2$  ( $C_k = k^2$ ). For the given  $y = 200, 2000, 5000, 10000$ . We will compare the results of expectation  $E_t(L - y)^+$  between Monte Carlo simulation, and the combined method whose intervals of buckets are in different ranges. We constructed the interval of buckets equal to 1 unit from 0 to 6,000 and increase 1,000 units for the rest buckets until 10,000, then shift 10,000 units bucket by bucket until reach 50,000.

Then the value of  $E_t(L - y)^+$  at  $t = 1$  is

**Table 4**

<b>y</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>Result of combined method</b>	<b>Monte Carlo</b>	<b>95% interval</b>
<b>200</b>	746.3768	2.6716	0.0904	749.0936	745.9595	738.6836 - 753.2354
<b>2000</b>	152.6086	12.8832	1.3904	166.187	166.1214	158.8455 - 173.3973
<b>5000</b>	4.5047	2.2788	1.0769	7.32195	7.3769	0.101 - 14.6528
<b>10000</b>	0.0025	0.0052	0.0101	0.01275	0.0147	-7.2612 - 7.2906

From the table above, we see that the results from the combined method are quite precise and this takes around 50 times faster than Monte Carlo method. This shows us that even the potential losses are big, the answer is still accurate.

In table 5, we illustrate the answers we get from the combined method in many ways of bucket arrangement.

**Table 5**

<b>y</b>	<b>Case*</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>E(L- y)+</b>	<b>Time(sec.)**</b>
<b>2000</b>	1	152.437	12.882	1.5444	166.0912	5763.7
	2	152.5224	12.8814	1.7464	166.277	96.55% faster
	3	152.5225	12.8811	1.6687	166.23795	97.21% faster
<b>5000</b>	1	4.499	2.2769	1.0603	7.30605	5884
	2	4.5012	2.2771	0.9834	7.27	96.58% faster
	3	4.5013	2.2764	1.1279	7.34165	97.25% faster
<b>10000</b>	1	0.0025	0.0052	0.0101	0.01275	6023.1
	2	0.002	0.0045	0.0088	0.0109	96.71% faster
	3	0.002	0.0045	0.009	0.011	97.31% faster

\*Case 1: We divide the range of buckets to be equal to 1 unit from 0 to 6,000. Then shift 1,000 units each bucket until 10,000. And the rest buckets, their interval are equal to 10,000 until reach 50,000.

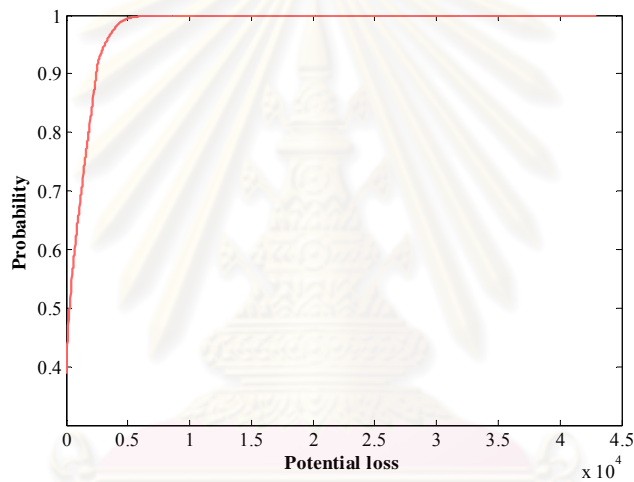
Case 2: We split the interval of buckets to be equal to 2 units from 0 to 1,000. Then shift 5 units

each bucket until 2,000. From 2,000 to 5,000, the interval is 10 units, next shift 100 units until 6,000. From 6,000 to 10,000, the interval is 1,000 units. And the rest buckets, their interval are equal to 10,000.

*Case 3:* We split the interval of buckets to be equal to 2 units from 0 to 500, 3 units from 500 to 1,001, 5 units from 1,001 to 2,001, 10 units from 2,001 to 5,001, 100 units from 5,001 to 6,001 and 1,000 units from 6,001 to 10,001. Then, shift 10,000 units for the rest buckets.

\*\*For case 2 and 3, the computing times are the times compared to the computing time of case 1.

From table 5, we know that since the potential loss is large, we can set the buckets a bit wider from the beginning. And for the well organized buckets, the speed of computation increases a lot while the answer is still precise.



**Figure 3:** *Cumulative distribution function for example II*

The graph above illustrates that we should separate buckets to be quite narrow from 0 to about 4,000. Then, we can extend it to be wider for the rest.

To compare the difficulty with the previous example, we found out that this example is a bit more complicated than the previous. Since the potential losses are high, the way to arrange buckets becomes more difficult. If we split buckets too narrow, it will take a great deal of time. On the other hand, if we separate buckets too wide, the output will be inaccurate. So, the difficulty is to organize size of buckets to be fit with the high potential loss data



**Example III**

The setting of this example are also the same as given in example I, but a default probability of each obligor is changed from 0.02 to be as follow.

**Table 6**

<b><i>k</i></b>	<b>Prob.</b>	<b><i>k</i></b>	<b>Prob.</b>	<b><i>k</i></b>	<b>Prob.</b>	<b><i>k</i></b>	<b>Prob.</b>	<b><i>k</i></b>	<b>Prob.</b>
<b>1</b>	0.9780	<b>11</b>	0.0730	<b>21</b>	0.0216	<b>31</b>	0.0483	<b>41</b>	0.9382
<b>2</b>	0.5035	<b>12</b>	0.6033	<b>22</b>	0.4505	<b>32</b>	0.4970	<b>42</b>	0.8200
<b>3</b>	0.3790	<b>13</b>	0.0386	<b>23</b>	0.3257	<b>33</b>	0.0520	<b>43</b>	0.5696
<b>4</b>	0.7677	<b>14</b>	0.5199	<b>24</b>	0.7477	<b>34</b>	0.7408	<b>44</b>	0.8078
<b>5</b>	0.9288	<b>15</b>	0.6465	<b>25</b>	0.7872	<b>35</b>	0.2778	<b>45</b>	0.2424
<b>6</b>	0.1400	<b>16</b>	0.0544	<b>26</b>	0.1802	<b>36</b>	0.7501	<b>46</b>	0.6232
<b>7</b>	0.9159	<b>17</b>	0.5371	<b>27</b>	0.9480	<b>37</b>	0.9847	<b>47</b>	0.8913
<b>8</b>	0.5835	<b>18</b>	0.8405	<b>28</b>	0.8051	<b>38</b>	0.0957	<b>48</b>	0.6960
<b>9</b>	0.4063	<b>19</b>	0.4471	<b>29</b>	0.2531	<b>39</b>	0.6710	<b>49</b>	0.8847
<b>10</b>	0.3989	<b>20</b>	0.9035	<b>30</b>	0.0307	<b>40</b>	0.0324	<b>50</b>	0.5163

We want to calculate  $E_t(L - y)^+$  at  $t = 1$  for the given  $y = 50, 75, 100, 200$ . We will compare the results among Monte Carlo simulation, the combined method with the interval of buckets equal to 1 unit, and the combined method whose intervals of buckets are in different ranges. We separate the buckets to be 1 unit each from 1 to 690. Then, shift 5 units from 690 to 900. Next buckets are set at 1,000 and 1,275 (B).

**Case I:** compare between Monte Carlo simulation and (A). Then the value of  $E_t(L - y)^+$  at  $t = 1$  is

**Table 7**

<b>y</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>result</b>	<b>Monte Carlo</b>	<b>95% interval</b>
<b>50</b>	639.2463	-0.0017	0	639.2446	639.753	639.1939 - 640.3121
<b>75</b>	614.2463	-0.0017	0	614.2446	614.7146	614.1555 - 615.2737
<b>100</b>	589.2463	-0.0017	0	589.2446	589.5963	589.0372 - 590.1554
<b>200</b>	489.2463	-0.0017	0	489.2446	489.734	489.1749 - 490.2931

**Case II:** compare between Monte Carlo simulation and (B). Then the value of  $E_t(L - y)^+$  at  $t = 1$  is

*Table 8*

<b>y</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>result</b>	<b>Monte Carlo</b>	<b>95% interval</b>
<b>50</b>	639.2349	0.0167	0.9513	639.7273	639.753	639.1939 - 640.3121
<b>75</b>	614.2349	0.0167	0.9513	614.7273	614.7146	614.1555 - 615.2737
<b>100</b>	589.2349	0.0167	0.9513	589.7273	589.5963	589.0372 - 590.1554
<b>200</b>	489.2349	0.0167	0.9513	489.7273	489.734	489.1749 - 490.2931

In this example, we want to know whether the probability of each obligor is not the same and quite high the result is acceptable. The results from the combined method and Monte Carlo simulation show that they are almost the same.

*Table 9*

<b>y</b>	<b>Case*</b>	<b>alpha 0</b>	<b>alpha 1</b>	<b>alpha 2</b>	<b>E(L- y)+</b>	<b>Time(sec.)**</b>
<b>50</b>	1	639.2463	-0.0017	-1.12E-06	639.2446	137.797
	2	639.0985	21.395	-8.72E+03	-3699.7065	64.52% faster
	3	449.2114	-1.0464	-0.0536	448.1382	90.66% faster
	4	639.2484	-0.0013	-0.0028	639.2457	31.73% faster
	5	639.2349	0.0167	0.9513	639.7273	42.45% faster
<b>100</b>	1	589.2463	-0.0017	-1.43E-06	589.2446	138.375
	2	589.0985	21.395	-8.72E+03	-3749.7065	68.65% faster
	3	399.2114	-1.0464	-0.0536	398.1382	90.39% faster
	4	589.2484	-0.0013	-0.0028	589.2457	32.24% faster
	5	589.2349	0.0167	0.9513	589.7273	43.32% faster
<b>200</b>	1	489.2463	-0.0017	-8.10E-07	489.2446	138.203
	2	489.0985	21.395	-8.72E+03	-3849.7065	68.24% faster
	3	299.2114	-1.0464	-5.36E-02	298.1382	90.10% faster
	4	489.2484	-0.0013	-0.0028	489.2457	33.62% faster
	5	489.2349	0.0167	0.9513	489.7273	42.86% faster

\*Case 1: We divide interval of every buckets to be equal to 1 unit.

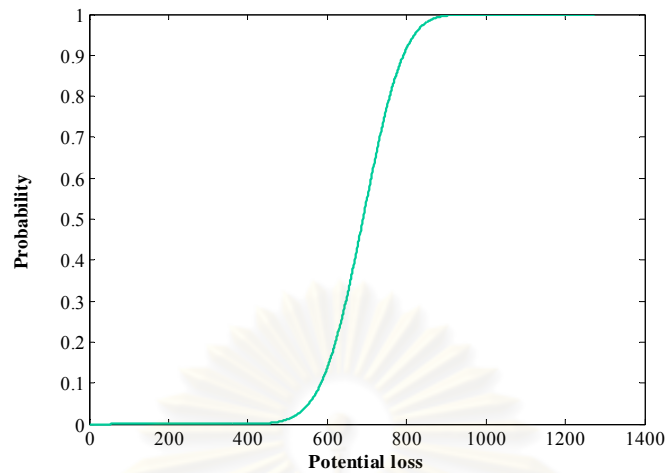
Case 2: We divide interval of every buckets to be equal to 2 units.

Case 3: We divide the range of buckets to be equal to 1 unit from 0 to 100 and shift 3 units each bucket from 100 to 199. Then, increase 10 units from 199 to 409. And the rest buckets, their interval are equal to 100.

Case 4: We divide the range of buckets to be equal to 1 unit from 0 to 800 and shift 3 units each bucket from 800 to 902. Then, increase 200 units from 902 to 1,302.

Case 5: We separate the buckets to be wide 1 unit each from 1 to 690. Then, shift 5 units from 690 to 900. Next buckets are set at 1,000 and 1,275.

\*\*For case 2 to 5, the computing times are the times compared to the computing time of case 1.



**Figure 4:** *Cumulative distribution function for example III*

From table 9 and figure 3, to create buckets that suit the data we have, it seems that we can set the broad intervals of early buckets and narrow those in the middle, then expand again for the rest buckets. For example, we set the interval to be 5 units each from 0 to 500, and 1 unit each from 500 to 900, then 5 units each from 900 to 1275. But, we find that the graph is “S” shape though, we still need to set very narrow buckets from the beginning as previous examples.

Comparing with two examples above, the high default probabilities do not make the bucket arrangement more difficult. The only one thing making the bucket arrangement more complex is the high amount of potential losses.

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## CHAPTER IV

### RESULTS

We find out that the combined method, whose sizes of buckets are arranged gives us the acceptable accurate value as the output of the method with the size of buckets is 1 unit each to estimate the CDO tranche price at time  $t$ . And it is much less time consuming to calculate compare to Monte Carlo simulation. This method works in both high and low potential loss cases. Furthermore, no matter the default probabilities of each obligor are high or low, this method still gives us the precise answer. However, the method is suitable just for portfolios that have low correlation. Also, the bucket arrangement is very important. If we set buckets too wide, the speed of computation increase, but the answers are not accurate. So, we need to arrange bucket sizes that make the calculation time as fastest as possible but still give us an acceptable value. We found that high default probabilities do not affect the difficulty of the bucket arrangement compared with low default probabilities given the same potential losses. But the difficulty is raised when the potential losses are high, because for high potential losses, we can set a size of early bucket wider than 1 unit each to fit the data. However we have to work harder to find that number; a size of early buckets, as it is easy to get an unacceptable output.

This combined method gives us the acceptable answer in high speed of time; however it is quite sensitive in the way of setting up buckets. So, we need to be careful when we set them up. There is a procedure to set up the range of buckets. It is important to keep the range of early buckets to be narrow such as 1 unit each. We use a cumulative distribution function (cdf.) of total losses to find a point that we can start to expand a size of buckets.

## CHAPTER V

### CONCLUSION

We develop a method which improves the calculation speed and is less complicated in term of computation to approximate the CDO tranche price. We combine two techniques; the correlation expansion (P.Glasserman and S.Suchintabandit, 2007) which gives an advantage that pricing in independent obligor models is easy to compute and the probability bucketing approach (John Hull and Alan White, 2004) which gives the advantage in term of calculation speed. We find that we can use this combined method which gives us the acceptable output to estimate the CDO tranche price at time  $t$  in high speed of time. However, we need to be careful when we organize sizes of buckets because it is quite sensitive. If we disorganize buckets, the result could be rejected.



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## REFERENCES

- Abate, J., Whitt, W. 1992. The Fourier-Series Method for Inverting Transforms of Probability Distributions. Queueing Systems 10 (1992): 5-58.
- Abate, J., Whitt, W. 1995. Numerical Inversion of Laplace Transforms of Probability Distributions. ORSA Journal on Computing Vol. 7, 1 (Winter 1995): 38-43.
- Ãgca, S., Chance, D. M. 2003. Speed and Accuracy Comparison of Bivariate Normal Distribution Approximations for Option Pricing. Journal of Computational Finance Vol.6 , 4: 61-70.
- Altman, E.I., B. Brady, A. Resti, and A. Sironi. 2002. The Link Between Default and Recovery Rates: Implications for Credit Risk Models and Procyclicality. Working paper, Stern School of Business, New York University (April 2002).
- Andersen, L., J. Sidenius, and S. Basu. 2003. "All your hedges in one basket. RISK Vol.16, 11 (November 2003): 67-72.
- Andersen, L., J. Sidenius. 2004. Extensions to the Gaussian Copula: Random Recovery and Random Factor Loadings. Working paper, Bank of America. June 2004.
- Cantor, R., D.T. Hamilton, and S. Basu. 2002. Default and Recovery Rates of Corporate Bond Issuers. Moody's investors Services (February 2002).
- Duffie, D., and K. Singleton. 1999. Modeling Term Structures of Defaultable Bonds. The Review of Financial Studies Vol. 12 (1999): 687-720.
- Gregory, J., Laurent, J.-P. 2004. In the Core of Correlation. RISK (October 2004): 87-91.

- Gupta, S. S. 1963. Probability Integrals of Multivariate Normal and Multivariate t. The Annals of Mathematical Statistics Vol.34, 3: 792-828.
- Hull, J. 1977. Dealing with Dependence in Risk Simulations. Operational Research Quarterly Vol. 28, 1 (1977): 201-218.
- Hull, J., and A. White. 2003. Valuing Credit Default Swap Options. The Journal of Derivatives Vol. 10, 3 (Spring 2003): 40-50.
- Hull, J., and A. White. 2004. Valuation of a CDO and an n-th to Default CDS Without Monte Carlo Simulation. The Journal of Derivatives Vol. 12, 2 (Winter 2004): 8-23.
- Laurent, J.-P., and J. Gregory. 2003. Basket Default Swaps, CDOs and Factor Copulas. Working paper, ISFA Actuarial School, University of Lyon.
- Li, D.X. 2000. On Default Correlation: A Copula Approach. The Journal of Fixed Income Vol. 9 (March 2000): 43-54.
- Li D.X. 1999. The Valuation of Basket Credit Derivatives. CreditMetrics Monitor (April 1999): 34-50.
- Merino S.& M. Nyfeler. 2002. Calculating Portfolio loss. RISK (August 2002): 82-86.
- Merton, R.C. 1974. On the Pricing of Corporate Debt: The Risk Structure of Interest Rates. Journal of Finance Vol.29 (1974): 449-470.
- Paul Glasserman and Sira Suchintabandit. 2007. Correlation expansions for CDO pricing. Journal of Banking & Finance (2007): 1375-1398.

Schmidt W. & I. Ward. 2002. Pricing Default Baskets. RISK (January 2002): 111-114.

Sudheer Chava, Catalina Stefanescu and Stuart Turnbull. 2008. Modeling the Loss Distribution. Working paper.

Vasicek O. 1997. The Loan Loss Distribution. Working paper, KMV.

Vasicek, O. A. 1998. A Series Expansion for the Bivariate Normal Integral. Journal of Computational Finance Vol.1, 4: 5-10.



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย





**APPENDICES**

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

## APPENDIX A

### Model

#### Correlation Expansion

A numerical method for computing  $E(L - y)^+$  which has been developed by P.Glasserman and S.Suchintabandit [2007] shows that the difficulty of computing the tranche price  $E(L - y)^+$ , whose obligors are correlated, can be expressed as a series of prices in independent obligor model. The form of this estimation method is:

$$E(L - y)^+ \approx \sum_J \text{const} \tilde{E}_J(L - y)^+ \quad (1)$$

This pricing method is based on computing joint probabilities of correlated normal random variables which is interpreted as the probability that obligors default at the same time.

$$P(X_1 > x_1, \dots, X_M > x_M) \approx \sum_J \text{const} \cdot \tilde{p}_1^{(J)} \dots \tilde{p}_M^{(J)}$$

where  $X_i$  are correlated  $N(0,1)$  random variables, and  $x_i$  are real numbers.

So, the first step is to estimate the value of  $\tilde{p}_1^{(J)} \dots \tilde{p}_M^{(J)}$  which represent the default joint probabilities of an independent obligor portfolio. Each obligor has a marginal default probability of  $\tilde{p}_i^{(J)}$ . The probability  $\tilde{p}_i^{(J)}$  can be computed by using the perturbation formula. Then, we can find the value of  $\tilde{E}_J(L - y)^+$ .

The next step is to calculate the value of  $\text{const}$ , and substitute them into (1). Therefore we receive  $E_i(L - y)^+$ .

#### **The Perturbation Formula**

According to definition 2.1 of perturbed probability in P.Glasserman and S.Suchintabandit, 2007, we can calculate  $\tilde{p}_i^{(J)}$  which is the perturbed probability of  $p_i = P(X_i > x_i) = \overline{\Phi}(x_i)$  ;  $\overline{\Phi}(x_i) = 1 - \Phi(x_i)$  with  $\Phi$  the cumulative normal distribution function by

$$\tilde{p}_i^{(J)} := p_i + \varphi(x_i) \sum_{k=1}^n v_J(i, k) H_{k-1}(x_i)$$

where,  $\varphi(x_i)$  is the standard normal density function

$H_n(x)$  is the Hermite polynomial of degree  $n$ .

$v_J(i, k)$  are computed as follow,

Assume that  $\rho_{ik}$  have a generalized  $d$ -factor structure, for all  $i, k = 1, \dots, M$

$$\rho_{ik} = \sum_{j=1}^d \varpi_j a_{ij} a_{kj}$$

where  $\varpi_j$  and  $a_{ij}$  are real scalars and rewrite  $a_{ij}$  as  $a(i, j)$ . For negative index  $j$ , we define the principle  $a(i, -j) = -a(i, j)$  and  $a(i, 0) = 0$ .

Let  $D^n$  refer to the Cartesian product and  $D^0 = \{\emptyset\}$ . Define  $D := \{0, \pm 1, \pm 2, \dots, \pm d\}$ , where  $d$  is the dimension of the factorization.

$$D^n := \underbrace{D \times D \times \dots \times D}_n = \{(j_1, \dots, j_n) : j_1, \dots, j_n \in D\}$$

Given  $J = (j_1, \dots, j_n)$  from the set  $D^n$  and a set of perturbation parameters  $\theta_1, \dots, \theta_n$ , the recursion of  $v_J(i, k)$  can be defined as,

$$v_J(i, k) = v_{J'}(i, k) + \theta_n a(i, j_n) v_{J'}(i, k-1)$$

where  $1 \leq i \leq M$  and  $k$  is an integer.  $J' = (j_1, \dots, j_{n-1}) \in D^{n-1}$  is the truncation of  $J$  by deleting the  $n^{\text{th}}$  coordinate. For  $J \in D^n$ ,  $v_J(i, 0) := 1$  and  $v_J(i, k) := 0$  where  $k > n$ . For all  $J \in D^n$  and  $k > 0$ , as  $\theta_1, \dots, \theta_n \rightarrow 0$ ,  $v_J(i, k) \rightarrow 0$ .

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

To make this clearer we can illustrate it as

$$\begin{aligned}
 J = \phi & : 1 \\
 J = j_1 & : 1 = \theta_1 a_{j_1} \times 0 + 1 \\
 & v_j(i,1) = \theta_1 a_{j_1} \times v_{j'}(i,0) + 0 \\
 J = (j_1, j_2) & : 1 = \theta_2 a_{j_2} \times 0 + 1 \\
 & v_j(i,1) = \theta_2 a_{j_2} \times 1 + v_{j'}(i,1) \\
 & v_j(i,2) = \theta_2 a_{j_2} \times v_{j'}(i,1) + 0 \\
 J = (j_1, j_2, j_3) & : 1 = \theta_3 a_{j_3} \times 0 + 1 \\
 & v_j(i,1) = \theta_3 a_{j_3} \times 1 + v_{j'}(i,1) \\
 & v_j(i,2) = \theta_3 a_{j_3} \times v_{j'}(i,1) + v_{j''}(i,2) \\
 & v_j(i,3) = \theta_3 a_{j_3} \times v_{j''}(i,2) + 0
 \end{aligned}$$

To calculate joint probabilities, the method is explained here. Let  $P_t$  be the probability measure under which  $X_1, \dots, X_M$  are  $N(0,1)$  random variables whose covariance matrix is

$$C_t := \begin{bmatrix} 1 & t\rho_{12} & t\rho_{13} & \dots & t\rho_{1M} \\ t\rho_{21} & 1 & t\rho_{23} & & t\rho_{2M} \\ t\rho_{31} & t\rho_{32} & 1 & & t\rho_{3M} \\ \vdots & & & \ddots & \\ t\rho_{M1} & t\rho_{M2} & t\rho_{M3} & & 1 \end{bmatrix}$$

As  $t$  varies from 0 to 1, there exist real numbers  $b_0, b_1, \dots$ ;

$$P_t(X_1 > x_1, \dots, X_M > x_M) = b_0 + b_1 t + b_2 \frac{t^2}{2!} + \dots$$

To estimate  $b_n$ , P.Glasserman and S.Suchintabandit have introduced an efficient method. With perturbation parameters  $\theta_1 = \dots = \theta_n = \theta$ , the coefficient  $b_n$  can be illustrate as the limit

$$\sum_{J \in D^n} w_J \tilde{P}_1^{(J)} \dots \tilde{P}_M^{(J)} \xrightarrow{\theta \downarrow 0} b_n$$

where the weight  $w_J$  is given as follows:  $w_J := 1$  if  $n = 0$ ; otherwise  $w_J = \varpi_J / (2\theta^2)^n$ . Define  $\varpi_0 := -2(\varpi_1 + \dots + \varpi_d)$  and  $\varpi_J := \varpi_{|j_1|} \dots \varpi_{|j_n|}$  for all  $J = (j_1, \dots, j_n)$  from the set  $D^n$ .

The appropriate value of  $\theta$  to compute the perturbed probabilities  $\tilde{p}_i^{(J)}$  and the weighted sum can be set in moderately small to achieve a precise value of  $b_n$ . Since they found that as  $\theta \rightarrow 0$ , the sum converges to  $b_n$  very quickly. Then we can estimate the joint probability.

For  $E_i(L - y)^+$  whose obligors are correlated, they apply the same method as above, calculating the joint probability from independent probabilities. Let  $\tilde{E}_J(L - y)^+$  indicate the expectation of a portfolio whose obligors default with perturbed probabilities  $\tilde{p}_i^{(J)}$  independently. Then, we have

$$\begin{aligned} E(L - y)^+ &\approx \sum_J \text{const} \tilde{E}_J(L - y)^+ \\ &= \alpha_0 + \alpha_1 t + \alpha_2 \frac{t^2}{2!} + \dots \end{aligned}$$

Then the coefficient  $\alpha_n$  in the above expansion can be illustrated as follows

$$\sum_{J \in D^n} w_J \tilde{E}_J(L - y)^+ \xrightarrow{\theta \downarrow 0} \alpha_n$$

The weight  $w_J$  is calculated the same as above.

## Probability Bucketing Approach

John Hull and Alan White [2004] develop a technique called probability bucketing with the purpose of estimating the probability that the total loss lies in the  $k$ -th bucket for all  $k$ . They build up the probability distribution of the loss by time  $T$ , conditional on the values of the risk factors  $Z_1, Z_2, \dots, Z_d$ . Assume that the recovery rate is known and can be stochastic, and there are  $M$  obligors. This technique can be explained as follows

Initially, divide potential losses into ranges  $\{0, b_0\}, \{b_0, b_1\}, \dots, \{b_{K-1}, \infty\}$  and refer to  $\{0, b_0\}$  as the 0-th bucket,  $\{b_{k-1}, b_k\}$  as the  $k$ -th bucket ( $1 \leq k \leq K-1$ ), and  $\{b_{K-1}, \infty\}$  as the  $K$ -th bucket. It is best to set  $b_0 = 0$  and  $b_k - b_{k-1} = u$  ( $1 \leq k \leq K-1$ ) for some constant  $u$ .

Let:  $P_T^{(k)}$  = the conditional probability that the loss by time T will be in the  $k$ -th bucket. ( $p_T(k|Z_1, \dots, Z_d)$ )  
 $A_k$  = the mean loss that the loss is in  $k$ -th bucket ( $1 \leq k \leq K$ ).

Then, calculate  $P_T^{(k)}$  and  $A_k$  by first assuming that there are no obligors, next assuming that there is one obligor, followed by assuming that there are two obligors, and so on. In case that there are no obligors, we are sure there will be no loss. As a result  $P_T^{(0)} = 1$  and  $P_T^{(k)} = 0$  for  $k > 0$ , also  $A_0 = 0$ . For the initial value  $A_k$  for  $k > 0$ , let  $A_k = 0.5(b_{k-1} + b_k)$  for  $1 \leq k \leq K-1$  and  $A_k = b_{K-1}$ .

The only assumption in the iterative procedure is that we concentrate at the recent value of  $A_k$  for all the probability associated with bucket  $k$ . Suppose that we have calculated  $P_T^{(k)}$  and  $A_k$  when the first  $i-1$  obligors are considered.

When  $u(k) > k$ , the formulas are:

$$P_T^{(k)} = P_T^{(k)*} - P_T^{(k)*} p_i$$

$$P_T^{u(k)} = P_T^{u(k)*} + P_T^{(k)*} p_i$$

$$A_k = A_k^*$$

$$A_{u(k)} = \frac{P_T^{u(k)*} A_{u(k)}^* + P_T^{(k)*} p_i (A_k^* + Y_i)}{P_T^{u(k)*} + P_T^{(k)*} p_i}$$

where :

$Y_i$  = the loss given default from the  $i$ -th obligor

$p_i$  = the default probability of the  $i$ -th obligor

$u(k)$  = the bucket including  $A_k + Y_i$  for  $0 \leq k \leq K$

And  $P_T^{(k)*}$ ,  $P_T^{u(k)*}$ ,  $A_k^*$ ,  $A_{u(k)}^*$  are the values of  $P_T^{(k)}$ ,  $P_T^{u(k)}$ ,  $A_k$ ,  $A_{u(k)}$  before the probability shift is considered.

The effect of  $i$ -th obligor is to move an amount of probability  $P_T^{(k)*} p_i$  from bucket  $k$  to  $u(k)$  ( $0 \leq k \leq K$ ).

If  $u(k) = k$ , then the formulas are:

$$P_T^{u(k)} = P_T^{(k)*}$$

$$P_T^{(k)} = P_T^{(k)*}$$

$$A_k = A_k^* + p_i Y_i$$

$$A_{u(k)} = A_k^* + p_i Y_i$$

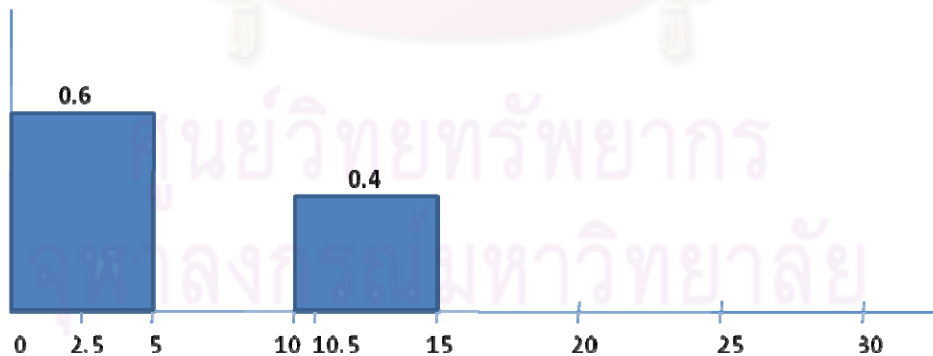
We then obtain the total loss distribution when all M obligors have been considered.

### Example

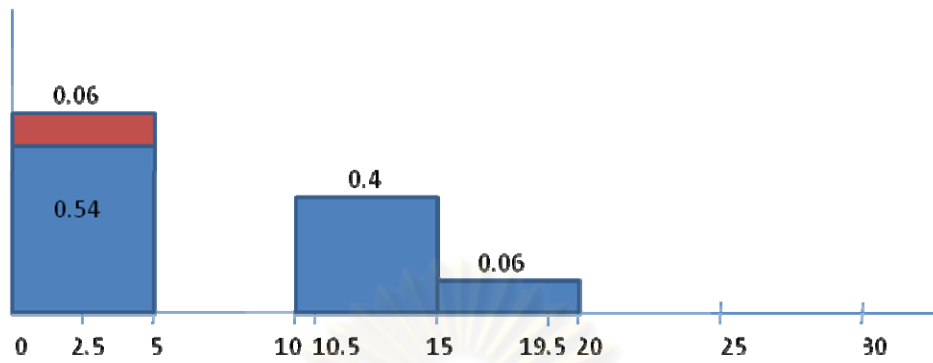
<i>i</i> -th obligor	$Y_i$	$p_i$
1	8	0.4
2	17	0.1
3	4	0.5

Suppose there are three obligors in portfolio and we would like to find the total loss distribution of these three obligors. Let each interval equal to 5.

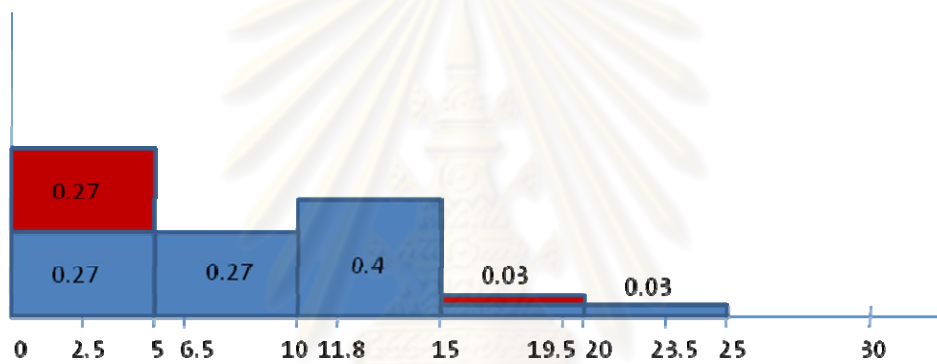
1. Suppose first, there are no obligors, thus there are no losses. Then assume that the 1-st obligor is considered.



2. Assume that the 2-nd obligor has been occurred.



3. Next, the 3-rd obligor is considered.





## APPENDIX B

### Probability bucketing code.

```
function [P,A] = BKT(loss,p,interval)
bin = length(interval)-1;
P = zeros(bin,1);
P(1,1) = 1;
A = (interval(1:bin) + interval(2:bin+1))./2;
M = 1;

for k = 1:length(loss)
    AA = A(1:M) + loss(k);
    U = sum(interval(:,ones(1,M))' < AA(:,ones(1,bin+1))),2);
    S1 = sparse(M,bin);
    S2 = sparse(M,bin);
    S1((U-1).*M + [1:M]') = P(1:M)*p(k);
    S2((U-1).*M + [1:M]') = P(1:M)*p(k).*AA;
    S1 = sum(S1,1)';
    S2 = sum(S2,1)';
    u = find(S1);
    A(u) = (A(u).*P(u) + S2(u))./(P(u) + S1(u));

    P(1:M) = P(1:M) - P(1:M)*p(k) ;
    P(u) = P(u) + S1(u) ;

    M =sum(interval < sum(loss(1:k)));
end
end
```

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

### Correlation Expansion code

```

function [p,PP,J] = CORR(d,n,I,theta)
%d = number of columns
%n = number of dimensions
%I = number of obligors
p = xlsread('pi_new.xls','P_new');
A = xlsread('pi_new.xls','A_new');
x = norminv(1-p,0,1);
Y = pdf('Normal',x,0,1);
J = dimension(n,d);
a = zeros(1,I);
E = zeros(1,n);
v = zeros(I,n+1);
PP = zeros(size(J,1),I);
for row = 1:1:size(J,1);
    for i = 1:1:I
        for m = 1:1:n
            if J(row,m) == 0
                a(i) = 0;
            else a(i) = sign(J(row,m))*A(i,abs(J(row,m)));
            end
            E(m) = a(i);
        end
        L(row,i) = {perturb1(theta,E,n)};
        u = L{row,i};
        for k = 2:1:n+1;
            v(i,k) = u(1,k)*herm(k-2,x(i));
            V(row,:) = {sum(v,2)'};
        end
    end
    PP(row,:) = p + Y.*V{row};
end
end

```

### Dimension function code

```

function a = dimension(n,d)
noValue = 2*d+1;
a = zeros(noValue^n,n);
for k=1:n
    value = -d;
    lot = (noValue^n)/(noValue^k);
    i = 1;
    while i <= size(a,1)
        j = 1;
        while j<=lot
            a(i,k) = value;
            i = i+1;
            j = j+1;
        end
        value = value+1;
        if value > d
            value = -d;
        end
    end
end
end
end

```

Perturb function code

```
function L = perturb1(theta,E,n)
    L = [0 1]*theta*E(1) + [1 0];
    i = 2;
    while size(L) <= n
        L = [0 L]*theta*E(i) + [L 0];
        i = i + 1;
    end
end
```

Hermite function code

```
function hermite = herm(m,n)
if m == 0
    hermite = 1;
elseif m == 1
    hermite = n;
elseif m == 2
    hermite = n^2-1;
elseif m == 3
    hermite = n^3-3*n;
elseif m == 4
    hermite = n^4-6*n^2+3;
elseif m == 5
    hermite = n^5-10*n^3+15*n;
elseif m == 6
    hermite = n^6-15*n^4+45*n^2-15;
elseif m == 7
    hermite = n^7-21*n^5+105*n^3-105*n;
elseif m == 8
    hermite = n^8-28*n^6+210*n^4-420*n^2+105;
elseif m == 9
    hermite = n^9-36*n^7+378*n^5-1260*n^3+945*n;
else m == 10
    hermite = n^10-45*n^8+630*n^6-3150*n^4+4725*n^2-945;
end
```

ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย

### Code for pricing CDO tranche at time t

```
function [B0,B1,B2] = M_test(d,n,I,theta,yy)

loss = xlsread('pi_new.xls','loss_new');
interval = xlsread('pi_new.xls','interval_new');
[p,PP,J] = CORR(d,n,I,theta);
W = abs(sign(J));
W2 = zeros(size(J));

[P1,A1] = BKT(loss,p,interval);

for row = 1:1:size(J,1)
[P,A] = BKT(loss,PP(row,:),'interval');
expected(row,1) = sum(max(0,A-yy).*P);

    if W(row,n) == 0
        W2(row,:) = W(row,:);
        for w = 1:1:n-1
            if W2(row,w) == 0
                W2(row,w) = -10;
            end
        end
    end
end

z = find(W == 0);
W(z) = -10;

B0 = sum(max(0,A1-yy).*P1);
B1 = sum(expected.*(W2(:,1)/(2*theta^2)));
B2 = sum(expected.*(prod(W,2)/(2*theta^2)^2));

end
```

### Monte Carlo simulation code

```
A = xlsread('pi_new.xls','A_new');
P = xlsread('pi_new.xls','P_new');
loss = xlsread('pi_new.xls','loss_new');
x = norminv(1-P,0,1);
m = 1000000;
for k = 1:1:m
E = mvnrnd(0,1,50);
Z = mvnrnd(0,1,5);
for i = 1:length(E)
X = (A(i,:)*Z) + (sqrt(1-(A(i,:)*A(i,:)))*E(i));
I(i) = X > x(i);
end
L(k) = sum(loss.*I');
end
p = L > 200;
expected = sum(max(0,L-200).*p)/m;
```

## BIOGRAPHY

Chanya Siriarayaphan, a Master's degree student in Finance, the Faculty of Commerce and Accountancy, Chulalongkorn University, was born on 18 December 1983 in Bangkok Thailand. She lives with her parents in Lopburi Thailand. She was graduated Bachelor's degree from the Faculty of Science and Technology, major Statistics, minor Actuarial Science, Thammasat University. She used to work for Bangkok Bank before started to do the Master's degree.



ศูนย์วิทยทรัพยากร  
จุฬาลงกรณ์มหาวิทยาลัย