International Journal of Cloud Applications and Computing

April-June 2011, Vol. 1, No. 2

Table of Contents

Research Articles

The Financial Clouds Review

Victor Chang, University of Southampton and University of Greenwich, UK

Chung-Sheng Li, IBM Thomas J. Watson Research Center, USA

David De Roure, University of Oxford, UK

Gary Wills, University of Southampton, UK

Robert John Walters, University of Southampton, UK

Clinton Chee, Commonwealth Bank, Australia

ABSTRACT

This paper demonstrates financial enterprise portability, which involves moving entire application services from desktops to clouds and between different clouds, and is transparent to users who can work as if on their familiar systems. To demonstrate portability, reviews for several financial models are studied, where Monte Carlo Methods (MCM) and Black Scholes Model (BSM) are chosen. A special technique in MCM, Least Square Methods, is used to reduce errors while performing accurate calculations. Simulations for MCM are performed on different types of Clouds. Benchmark and experimental results are presented for discussion. 3D Black Scholes are used to explain the impacts and added values for risk analysis. Implications for banking are also discussed, as well as ways to track risks in order to improve accuracy. A conceptual Cloud platform is used to explain the contributions in Financial Software as a Service (FSaaS) and the IBM Fined Grained Security Framework. This study demonstrates portability, speed, accuracy, and reliability of applications in the clouds, while demonstrating portability for FSaaS and the Cloud Computing Business Framework (CCBF).

Keywords: *3D Black Scholes, Black Scholes Model, Cloud Computing Business Framework, Enterprise Portability for Clouds, Financial Clouds, Least Square Methods, MATLAB and Mathematica Applications on Clouds, Monte Carlo Methods (MCM), Operational Risk*

1. INTRODUCTION

The Global economic downturn triggered by the finance sector is an interdisciplinary research question that expertise from different sectors needs to work on altogether. There are different interpretations for the cause of the problem. Firstly, Hamnett (2009) conducted a study to investigate the cause, and concluded unsustainable mortgage lending leads to out of control status and that the housing bubble and subsequent collapse were result of these. Irresponsible mortgage lending was the cause for Lehman Brother collapse that has triggered global financial crisis. Secondly, Lord Turner, Chair of the Financial Service Authority (FSA), is quoted as follows: "The problem, he said, was that banks' mathematical models assumed a 'normal' or 'Gaussian' distribution of events, represented by the bell curve, which danger-

DOI: 10.4018/ijcac.2011040104

ously underestimated the risk of something going seriously wrong" (Financial Times, 2009). Thirdly, there are reports showing the lack of regulations on financial practice. Currently there are remedies proposed by several governments to improve on this (City A.M., 2010). All the above suggested possibilities contribute to complexity that caused global downturn. However, Cloud Computing (CC) offers a good solution to deal with challenges in risk analysis and financial modelling. The use of Cloud resources can improve accuracy of risk analysis, and knowledge sharing in an open and professional platform (Chang, Wills, & De Roure, 2010a, 2010c). Rationales are explained as follows. The Clouds provide a common platform to run different modelling and simulations based on Gaussian and non-Gaussian models, including less conventional models. The Clouds offer distributed high-performing resources for experts in different areas within and outside financial services to study and review the modelling jointly, so that other models with Monte Carlo Methods and Black Scholes Models can be investigated and results compared. The Clouds allow regulations to be taken with ease while establishing and reminding security and regulation within the Clouds resources.

2. LITERATURE REVIEW

Literature review is presented as follows. Three challenges in business context and Software as a Service (SaaS) are explained. This paper is focused on the third issue, enterprise portability, and how financial SaaS is achieved with portability. Financial models with Monte Carlo methods and Black Scholes models are also explained.

2.1. Three Challenges in Business Context

There are three Cloud Computing problems experienced in the current business context (Chang, Wills, & De Roure, 2010b, 2010c). Firstly, all cloud business models and frameworks proposed by several leading researchers are either qualitative (Briscoe & Marinos, 2009; Chou, 2009; Weinhardt et al., 2009; Schubert, Jeffery, & Neidecker-Lutz, 2010) or quantitative (Brandic et al., 2009; Buyya et al., 2009; Patterson et al., 2009). Each framework is self-contained, and not related to others' work. There are few frameworks or models which demonstrate linking both quantitative and qualitative aspects, and when they do, the work is still at an early stage.

Secondly, there is no accurate method for analysing cloud business performance other than the stock market. A drawback with the stock market is that it is subject to accuracy and reliability issues (Chang, Wills, & De Doure, 2010a, 2010c). There are researchers focusing on business model classifications and justifications for which cloud business can be successful (Chou, 2009; Weinhardt et al., 2009). But these business model classifications need more cases to support them and more data modelling to validate them for sustainability. Ideally, a structured framework is required to review cloud business performance and sustainability in systematic ways.

Thirdly, communications between different types of clouds from different vendors are often difficult to implement. Often work-arounds require writing additional layers of APIs, or an interface or portal to allow communications. This brings interesting research questions such as portability, as portability of some applications from desktop to cloud is challenging (Beaty et al., 2009; Patterson et al., 2009). Portability refers to moving enterprise applications and services, and not just files or VM over clouds.

2.2. Financial Models

Gaussian-based mathematical models have been frequently used in financial modelling (Birge & Massart, 2001). As the FSA has pointed out, many banks' mathematical models assumed normal (Gaussian) distribution as an expected outcome, and might underestimate the risk for something going wrong. To address this, other non-Gaussian financial models need to be investigated and demonstrated for how financial SaaS can be successfully calculated and executed on Clouds. Based on various studies (Feiman & Cearley, 2009; Hull, 2009), one model for pricing and one model for risk analysis should be selected respectively. A number of methods for calculating prices include Monte Carlo Methods (MCM), Capital Asset Pricing Models and Binomial Model. However, the most commonly used method is MCM since MCM is commonly used in stochastic and probabilistic financial models, and provides data for investors' decision-making (Hull, 2009). MCM is thus chosen for pricing. On the other hand, methods such as Fourier series, stochastic volatility and Black Scholes Model (BSM) are used for volatility. As a main stream option, BSM is selected for risk analysis, since BSM has finite difference equations to approximate derivatives. Origins in literature and mathematical formulas in relation to MCM and BSM are presented in the next two sections.

2.2.1. Monte Carlo Methods in Theory

Monte Carlo Simulation (MCS), originated from mathematical Monte Carlo Methods, is a computational technique used to calculate risk analysis and the probability of an event or investment to happen. MCS is based on probability distributions, so that uncertain variables can be described and simulated with controlled variables (Hull 2009; Waters 2008). Originated from Physics, Brownian Motions follow underlying random variables can influence the Black-Scholes models, where stock price becomes

$$dS = \mu S dt + \sigma S dW_t \quad (1)$$

where W is Brownian the dW term here stands in for any and all sources of uncertainty in the price history of the stock. The time interval is divided into M units of length δt from time 0 to T in a sampling path, and the Brownian motion over the interval dt are approximated by a single normal variable of mean 0 and variance δt, and leading to

$$S(k\delta t) = S(0)\exp\left(\sum_{i=1}^{k}\left[\left(\mu - \frac{\sigma^2}{2}\right)\delta t + \sigma\varepsilon_i\sqrt{\delta t}\right]\right) \quad (2)$$

for each k between 1 and M, and each ε_i is drawn from a standard normal distribution. If a derivative H pays the average value of S between 0 and T then a sample path ω corresponds to a set $\{\varepsilon_1, \ldots, \varepsilon_M\}$ and hence:

$$H(\omega) = \frac{1}{M+1}\sum_{k=0}^{M} S(k\delta t) \quad (3)$$

The Monte Carlo value of this derivative is obtained by generating N lots of M normal variables, creating N sample paths and so N values of H, and then taking the mean. The error has order $\varepsilon = O(N^{-1/2})$ convergence in standard deviation based on the central limit theorem.

2.2.2. Black Scholes Model (BSM)

The BSM is commonly used for financial markets and derivatives calculations. It is also an extension from Brownian motion. The BSM formula calculates call and put prices of European options (a financial model) (Hull, 2009). The value of a call option for the BSM is:

$$C(S,t) = SN(d_1) - Ke^{-r(T-t)}N(d_2) \quad (4)$$

where $d_1 = \dfrac{\ln(\frac{S}{K}) + (r + \frac{\sigma^2}{2})(T-t)}{\sigma\sqrt{T-t}}$ and $d_2 = d_1 - \sigma\sqrt{T-t}$

The price for the put option is:

$$P(S,t) = Ke^{-r(T-t)} - S + (SN(d_1) - Ke^{-r(T-t)}N(d_2)) = Ke^{-r(T-t)} - S + C(S,t) \quad (5)$$

For both formulas (Hull, 2009),

- N(•) is the cumulative distribution function of the standard normal distribution
- T - t is the time to maturity
- S is the spot price of the underlying asset
- K is the strike price
- r is the risk free rate
- σ is the volatility in the log-returns of the underlying asset.

2.3. Least Square Methods (LSM) for Monte Carlo Simulations (MCS)

Variance Gamma Processes are used in our previous papers (Chang, Wills, & De Roure, 2010a, 2010c), and although it reduces errors while calculating pricing and risk analysis on Clouds, it can only go up to 20,000 simulations in one go before performance drops off. In addition, it takes approximately 10 seconds for error correction due to stratification of sampling, although it takes less than 1 second for 5,000 simulations per attempt for executing financial applications with Octave 3.2.4 on Clouds. This leads us to investigate other methodology that can offer much more simulations to be executed in one go, in other words, improvements in performance on Clouds while maintaining accuracy and quality of our simulations. Monte Carlo Methods (MCM) are used in our simulations, and this means other methods supporting MCM are required to meet our objectives. Various methods such as stochastic simulation, Terms Structure Models (Piazzesi, 2010), Triangular Methods (Mullen et al., 1988; Mullen & Ennis, 1991), and Least Square Methods (LSM) are studied (Longstaff & Schwartz, 2001; Moreno & Navas, 2001; Choudhury et al., 2008). LSM is chosen because of the following advantages. Firstly, LSM provides a direct method for problem solving, and is extremely useful for linear regressions. LSM only needs a short starting time, and is therefore a good choice. Secondly, Terms Structure Models and Triangular Methods are not necessarily used in the Clouds. LSM can be used in the Clouds, because often jobs that require high computations in the Clouds, need extensive resources and computational powers to run. LSM is suitable if a large problem is divided into several sections where each section can be calculated swiftly and independently. This also allows improvements in efficiency.

Here is the explanation for the LSM. There is a data set (x_1,y_1), (x_2, y_2),....,(x_n, y_n) and the fitting curve $f(x)$ has the deviation d_1, d_1, ..., d_n which are caused from each data point, the least square method produces the best fitting curve with the property as follows

$$Minimum\ Least\ Square\ Error(\prod) = d_1^2 + d_2^2 + ... + d_{x-1}^2 = \sum_{i-1}^{n} d_i^2 = \sum_{i=1}^{n} [y_i - f(x_i)]^2 \quad (6)$$

The least squares line method uses an equation $f(x) = a + bx$ which is a line graph and describes the trend of the raw data set (x_1,y_1), (x_2, y_2),....,(x_n, y_n). The n should be greater or equal to 2 ($n \geq 2$)in order to find the unknowns a and b. So the equation for the least square line is

$$\prod = d_1^2 + d_2^2 + ... + d_n^2 = \sum_{i=1}^{n} d_1^2 = \sum_{i=1}^{n} [y_i - (a + bx_i)]^2 \quad (7)$$

The least squares line method uses an equation $f(x) = a + bx + cx^2$ which is a parabola graph. The n should be greater or equal to 3 ($n \geq 3$)in order to find the unknowns a, b, *and* c. When you get the first derivatives of $\prod$ in parabola, you will have

$$\prod = d_1^2 + d_2^2 + ... + d_{n-1}^2 + d_n^2 = \sum_{i=1}^{n} d_i^2 = \sum_{i=1}^{n} [y_i - (\sum_{i=1}^{n} a + \sum_{i=1}^{n} bx_i + \sum_{i=1}^{n} cx_i^2)]^2 \quad (8)$$

The LSM has been mathematically proven, and allows advanced calculations of complex systems. The LSM is the most suitable for a complex problem divided into several sections where each section runs its own calculations. These complex systems include robot, financial modelling and medical engineering. Longstaff and Schwartz (2001) have developed an algorithm based on LSM Monte Carlo simulations (MCS) to estimate best values precisely. Moreno and Navas (2001) have adopted a similar approach, and demonstrate their algorithm and robustness of LSM MCS for pricing American derivatives. Choudhury (2008) used an approach presented Longstaff and Schwartz, except they focused on code algorithms and performance optimisation. These three papers have demonstrated how LSM can be used for financial computing to achieve accurate estimation and optimisation. Abdi (2009) demonstrate that LSM is very useful for regression and explain why LSM is popular and versatile for calculations. He also states the drawback is that LSM does not cope well with extreme calculations, but such volatile calculations will be handled by 3D Black Scholes (Section 4).

2.4. The Cloud Computing Business Framework

To address the three challenges in business context earlier, the Cloud Computing Business Framework (CCBF) is proposed. The core concept of CCBF is an improved version from Weinhardt's et al. (2009) Cloud Business Model Framework (CBMF) where they demonstrate how technical solutions and Business Models fit into their CBMF. The CCBF is proposed to deal with four research problems:

1. Classification of business models with consolidation and explanations of its strategic relations to IaaS, PaaS and SaaS.
2. Accurate measurement of cloud business performance and ROI.
3. Dealing with communications between desktops and clouds, and between different clouds offered by different vendors, which focus on enterprise portability.
4. Providing linkage and relationships between different cloud research methodologies, and between IaaS, PaaS, SaaS and Business Models.

The Cloud Computing Business Framework is a highly-structured conceptual and architectural framework to allow a series of conceptual methodologies to apply and fit into Cloud Architecture and Business Models. Based on the summary in Section 2.1, our research questions can be summed up as: (1) Classification; (2) Sustainability; (3) Portability and (4) Linkage. This paper focuses on the third research question, Portability, which is described as follows.

Portability: This refers to enterprise portability, which involves moving the entire application services from desktops to clouds and between different clouds. For financial services and organisations that are not yet using clouds, portability involves a lot of investment in terms of outsourcing, time and effort, including rewriting APIs and additional costs. This is regarded as a business challenge. Portability deals with IaaS, PaaS and SaaS. Examples in Grid, Health and Finance will be demonstrated. Financial SaaS (FSaaS) Portability is the focus for this paper.

2.5. Financial Software as a Service (FSaaS)

In relation to finance, portability is highly relevant. This is because a large number of financial applications are written for desktops. There are financial applications for Grid but not all of them are portable onto clouds. Portability often requires rewrites in software design and the API suitable for clouds. Apart from portability, factors such as accuracy, speed, reliability and security of financial models from desktop to clouds must be taken into consideration. The second problem related to finance is there are few financial clouds as described in opening section. Salesforce offers CRM but it is not directly related to financial modelling (FM). Paypal is a payment system and not dealing with financial modelling. Enterprise portability from desktops to clouds, and between different clouds, is useful for businesses and financial services, as they cannot afford to spend time and money migrating the entire applications, API libraries and resources to clouds. Portability must be made as easy as possible. However, there are more advantages in moving all applications and resources to clouds. These added values include the following benefits:

- The community cloud – this encourages groups of financial services to form an alliance to analyse complex problems.
- Risk reduction – financial computing results can be compared and jointly studied together to reduce risks. This includes running other less conventional models (non-Gaussians) to exploit causes of errors and uncertainties. Excessive risk taking can be minimised with the aid of stricter regulations.

Financial Software as a Service (FSaaS) is the proposal for dealing with two finance-specific problems. FSaaS is designed to improve the accuracy and quality of both pricing and risk analysis. This is essential because incorrect analysis or excessive risk taking might cause adverse impacts such as financial loss or severe damage in credibility or credit crunch. Research demonstration is on SaaS, which means it can calculate best prices or risks based on different values in volatility, maturity, risk free rate and so forth on cloud applications. Different models for FSaaS are presented and explained from Section 2.3 onwards, in which Monte Carlo Methods (MCM) and Black Scholes Models (BSM) will be demonstrated as the core models used in FSaaS.

3. FAAS PORTABILITY: MONTE CARLO SIMULATIONS WITH LEAST SQUARE METHODS

This section describes how Financial SaaS portability on clouds can be achieved. This mainly involves Monte Carlo Methods (MCM) and Black Scholes Model (BSM). Before describing how they work and how validation and experiments are done, current practice in Finance is presented as follows. Mathematical models such as MCM are used in Risk Management area, where models are used to simulate the risk of exposures to various types of operational risks. Monte Carlo Simulations (MCS) in Commonwealth Bank Australia are written in Fortran and C#. Such simulations take several hours or over a day (Chang, Wills, & De Roure, 2010c). The results may be needed by the bank for the quarterly reporting period.

Monte Carlo Methods (MCM) are suitable to calculate best prices for buy and sell, and provides data for investors' decision-making (Waters, 2008). MATLAB is used due to its ease of use with relatively good speed. While the volatility is known and provided, prices for buy and sale can be calculated. Chang, Wills, and De Roure (2010b, 2010c) have demonstrated their examples on how to calculate both call and put prices, with their respective likely price, upper limit and lower limit.

3.1. Motivation for Using the Least Square Method

As discussed in Section 2.3, Variance-Gamma Processes (VGP) with Financial Clouds and FSaaS with error reductions are demonstrated by Chang, Wills, and De Roure (2010a, 2010b, 2010c). It has two drawbacks: (1) the program focuses on error correction, which takes time, and seems to make the program slow to start; and (2) 20,000 simulations per attempt is the optimum. This is perhaps because of the high amount of memory required for VGP. Improvements are necessary, including the use of another

Table 1. The first part of coding algorithm for LSM

```
S=100; %underlying price
X=100; %strike
T=1; %maturity
r=0.04; %risk free rate
dividend=0;
v=0.2; % volatility
nsimulations=10000; % No of simulations, which can be updated
nsteps=10; % 10 steps are taken. Can be changed to 50, 100, 150 and 200 steps.
CallPutFlag=”p”;
%%%%%%%%%%%%%%%%%%%%%%%%%%
%AnalyAmerPrice=BjerkPrice(CallPutFlag,S,X,r,dividend,v,T)
r=r-dividend; %risk free rate is unchanged
%AnalyEquropeanPrice=BlackScholesPrice(CallPutFlag,S,X,T,r,v)
if CallPutFlag==”c”,
        z=1;
else
        z=-1;
end;
```

HPC language or a better method. Adopting a better methodology not only enhances performance but also resolves some aspects of challenges. Barnard et al. (2003) demonstrate that having the right method is more important than using a particular language.

The Least Square Methods (LSM) fits into the improvement plan with the following rationale. Firstly, LSM provides a quick execution time, more than 50% compared with VGP (as shown in Section 5). Secondly, it allows the number of simulations to be pushed to 100,000 in one go, before encountering issues such as stability and performance. By offering these two distinct advantages over VGP, LSM is therefore a more suitable method for FSaaS to achieve speed, accuracy and performance. In addition, LSM has been extensively used in robots, or intelligent systems where a major problem is divided into sections, and each section is performed with fast and accurate calculations.

3.2. Coding Algorithm for the Least Square Method

This section describes the coding algorithm for the Least Square Method. Table 1 shows the initial part of the code, where key figures such as maturity, volatility and risk free rate are given. This allows us to calculate and track call prices if variations for maturity, risk free rate and volatility change. Similarly, we can modify our code to track volatility for risk analysis if other variables are changed.

Both American price and European price methods are commonly used in Monte Carlo Simulations (Hull, 2009). It is an added value to calculate both prices in one go, and so both options are included in our code.

The next step involves defining the three important variables for both American and European options, which include cash flow from continuation (CC), cash flow from exercise (CE) and exercise flag (EF), shown in Table 2. The ‘for’ loop is to start the LSM process. Table 3 shows how the three variables CC, CE and EF are updated.

Table 4 shows the main body of LSM calculations. The ‘regrmat’ function is used to perform regression of continuation value. This value is calculated, and fed into the ‘ols’ function, which is a built-in function offered by open-source Octave to calculate ordinary LSM estimation. The p value is the outcome of the ‘ols’ function, which is then used to determine final values of CC, EF and CE. In MATLAB, the equivalent function is ‘lscov’ for the LSM.

Table 5 shows the last part of the algorithm for the LSM. EF, calculated in Table 4 is used

Table 2. The second part of coding algorithm for the LSM

```
smat=zeros(nsimulations,nsteps);
CC=zeros(nsimulations,nsteps); %cash flow from continuation
CE=zeros(nsimulations,nsteps); %cash flow from exercise
EF=zeros(nsimulations,nsteps); %Exercise flag
dt=T/(nsteps-1);
smat(:,1)=S;

drift=(r-v^2/2)*dt;
qrdt=v*dt^0.5;
for i=1:nsimulations,
    st=S;
    curtime=0;
    for k=2:nsteps,
        curtime=curtime+dt;
        st=st*exp(drift+qrdt*randn);
        smat(i,k)=st;
    end
end
```

Table 3. The third part of coding algorithm for the LSM

```
CC=smat*0; %cash flow from continuation
CE=smat*0; %cash flow from continuation
EF=smat*0; %Exercise flag
st=smat(:,nsteps);
CE(:,nsteps)=max(z*(st-X),0);
CC(:,nsteps)=CE(:,nsteps);
EF(:,nsteps)=(CE(:,nsteps)>0);

paramat=zeros(3,nsteps); %coefficient of basis functions
```

to decide values of an important variable 'pay-off_sum', which is then used to calculate the best price for American and European options.

Upon running the MATLAB application, 'lsm', it calculates the best pricing values for American and European options. The following shows the outcome of executing LSM code.

```
> lsm
MCAmericanPrice = 6.3168
MCEuropeanPrice = 5.9421
```

4. A PARTICULAR FSAAS: THE 3D BLACK SCHOLES MODEL BY MATHEMATICA

Black Scholes Model (BSM) has been extensively used in financial modelling and optimisation. Chang, Wills and De Roure (2010a, 2010c) have demonstrated their Black Scholes MATLAB applications running on Clouds for risk analysis. Often risk analysis is presented in visualisation, so that it makes analysis easier to read and understand. MATLAB is useful for calculation and 3D computation, but its 3D computational performance tends to be more time-consuming than Mathematica, which offers commands to compute 3D diagrams swiftly. For this reason, Mathematica is used as the platform for demonstration.

Miller (2009) explain how Mathematica can be used for BSM, and he demonstrates that it is relatively complex to model BSM, so the Black Scholes formulas (BSF) are therefore the best to be expressed in terms of auxiliary function. His rationale is that BSM is based on an

Table 4. The fourth part of coding algorithm for the LSM

```
for k=nsteps-1:-1:2,
    st=smat(:,k);
    CE(:,k)=max(z*(st-X),0);

     %Only the positive payoff points are input for regression
     idx=find(CE(:,k)>0);
     Xvec=smat(idx,k);
     Yvec=CC(idx,k+1)*exp(-r*dt);
     % Use regression - Regress discounted continuation value at the
     % next time step to S variables at current time step
     regrmat=[ones(size(Xvec,1),1),Xvec,Xvec.^2];

     p=ols(Yvec,regrmat); %p = lscov(Yvec, regrmat) for MATLAB CC(idx,k)=p(1)+p(2)*Xvec+p(3)*Xvec.^2;
     %If exercise value is more than continuation value, then
     %choose to exercise
    EF(idx,k)=CE(idx,k) > CC(idx,k);
     EF(find(EF(:,k)),k+1:nsteps)=0;
     paramat(:,k)=p;
     idx=find(EF(:,k) == 0);
     %No need to store regressed value of CC for next use
     CC(idx,k)=CC(idx,k+1)*exp(-r*dt);
     idx=find(EF(:,k) == 1);
     CC(idx,k)=CE(idx,k);
end
```

Table 5. The fifth part of coding algorithm for the LSM

```
payoff_sum=0;
for i=1:nsteps,
     idx=find(EF(:,i) == 1);
     st=smat(idx,i);
     payoffvec=exp(-r*(i-1)*dt)*max(z*(st-X),0);
     payoff_sum=payoff_sum+sum(payoffvec);
end

MCAmericanPrice=payoff_sum/nsimulations

st=smat(:,nsteps);
payoffvec=exp(-r*(nsteps-1)*dt)*max(z*(st-X),0);
payoff_sum=sum(payoffvec);
MCEurpeanPrice=payoff_sum/nsimulations
```

arbitrage argument in which any risk premium above the risk-free rate is cancelled out. Hence, both BSF and auxiliary functions take the same five variables as follows.

p = current price of the stock.
k = exercise price of the option.
sd = volatility of the stock (standard deviation of annual rate of return)
r = continuously compounded risk-free rate of return, e.g., the return on U.S. Treasury bills with very short maturities.
t = time (in years) until the expiration date

The first step is to define the auxiliary function, 'AuxBS', which is then used to define Black Scholes function. The code algorithm and formals are presented as follows:

AuxBS[p_,k_,sd_,r_,t_] = (Log[p/k]+r t)/(sd Sqrt[t])+.5 sd Sqrt[t]

This is equivalent to 0.5 sd $\sqrt{t}$ +(r t+Log[p/k])/(sd $\sqrt{t}$) (9)

Similarly, Black Scholes can be defined as:

BlackScholes[p_,k_,sd_,r_,t_] =
p Norm[AuxBS[p,k,sd,r,t]]- k Exp[-r t]
(Norm[AuxBS[p,k,sd,r,t]-sd Sqrt[t]])

The formula is: -e^{-rt} k Norm[-0.5 sd

$\sqrt{t}$ +(rt+Log[p/k])/(sd $\sqrt{t}$)]+p Norm[0.5 sd $\sqrt{t}$ +(r t+Log[p/k])/(sd $\sqrt{t}$)] (10)

'Norm' is a function in Mathematica to compute complex mathematical modelling such as Gaussian integers, vectors, matrices and so on. By using these two functions effectively, pricing and risks can be calculated and then presented in 3D Visualisation. The advantages are discussed in the next section.

4.1. 3D Black Scholes

Methods such as Fourier series, stochastic volatility and BSM are used for volatility. As a main stream option, BSM is selected for risk analysis in this paper, since BSM has finite difference equations to approximate derivatives. Our previous papers (Chang, Wills, & De Roure, 2010a, 2010c) have demonstrated risk and pricing calculations based on Black Scholes Model (BSM). Results are presented in numerical forms, and occasionally require users and collaborators to visualise some scenarios of numerical computation in their minds. In other papers by Chang, Wills, and De Roure (2010b, 2010c), they demonstrate that Cloud business performance can be presented by 3D Visualisation. Where computational applications can be presented using 3D Visualisation, this can improve usability and understanding (Pajorova & Hluchy, 2010). Currently the focus of MCM is to demonstrate portability on top of computational simulations and modelling in pricing on different Clouds, and this does not need results to be on 3D formats. However, BSM is used to investigate risk. Risk can be difficult to be accurately measured, and models may have possibilities to undermine or miss areas and probability of risk. It is difficult to keep track risks if extreme circumstances happen. The use of 3D Visualisation can help to exploit any hidden errors or missing calculations. Thus, it helps the quality of risk analysis.

4.1.1. Scenarios in Risk Analysis with 3D Visualisation

This section describes some scenarios to calculate and present risks. The first scenario involves investigations of profits/loss in relation to put price. The call price (buying price) for a particular investment is 60 per stock. The put price (selling price) to get zero profit/loss is 60. The risk-free rate, the guarantee rate that will not incur loss, is between 0 and 0.5%. However, the profit and loss will be varied due to impacts of volatility, which means selling price between 50 and 60 will get to a different extent of loss. Similarly, selling prices between 60 and 70 will get to a different extent of profits. The intent is to find out the percentage of profit and loss for massive sale, and the risk associated with it. While using auxiliary and Black Scholes function, the result can be computed in 3D swiftly and presented in Figure 1, which is similar to a 3D parabola.

The second scenario is to identify the best put price for a range of fluctuating volatilities. Volatility is used to quantify the risk of the financial instrument, and is subject to fluctuation that may result in different put prices. The volatility ranges between 0.20% and 0.40%, the best put price is between 6.5 and 9.2, and the risk-free rate is between 0 and 0.5%. The higher the risk is, the more the return will be. However, this situation is reversed when risk (volatility in this case) goes beyond cut-off volatility. Hence, the task is to keep track the risk pattern, and to identify the cut-off point for volatility. Similarly, auxiliary and Black Scho-

Figure 1. The 3D risk analysis to investigate volatile percentage of profits and loss

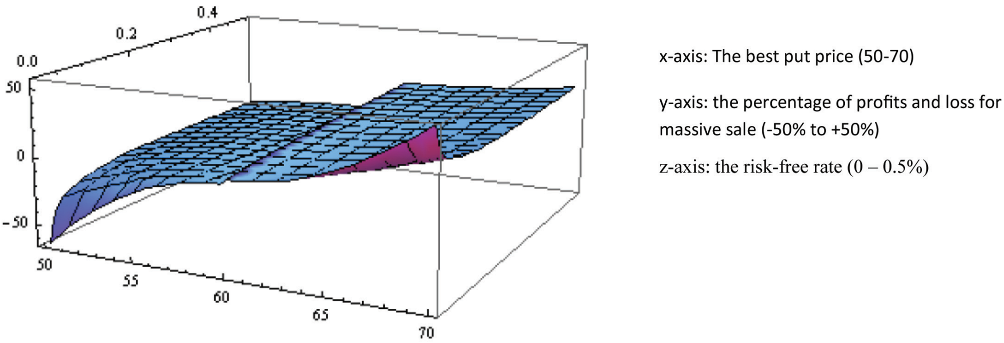
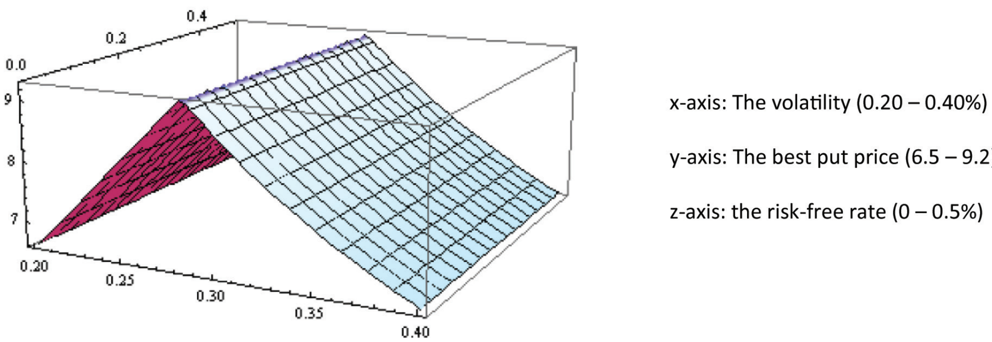


Figure 2. The 3D risk analysis to investigate the best put price in relations to fluctuating volatility



les functions are used to compute 3D Visualisation swiftly, and result is presented in Figure 2 which looks like an inverted-V and shows the best price is 9 while volatility is 0.30.

4.1.2. Delta and Theta: Scenarios in Risk Analysis with 3D Visualisation

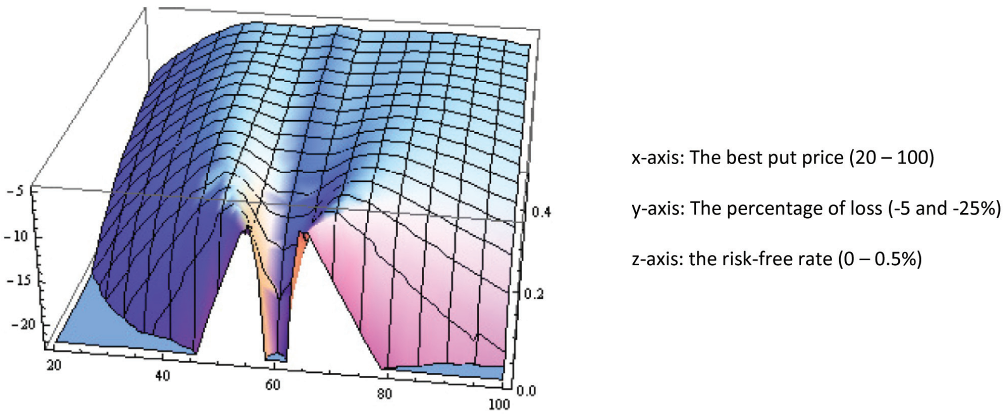
In BSM, the partial derivative of an option value with respect to stock price is known as Delta. Hull (2009) and Millers (2009) assert that Delta is useful in risk measurement for an option because it indicates how much the price of an option will respond to a change of price of the stock. Delta is a useful tool in risk management where a portfolio contains more than one option of the stock. The derivative function, D, is built in Mathematica. This much simplifies coding for Delta, which can be presented as

Delta[p_,k_,sd_,r_,t_] = D[BlackScholes[p,k,sd,r,t],p] which corresponds to this formula

$$(0.398942\ \frac{1}{2}(0.5sd\sqrt{t}\ \frac{rt\ \ Log-\frac{p}{k}-}{sd\sqrt{t}})^2)/(sd\ \sqrt{t})\text{-}(0.398942\ rt\frac{1}{2}(0.5sd\sqrt{t}\ \frac{rt\ \ Log-\frac{p}{k}-}{sd\sqrt{t}})^2\ k)/(p\ sd\ \sqrt{t})$$
$$+Norm[0.5\ sd\ \sqrt{t}+(r\ t+Log[p/k])/(sd\ \sqrt{t})] \quad (11)$$

Delta computes positive derivatives in BSM, and to get an inverted Delta, a new function, Theta, is introduced.

Figure 3. The 3D risk analysis to explore the percentage of loss and the best put price in relations to the impact of economic downturn



Theta[p_,k_,sd_,r_,t_] = -D[BlackScholes[p,k,sd,r,t],t] which corresponds to this formula

$$0.398942\ rt\frac{1}{2}\left(0.5sd\sqrt{t}\,\frac{rt\ Log-\frac{p}{k}-}{sd\sqrt{t}}\right)^2 \text{ k (r/(sd } \sqrt{t}\text{)-(0.25 sd)/} \sqrt{t} \text{ -(r t+Log[p/k])/(2 sd } t^{3/2}\text{))-0.398942 } \frac{1}{2}\left(0.5sd\sqrt{t}\,\frac{rt\ Log-\frac{p}{k}-}{sd\sqrt{t}}\right)^2 \text{ p (r/(sd } \sqrt{t}\text{)+(0.25 sd)/} \sqrt{t} \text{ -(r t+Log[p/k])/(2 sd } t^{3/2}\text{))- } e^{-rt} \text{ k r Norm[-0.5 sd } \sqrt{t} \text{ +(r t+Log[p/k])/(sd } \sqrt{t}\text{)]} \quad (12)$$

The third scenario is to investigate the extent of loss in an organisation during the financial crisis between 2008 and 2009, and to identify which put prices (in relations to volatility) will get the least extent of loss while keeping track of risks in 3D. This needs using Theta function to present the risk and pricing in relations high volatility. The put price is between 20 and 100, and the percentage of loss is between -5% and -25%, and the risk-free rate is 0 and 0.5%. In this case, risk-free rate means the percentage this organisation can get assistance from. The Theta function is used to compute the 3D risk swiftly and to get the result in Figure 3. This shows the percentage of loss gets better when the put prices are raised to approximately 55. However, when it gets to 60, this is the price that uncontrolled volatility (such as human speculation or natural disasters) takes hold and the percentage of loss goes down sharply at -25%. The percentage of loss is raised to -5%, and is slowly lowering its value to-25%. However, if the risk-free rate is improved up to 0.5%, the extent of loss is less, and stays nearly at -5%. It means credit guarantee from somewhere may help this organisation with the minimum impacts from loss. However, this is just a computer simulation and does not reflect the real difficulty faced by this organisation. Even so, our FSaaS simulations can produce a range of likely outcomes, which are valuable to decision-makers.

5. EXPERIMENT AND BENCHMARK IN THE CLOUDS

Monte Carlo Simulations with LSM can be used for FSaaS on Public, Private and Hybrid Clouds. This is further enhanced by the use of open source package, Octave 3.2.4, so that there is no need to write additional APIs to achieve enterprise portability. Applications written on the developer platform can be portable and executable on different desktops and Clouds of different hardware and software requirements, and execute as if they are on the same platform.

3D Black Scholes has fast execution time and only runs in Mathematica, which is not yet portable to different Clouds due to licensing issues and also there is no open source alternative to simplify the process of enterprise portability. At the time of writing, MATLAB licences on Private Clouds are still under development, and therefore results on MATLAB have only Private Cloud in Virtual Machines. Chang, Wills and De Roure (2010a, 2010c) have demonstrated the same FSaaS application running with Octave and MATLAB on different Clouds, and their results demonstrate that the execution speed on MATLAB is approximately five times quicker than Octave, though MATLAB is more expensive and needs to deal with licensing issues regularly.

5.1. Experiments with Octave in Running the LSM on Different Clouds

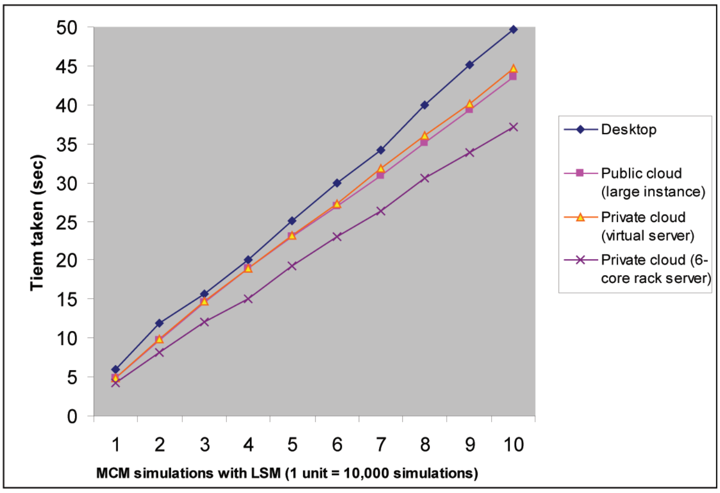
Code written for LSM in Section 3.2 has been used for experimenting and benchmarking in the Clouds. 10,000 to 100,000 simulations (increase with an additional 10,000 simulations each time) of Monte Carlo Methods (MCM) adopting LSM are performed and the time taken at each of a desktop, private clouds and Amazon EC2 public clouds are recorded and averaged with three attempts. Hardware specifications for desktop, public cloud and private clouds are described as follows.

The desktop has 2.67 GHz Intel Xeon Quad Core and 4 GB of memory (800 MHz) with installed. One Amazon EC2 public cloud is used. The first virtual server is a 64-bit Ubuntu 8.04 with large resource instance of dual core CPU, with 2.33 GHz speed and 7.5GB of memory. There are two private clouds set up. The first private cloud is hosted on a Windows virtual server, which is created by a VMware Server on top of a rack server, and its network is in a network translated and secure domain. The virtual server has 2 cores of 2.67 GHz and 4GB of memory at 800 MHz. The second private cloud is a 64-bit Windows server installed on a rack, with 2.8GHz Six Core Opteron, 16 GB of memory. All these five settings have installed Octave 3.2.4, an open source compiler equivalent to MATLAB. The experiment began by running the FSaaS code (in Section 3.2) on desktop, private clouds and public cloud and started one at a time. Three attempts for each set of simulations are done, and the result is the average of three attempts. Benchmark is execution time, since it is a common benchmark used in several financial applications. Figure 4 shows the complete result of running FSaaS code on different Clouds.

Figure 4 shows the execution time for FSaaS application on desktop, public cloud and two private clouds. Experiments confirm with the followings. Firstly, enterprise portability is achieved and the FSaaS application can be executed on different platforms. Secondly, the improved FSaaS application can go for 100,000 simulations in one go on Clouds. Although above 100,000 simulations in one go, factors such as performance and stability must be balanced, before tuning up the capabilities of our FSaaS. The six-core processing rack server has the most advanced CPU, disk, memory, 64-bit operating system and networking hardware, and is not surprising that it is always the quickest. Although the desktop has similar hardware specification to server, it comes out slowest in all experiments. The difference between the Public Cloud (large instance) and Private Cloud (virtual server) is minimal. Although the large instance of a public cloud has the edge on hardware specification against the Virtual Private Cloud (VPC), the networking speed within the VPC is faster than the Public Cloud, and this explains the small differences between them.

Benchmark results show pricing and risk analysis can be calculated rapidly with accurate outcomes. Portability is achieved with a good reliable performance in clouds. These experiments demonstrate portability, speed, accuracy

Figure 4. Timing benchmark comparison for desktop, public cloud and two private clouds for time steps = 10



and reliability from desktop to clouds. Figure 4 shows the benchmark graph.

5.2. Experiments with MATLAB in Running the LSM on Desktop and one Private Cloud

MATLAB is used for high performance Cloud computation, since it allows faster calculations than Octave. The drawback of using MATLAB 2009 is license, which means all desktops and Cloud resources must be licensed prior setting up experiments. For this reason, only desktop and a Private Cloud (virtual machine) are used for experiments. The use of MATLAB 2009 reduces execution time for FSaaS, and also allows experiments to proceed with a higher number of time steps. The more the time steps used, the more accurate the outcome is, although higher numbers of time steps need more computing resources.

Five different sets of experiments are designed, and each set of experiments count execution time from 10,000 to 100,000 simulations as described in Section 5.2. The only difference is time step. The first experiment gets time step equals to 10, and second experiment has time step equals to 50, and the third experiment sets time steps equal to 100, and the fourth experiment has time steps equal to 150, and finally, the last experiment gets time step equal to 200. The time step can be increased up to 1,000, but performance seems to drop off, particularly for experiments running high numbers of simulations. For this reason, the maximum time step in the experiments is limited to 200. Results for each set of experiments are recorded and shown in Figure 5, 6, 7, 8 and 9.

The results presented in Figures 5, 6, 7, 8 and 9 have the following implications. Firstly, the execution time and number of simulations are directly proportional to each other. It means the higher the number of simulations to be computed, the longer the execution time on desktop and Private Cloud. It is not so obvious to identify linear relationship with a lower time step involved. This is likely because that execution time is so quick to complete that the range of errors and uncertainties are higher. When the time steps increases, it is easier to identify the linear relationship. This linear relationship also confirms what LSM suggests and recommends. Secondly, MATLAB 2009 offers quick execution time for the portability to Cloud, and the significant time reduction is experienced. However, the licensing issue still prevents from

Figure 5. MATLAB timing benchmark for time step = 10

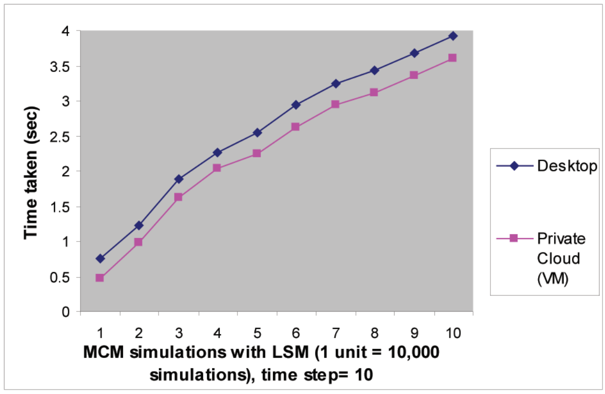


Figure 6. MATLAB timing benchmark for time step = 50

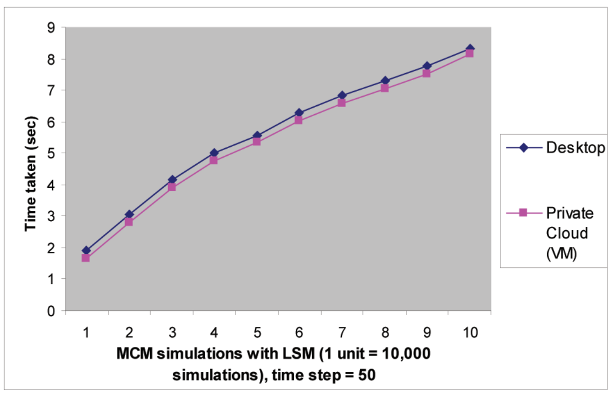


Figure 7. MATLAB timing benchmark for time step = 100

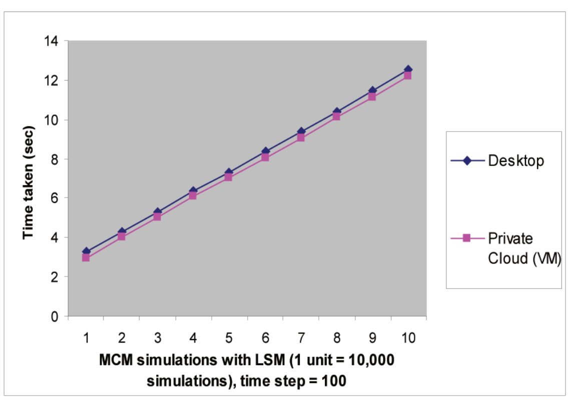


Figure 8. MATLAB timing benchmark for time step = 150

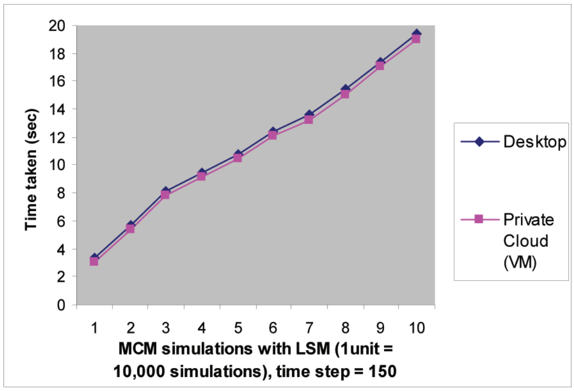
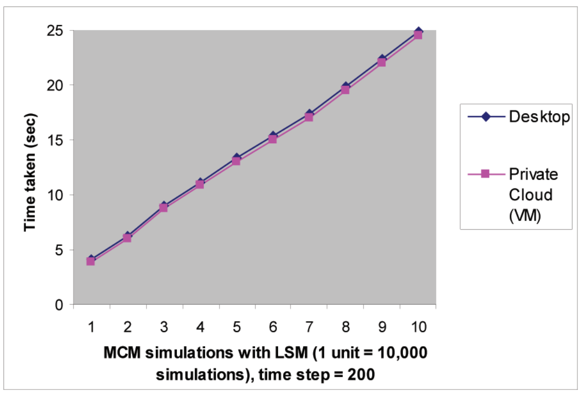


Figure 9. MATLAB timing benchmark for time step = 200



a large scale of adoptions to different Clouds. This means portability should be made as easy as possible, and not only includes technical implementations but also licensing issues. However, this paper will not go into details about licensing.

6. A CONCEPTUAL CLOUD PLATFORM: IMPLEMENTATIONS AND WORK-IN-PROGRESS

As discussed in previous sections, the primary objective for optimal provisioning and runtime management of cloud infrastructures at the infrastructure, platform, and software as a service levels is to optimise the delivery of the overall business outcome of the user. Improved business outcome in general refers to the increased revenue or reduced cost or both. Uncertainties of outcome, measured in terms of variance, are often regarded as negative impacts (or *risk*) and must be accounted for in the pricing calculations of the service delivery.

There are many types of risks that might impact the variance of the business outcome – including market risk, credit risk, liquidity risk, legal/reputation risk and operational risk. (Risk taxonomy was previously established in the context of various banking regulations such as Basil II.) Among these types of risks, operational risk is considered to be most directly related to the IT infrastructure as it might impact the business through internal and external fraud, workplace safety, business practice, damage to physical assets, business disruption and system failures, and execution delivery and process management. In particular, over and under capacity, general availability of the system, failed transactions, loss of data due to virus or intrusion, poor business decision due to poor data, and failure of communication systems are all considered as part of the business disruption and system failures and need to be considered as part of the operational risk.

Behaviour models of systems are often constructed to predict the likely outcome under different context and scenarios. Both analytical and simulations methodologies have been applied to these behaviour models to predict the likely outcomes, and our demonstrations in MCM and BSM present some of these predictability features. Maximize the outcome requires minimizing the risk, cost, and maximize the performance.

In regard to all possible causes, "Poor business decision due to poor data quality" is the one that we address. The proposal of FSaaS can track and display risks in 3D Visualisation, so that there is no hidden area or missing data not covered within simulations. Accurate results can be computed quickly for 100,000 simulations in one go, and this greatly helps directors to make the right business decisions.

Apart from MCM and BSM simulations, other technologies such as workflows are used to present risks in business processes and help making the right business decision. This includes Risk Tolerance, which is commonly associated with the industry framework and business processes and have to be established top down. Figure 10 shows a business process-based behaviour model of a typical e-commerce operation. The customer interacts with the web site through web server for placing a new order or initiates a return/exchange. Either of the two scenarios will require interaction with the customer order system and accessing the customer records. A new order might also involve preparing the billing, sending the request to the warehouse for fulfillment. This business process based behaviour model clearly illustrates different types of operational risk involved during various stage of the business process. In Figure 10, the types of operational risk identified from the front end part of the business process includes Business Reputation Natural Disaster, System failure/System Capacity Security, and other system failures and security issues. Business Risk includes Business Reputation and other system failures and security issues.

6.1. Contributions from Southampton: The Financial Software as a Service (FSaaS)

Figure 11 shows a conceptual architecture based on Operational Risk Exchange (www.orx.org), which currently includes 53 banks from 18 countries for sharing the operational risk data (a total of 177K loss incidents with a total of 62B Euros of loss as of the end of 2010), and demonstrated how financial clouds could be implemented successfully for aggregating and sharing operational risk data. One of the main contributions from the University of Southampton is the use of MCM (MATLAB) for pricing and BSM (Mathematica) for risk analysis. This cloud platform offers calculation for risk modelling, fraud detection, pricing analysis and a critical analysis with warning over risk-taking. It reports back to participating banks and bankers about their calculations, and provides useful feedback for their potential investment.

Risk data computed by different models such as MCM, BSM and other models can be simulated and shared within the secure platform that offers anonymisation and data encryption. It also allows bank clients to double check with mortgage lending interests and calculations whether they are fit for purpose. This platform also works closely with regulations and risk

Figure 10. The operational risk and business risk analysis by workflow

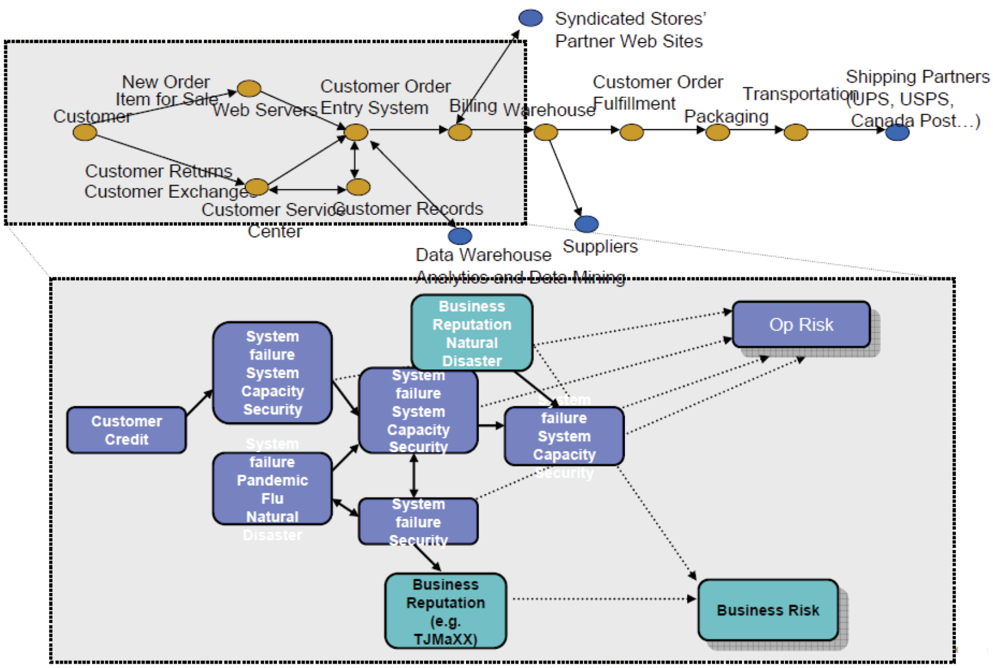


Figure 11. A conceptual financial cloud platform [using orx.org as an example] and contributions from Southampton in relations to this platform

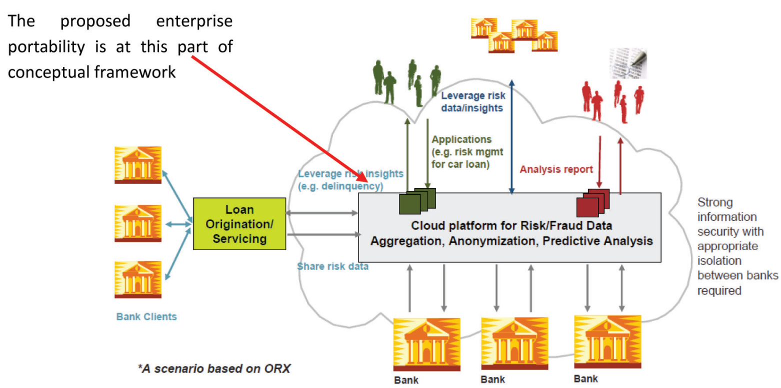
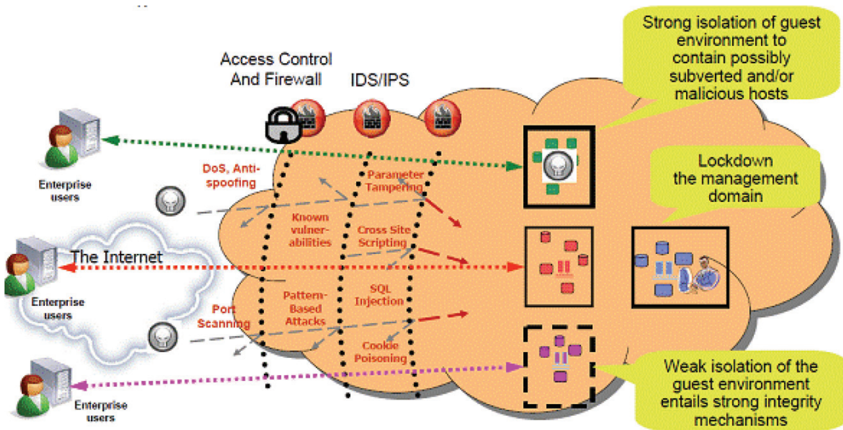


Figure 12. The IBM Fine-Grained Security Framework (Li, 2010)



control, thus risks are managed and monitored in the Financial Cloud platform. Our FSaaS is one part of the platform (as shown in the red arrow) to demonstrate accuracy, performance and enterprise portability over Clouds, and is not only conceptual but has been implemented.

6.2. The IBM Fined Grained Security Framework

Figure 12 shows the Fined Grained Security Framework currently being developed at IBM Research Division. The framework consists of layers of security technologies to consolidate security infrastructure used by financial services. In additional to the traditional perimeter defence mechanisms such as access control, intrusion detection (IDS) and intrusion prevention (IPS), this fine-grained security framework introduced fine-grained perimeter defence at a much finer granularity such as a virtual machine, a database, a JVM, or a web service container.

Starting with the more traditional approach side, the first layer of defence is Access Control and firewalls, which only allow restricted members to access. The second layer consists of Intrusion Detection System (IDS) and Prevent System (IPS), which detect attack, intrusion and penetration, and also provide up-to-date technologies to prevent attack such as DoS, anti-spoofing, port scanning, known vulnerabilities, pattern-based attacks, parameter tampering, cross site scripting, SQL injection and cookie poisoning.

The novel approach in the proposed fine-grained approach imposes the additional protection in terms of isolation management – which enforces top down policy based security management; integrity management – which monitors and provides early warning as soon as the behaviour of the fine-grained entity starts to behave abnormally; and end-to-end continuous assurance which includes the investigation and remediation after abnormally is detected. This environment intends to provide strong isolation of guest environment in an infrastructure or platform as a service environment and to contain possibly subverted and malicious hosts for security. Weak isolation can also be provided when multiple guest environments need to collaborate and work closely – such as in a three tier architecture among web server, application server, and database environment. Weak isolation usually focuses more on monitoring and captures end-to-end provenance so that investigation and remediation can be greatly facilitated. Strong isolation and integrity management is also required for the cloud management infrastructure – as this is often among the first few vulnerabilities of the cloud are exposed. See Figure 12 for details.

7. DISCUSSIONS

7.1. Variance in Volatility, Maturity and Risk Free Rate

Calculating the impacts of volatility, maturity and risk free rate is helpful to risk management. Our code in Section 3.2 can calculate these three aspects with these observations. Firstly, the higher the volatility is, the lower the call price, so that risk can be minimised. Secondly, the more the maturity becomes, the higher the call price, which improves higher returns of assets before the end of life in a bond or a security. Thirdly, the higher the risk free rate, the higher the call price, as high risk free rate has reduced risk and boosts on investors' confidence level. Both Monte Carlo Methods and Black Scholes models are able to calculate these three aspects.

7.2. Accuracy

Monte Carlo Simulations are suitable to analyse pricing and provide reliable calculations up to several decimal numbers. In addition, the use of LSM reduces errors and thus improves the quality of calculation. New and existing ways to improve error corrections are under further investigation while achieving enterprise SaaS portability onto Clouds. In addition, the use of 3D Black Scholes will ensure the accuracy and quality of risk analysis. Risks can be quantified and also presented in 3D Visualisation, so that risks can be tracked and checked with the ease.

7.3. Implications for Banking

There are implications for banking. Firstly, security is a main concern. This is in particular when Cloud vendors tend to mitigate this risk technically by segregating different parts of the Clouds but still need to convince clients about the locality of their data, and data protection and security. Security concerns for banks in terms of using Cloud Computing, may be limited to cases where data need to be transferred (even for a moment) to the cloud infrastructure. However, certain risk management simulations, such as those involving Monte Carlo, where input data are usually random data based on statistical distribution (instead of using real client data), then these computations can be performed on the cloud without security concerns.

Secondly, financial regulators are imposing tighter risk management controls. Thus, financial institutions are involved in running more analytical simulations to calculate risks to the client organisations. This may present a greater need for the use of the Cloud computation and resources. Thirdly, portability of the Cloud can imply letting clients install their own libraries. Users who run MATLAB on the Cloud may only need the MATLAB application script or executable and to install the MATLAB Runtime once on the Cloud. For financial simulations written in Fortran or C++, users may also need Mathematical libraries to be installed in the Cloud. The Cloud must facilitate an easy way to install and configure user required libraries, without the need to write additional APIs like several practices do.

Portability would be important since bank personnel who run the simulations, should be able to install the necessary software infrastructure such as 'dlls'. One key benefit offered by Cloud is the cost. In Risk Management where mathematical models are always changing and becoming more advanced, the hardware requirement changes with it. Using the cloud service such as FSaaS would reduce upgrade costs. Hence greater hardware requirement may be facilitated by upgrading cloud subscription to a higher level, instead of decommissioning the company's own servers and replaced by new ones.

7.4. Enterprise Portability to the Clouds

Enterprise portability involves moving entire application services from desktops to clouds and between different Clouds, so that users need not worry about complexity and work as if on their familiar systems. This paper demonstrates financial clouds that modelling and simulations can take place on the Clouds, where users can connect and compute. This has the following advantages:

- Performance and speed: Calculations can be completed in a short time.
- Accuracy: The improved models based on LSM provide a more accurate range of prices comparing to traditional computation in normal distribution.
- Usability: users need not worry about complexity. This includes using iPhone or other user-friendly resources to compute. However, this is not the focus of this paper.

However, the drawback for portability is that additional APIs need to be written (Chang, Wills, & De Roure, 2010c). Clouds must facilitate an easy way to install and configure user required libraries, without the need to write additional APIs like several practices do. If writing APIs is required for portability, an alternative is to make APIs as easy and user-friendly as Facebook and iPhone do. In our demonstration, there is no need to write additional APIs to execute financial clouds.

7.5. Other Alternatives Such as Parallel Computing

In parallel computing, one way to speed up is to divide the data up into chunks and compute on different machines. However, there is an overhead in designing the problem (requiring human design effort) also there is machine overhead in sending the chunks of data to different machines and having a host machine to keep track of it. In a cloud, this may involve sending to different parts of the cloud and depending on how busy the cloud is; perhaps it will take longer in waiting time than when it actually takes to compute the chunk of data.

MCM is used for simulating losses due to Operational Risks, and there are plans in the Commonwealth Bank, Australia, to perform experiments in parallel computing with virtual machines, which have been recently set up.

8. CONCLUSION AND FUTURE WORK

FSaaS including MCM and BSM are used to demonstrate how portability, speed, accuracy and reliability can be achieved while demonstrating financial enterprise portability on different Clouds. This fits into the third objective in the CCBF to allow portability on top of, secure, fast, accurate and reliable clouds. Financial SaaS provides a useful example to provide pricing and risk analysis while maintaining a high level of reliability and security. Our research purpose is to port and test financial applications to run on the Clouds, and ensure enterprise level of portability is workable, thus users can work on Clouds as they work on their desktops or familiar environments. Six areas of discussions are presented to support our cases and demonstration.

Benchmark is regarded as time execution to complete calculations after portability is achieved. Timing is essential since less time with accuracy is expected in using Financial SaaS on Clouds. The LSM provides added values and improvements. Firstly, it has a short starting and execution time to complete pricing calculations, and secondly, it allows 100,000 simulations in one go in different Clouds. This confirms enterprise portability can be delivered with LSM application on Octave 3.2.4 and MATLAB 2009. Five sets of experiments with MATLAB in running the LSM are performed, where the time steps have been increased for each set. The results confirm the linear relationship and also fast execution time for up to 100,000 simulations in one go on the Private Cloud. Portability should be made as easy as possible, and not only includes technical implementations but also licensing issues. In addition, the 3D Black Scholes presentation can enhance the quality of risk analysis, since risks are not easy to track down in real-time. The 3D Black Scholes improve the risk analysis so that risks can be presented in BSM formulas and are easier to be checked and understood. Three different scenarios of risk analysis are illustrated, and 3D simulations can provide a range of likely outcomes, so that decision makers can avoid potential pitfalls.

Implementations and work-in-progress for a conceptual Cloud Platform have been demonstrated. This includes the use of workflow to present risks in business processes, including the operational risk and business risk, so that risk tolerance can be established and the analysis can help making the right decision. Contributions from Southampton are the implementation of FSaaS, which allow pricing calculations and risk modelling to be computed fast and accurately to meet the research and business demands. Technical implementation in enterprise portability also meets challenges in business context: reduce time and cost with better performance. The IBM Fined Grained Security Framework provides a comprehensive model to consolidate security, which impose the additional protection in terms of isolation management and integrity management. This ensures trading, transaction and any financial related activities on Clouds are further protected and safeguarded.

Future work may include the following. HPC languages such as Visual C++ and/or .NET Framework 3.5 (or 4.0) will be used for the next stages. Other methods such as parallelism in MCM are potentially possible for further investigations. New error correction methods related to MCM will be investigated, and any useful outcomes will be discussed in the future work. New techniques to improve current 3D Black Scholes Visualisation will be investigated. There are plans to investigate Financial SaaS and its enterprise portability over clouds with Commonwealth Bank Australia, IBM US and other institutions, so that better platforms, solutions and techniques may be demonstrated. We hope to present different perspectives, recommendations and solutions for risk analysis, pricing calculations, security and financial modelling on Clouds, and to deliver improved prototypes, proof of concepts, advanced simulations and visualisation.

ACKNOWLEGMENT

We greatly thank Howard Lee, a Researcher from Deakin University in Australia, for his effort to inspect our code and improve overall quality.

REFERENCES

Abdi, H. (2009). *The methods of least squares*. Dallas, TX: The University of Texas.

Assuncao, M. D., Costanzo, A., & Buyya, R. (2010). A cost-benefit analysis of using cloud computing to extend the capacity of clusters. *Journal of Cluster Computing*, *13*, 335–347. doi:10.1007/s10586-010-0131-x

Barnard, K., Duygulu, P., Forsyth, D., De Freitas, N., Blei, D. M., & Jordan, M. I. (2003). Matching words and pictures. *Journal of Machine Learning Research*, 1107–1135. doi:10.1162/153244303322533214

Beaty, K., Kochut, A., & Shaikh, H. (2009, May 23-29). Desktop to cloud transformation planning. In *Proceedings of the IEEE International Symposium on Parallel and Distributed Processing*, Rome, Italy (pp. 1-8).

Birge, L., & Massart, P. (2001). Gaussian model selection. *Journal of the European Mathematical Society*, *3*(3), 203–268. doi:10.1007/s100970100031

Brandic, I., Music, D., Leitner, P., & Dustdar, S. (2009, August 25-28). VieSLAF framework: Enabling adaptive and versatile SLA-management. In *Proceedings of the 6th International Workshop on Grid Economics and Business Models*, Delft, The Netherlands.

Briscoe, G., & Marinos, A. (2009, June 1-3). Digital ecosystems in the clouds: Towards community cloud computing. In *Proceedings of the 3rd IEEE International Conference on Digital Ecosystems and Technologies*, New York, NY (pp. 103-108).

Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Journal of Future Generation Computer Systems*, *25*(6), 559–616. doi:10.1016/j.future.2008.12.001

Chang, V., Mills, H., & Newhouse, S. (2007, September). *From open source to long-term sustainability: Review of business models and case studies*. Paper presented at the UK e-Science All Hands Meeting, Nottingham, UK.

Chang, V., Wills, G., & De Roure, D. (2010a, July 5-10). A review of cloud business models and sustainability. In *Proceedings of the Third IEEE International Conference on Cloud Computing*, Miami, FL.

Chang, V., Wills, G., & De Roure, D. (2010b). Case studies and sustainability modelling presented by cloud computing business framework. *International Journal of Web Services Research*.

Chang, V., Wills, G., & De Roure, D. (2010c, September 13-16). *Cloud business models and sustainability: Impacts for businesses and e-research*. Paper presented at the UK e-Science All Hands Meeting Software Sustainability Workshop, Cardiff, UK.

Chang, V., Wills, G., De Roure, D., & Chee, C. (2010, September 13-16). *Investigating the cloud computing business framework - modelling and benchmarking of financial assets and job submissions in clouds*. Paper presented at the UK e-Science All Hands Meeting on Research Clouds: Hype or Reality Workshop, Cardiff, UK.

Chou, T. (2009). *Seven clear business models*. Active Book Press.

Choudhury, A. R., King, A., Kumar, S., & Sabharwal, Y. (2008). Optimisations in financial engineering: The least-squares Monte Carlo method of Longstaff and Schwarts. In *Proceedings of the IEEE International Symposium on Parallel and Distributed Computing* (pp. 1-11).

City, A. M. (2010). *Business with personality.* Retrieved from http://www.cityam.com

Feiman, J., & Cearley, D. W. (2009). *Economics of the cloud: Business value assessments*. Stamford, CT: Gartner RAS Core Research.

Financial Times. (2009). *Interview with Lord Turner, Chair of Financial Services Authority*. Retrieved from http://www.ft.com/cms/s/0/d76d0250-9c1f-11dd-a42e-000077b07658.html#axzz1Iqssz7az

Hamnett, C. (2009). *The madness of mortgage lenders: Housing finance and the financial crisis*. London, UK: King's College.

Hull, J. C. (2009). *Options, futures, and other derivatives* (7th ed.). Upper Saddle River, NJ: Pearson/Prentice Hall.

Li, C. S. (2010, July 5-10). Cloud computing in an outcome centric world. In *Proceedings of the IEEE International Conference on Cloud Computing*, Miami, FL.

Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American options by simulations: A simple least-squares approach. *Review of Financial Studies*, *14*(1), 113–147. doi:10.1093/rfs/14.1.113

Millers, R. M. (2011). *Option valuation*. Niskayuna, NY: Miller Risk Advisor.

Moreno, M., & Navas, J. F. (2001). On the robustness of least-square Monte Carlo (LSM) for pricing American derivatives. *Journal of Economic Literature Classification.*

Mullen, K., & Ennis, D. M. (1991). A simple multivariate probabilistic model for preferential and triadic choices. *Journal of Psychometrika*, *56*(1), 69–75. doi:10.1007/BF02294586

Mullen, K., Ennis, D. M., de Doncker, E., & Kapenga, J. A. (1988). Models for the duo-trio and triangular methods. *Journal of Bioethics*, *44*, 1169–1175.

Pajorova, E., & Hluchy, L. (2010, May 5-7). 3D visualization the results of complicated grid and cloud-based applications. In *Proceedings of the 14th International Conference on Intelligent Engineering Systems*, Las Palmas, Spain.

Patterson, D., Armbrust, M., Fox, A., Griffith, R., Jseph, A. D., Katz, R. H., et al. (2009). *Above the clouds: A Berkeley view of cloud computing* (Tech. Rep. No. UCB/EECS-2009-28). Berkeley, CA: University of California.

Piazzesi, M. (2010). *Affine term structure models*. Amsterdam, The Netherlands: Elsevier.

Schubert, L., Jeffery, K., & Neidecker-Lutz, B. (2010). *The future for cloud computing: Opportunities for European cloud computing beyond 2010 (public version 1.0)*. Retrieved from http://cordis.europa.eu/fp7/ict/ssai/docs/cloud-report-final.pdf

Waters, D. (2008). *Quantitative methods for business* (4th ed.). Upper Saddle River, NJ: Prentice Hall.

Weinhardt, C., Anandasivam, A., & Blau, B., & Sto-Ber, J. (2009). Business models in the service world. *IT Professional*, *11*(2). doi:10.1109/MITP.2009.21

Weinhardt, C., Anandasivam, A., Blau, B., Borissov, N., Meinl, T., & Michalk, W. (2009). Cloud computing – a classification, business models, and research directions. *Journal of Business and Information Systems Engineering*, *1*(5), 391–399. doi:10.1007/s12599-009-0071-2