ÉCOLE POLYTECHNIQUE

EA MAP513 4.1 : Wishart Processes and Multidimensional Stochastic Volatility Model

Yiming QIN and Benxin ZHONG

12 décembre 2020

Abstract

The Wishart processes are processes whose values are symmetric positive matrices. These processes are used massively in finance for modeling the instant covariance between different assets. Yet, existing simulation methods for Wishart process are usually limited for certain conditions or in lack of precision. In [1] and [2], methods for exact and approximate simulation of Wishart and CIR process without parameters limits are proposed.

Our work is composed of two parts. In the first stage of our project, we researched and implemented the exact simulation and the discretisation schemes for CIR process as well as Wishart process. We have also given the comparison of executing and converging speed for each method. In the second part, we will use this method to implement some financial applications.

We have generally finished the first part of project till now. The work is mainly based on first two papers given in the reference and our codes can be accessed on MAP513-Processus-Wishart.

TABLE DES MATIÈRES

1 INTRODUCTION

This project focuses on simulation methods of Wishart process.

The most basic form of a Wishart process is defined by $S_t = N^T N$, where N_t is a $n \times p$ Brownian matrix. This process satisfies the following differential equation[6] :

$$dS_t = \sqrt{S_t}dB_t + dB_t^T\sqrt{S_t} + nIdt$$

While here in this paper, we will focuse on a more generalized case on the squares of a Ornstein-Uhlenbeck process X_t satisfying the folloing SDE :

$$dX_t = \lambda dN_t + \beta X_t dt$$

Let $S_t = X_t^T X_t, s_0 = x_0^T x_0$ and replace β and α by $p \times p$ matrices $x = (b_{ij})$ and $a = (a_{ij})$. The SDE of this process is can be written as a special kind of affine process :

$$dX_t = (\bar{\alpha} + B(X_t))dt + (\sqrt{X_t}dW_t a + a^T dW_t^T \sqrt{X_t}), \tag{1.1}$$

$$X_0 = x. \tag{1.2}$$

For Wishart processes, we have

$$\exists \alpha \geq 0, \text{ s.t. } \bar{\alpha} = \alpha a^T a, \text{ and}$$

$$\exists b \in \mathscr{M}_d(\mathbb{R}), \text{ s.t. } B(x) = bx + xb^T.$$

Specifically, when $d = 1$, Wishart process is called the Cox-Ingersoll-Ross (CIR) process. Its general formula is

$$dX_t = (a - kX_t)dt + \sigma\sqrt{X_t}dW_t. \tag{1.3}$$

CIR process has already been broadly studied. The simulation of CIR process plays an important role in the simulation of Wishart process, therefore in this project, we started with the CIR process by presenting a simulation method. We denote by $\text{WIS}_d(x, \alpha, b, a)$ (respectively, $\text{AFF}_d(x, \bar{\alpha}, B, a)$) the law of Wishart (and affine) process $(X_t^x)_{t\geq 0}$.

The simulation stage of this project contains principally two parts : the exact simulation and the discretization schemes. In the first part, we give a method to simulate the exact distribution of CIR processes, based on which we develop the exact simulation of Wishart processes. In the second part, we implement the potential 2nd order and 3rd order discretization schemes for CIR processes and then for Wishart processes.

2 Simulation of CIR Processes

Firstly, we consider the Cox-Ingersoll-Ross (CIR) process, a commonly used model for describing the evolution of interest rates. It describes interest rate movements as driven by only one source of market risk.

Mathematically, a CIR process is an 1-dim Wishart process and its implementation will be necessary for that of a multi-dimensional Wishart process. The general form of the SDE of a CIR process is :

$$X_t^x = x + \int_0^t (a - kX_s^x)ds + \sigma \int_o^t \sqrt{X_s^x} dW_s, \quad x \geq 0, \tag{2.1}$$

where k is non-negative, and a and σ are positive.

2.1 Exact simulation

By defining

$$\nu := \frac{4a}{\sigma^2}$$

and

$$\eta_k(h) := \frac{4k\exp(-kh)}{\sigma^2(1-\exp(-kh))},$$

then for $s < t$,

$$X_t^x | X_s^x \overset{\text{law}}{=} \frac{\exp(-k(t-s))}{\eta(t-s)} K_s^t, \tag{2.2}$$

where $K_s^t \sim \chi_\nu^2\left(\lambda = X_s\eta(t-s)\right)$, the *non-central chi-squared* distribution with degree of freedom ν and non-centrality parameter $\lambda = X_s\eta(t-s)$ [7].

Therefore we can generate an exact simulation of CIR processes by simulating the non-central chi-squared distribution. With methods provided by `numpy`, this exact simulation is very convenient.

2.2 Discretisation schemes

In this project, we studied and implemented the *weak 2nd and 3rd* discretization schemes proposed by Alfonsi[2].

Definition 2.1. *We denote by $\mathscr{C}_K^\infty(\mathbb{D},\mathbb{R})$ the set of $\mathscr{C}^\infty$ real-valued functions with a compact support in $\mathbb{D}$. Let $x \in \mathbb{D}$, a discretization scheme $(\hat{X}_{t_i^n}^n)_{0\leq i\leq n}$ for the SED $(X_t^x)_{0\leq t\leq T}$ is said to be a weak r th-order scheme if :*

$$\forall f \in \mathscr{C}_K^\infty(\mathbb{D},\mathbb{R}), \exists K > 0, \ s.t. \ |\mathbb{E}(f(X_T^x)) - \mathbb{E}(f(\hat{X}_{t_n^t}^n))| \leq \frac{K}{n^r}.$$

We start by considering the infinitesimal generator L of CIR processes,

$$\begin{aligned} Lf(x) &= (a-kx)\frac{d}{dx}f(x) + \frac{1}{2}\sigma^2 x\frac{d^2}{dx^2}f(x), \\ &= (V_0 + \frac{1}{2}(V_1)^2)f(x). \end{aligned}$$

Here we decompose the generator into two parts, V_0 and V_1, where

$$V_0 f(x) = (a - kx - \frac{\sigma^2}{4}\frac{d}{dx})f(x), \tag{2.3}$$

$$V_1 f(x) = \sigma\sqrt{x}\frac{d}{dx}f(x). \tag{2.4}$$

Note that these two ODEs, $\frac{\partial}{\partial t}X_i(t,x) = V_i(X_i(t,x))$, could be solved explicitly, we have

$$X_0(t,x) = xe^{-kt} + (a - \frac{\sigma^2}{4})\psi_k(t), \tag{2.5}$$

$$X_1(t,x) = \left((\sqrt{x} + \frac{\sigma}{2}t)^+\right)^2. \tag{2.6}$$

where $\psi_k(t) = \frac{1-e^{-kt}}{k}$ for $k \neq 0$, and $\psi_0(t) = t$.

2.2.1 Weak 2nd order discretization

Here we define the function ϕ,

$$\phi(x,t,w) := e^{-\frac{k}{2}}\left(\sqrt{(a-\frac{\sigma^2}{4})\psi_k(\frac{t}{2}) + e^{-\frac{kt}{2}}x} + \frac{\sigma}{2}w\right)^2 + (a-\frac{\sigma^2}{4})\psi_k(\frac{t}{2}), \tag{2.7}$$

We have

$$\phi(x,t,\sqrt{t}N) = X_0\left(\frac{t}{2}, X_1\left(\sqrt{t}N, X_0(\frac{t}{2},x)\right)\right).$$

Proposition 2.1. *If $\sigma^2 \leq 4a$, $\phi(x,t,\sqrt{t}N)$ is well-defined and is a potential weak 2nd order scheme for L. Where $N \sim \mathcal{N}(0,1)$ [2].*

In case where $\sigma^2 > 4a$, we consider to simulate N with a bounded variable Y which fits the five first moments of a normal variable. We choose $Y^{(5)}$, s.t. $\mathbb{P}[Y = \sqrt{3}] = \frac{1}{6}$, $\mathbb{P}[Y = -\sqrt{3}] = \frac{1}{6}$ and $\mathbb{P}[Y = 0] = \frac{2}{3}$. This random variable is bounded by $\sqrt{3}$. Then we define the 2nd order boundary function $\mathbf{K}_2$ by

$$\mathbf{K}_2(t) := \mathbf{1}_{\sigma^2>4a}e^{\frac{kt}{2}}\left((\frac{\sigma^2}{4}-a)\psi_K(\frac{t}{2}) + \left[\sqrt{e^{\frac{kt}{2}}[(\frac{\sigma^2}{4}-a)\psi_k(\frac{t}{2})]} + \frac{\sigma}{2}\sqrt{3t}\right]^2\right) \tag{2.8}$$

Proposition 2.2. *For the case where $\sigma^2 > 4a$, If $x \geq \boldsymbol{K}_2(t)$, $\phi(x,t,\sqrt{t}Y)$ is well defined and is a potential weak 2nd order scheme for L[2].*

Finally we consider the case where $\sigma^2 > 4a$, and x is near to 0. Let $\tilde{u}_q(t,x) := \mathbb{E}[(X_t^x)^q]$ for $q \in \mathbb{N}$, $\Delta(t,x) := 1 - \frac{\tilde{u}_1(t,x)^2}{\tilde{u}_2(t,x)}$, and $\pi(t,x) := \frac{1-\sqrt{\Delta(t,x)}}{2}$.

Proposition 2.3. *Let $U \sim \mathscr{U}([0,1])$. Then $\hat{X}_t^x = g(t,x) := \mathbf{1}_{U \leq \pi(t,x)} \frac{\tilde{u}_1(t,x)}{2\pi(t,x)} + \mathbf{1}_{U > \pi(t,x)} \frac{\tilde{u}_1(t,x)}{2(1-\pi(t,x))}$ is a potential second order scheme on $x \in [0, \boldsymbol{K}_2(t)]$.*

Therefore, the iteration of this weak 2nd order discretization is :

1. If $\sigma^2 \leq 4a$: $\hat{X}^x_{t_n^{i+1}} = \phi(\hat{X}_{t_n^i}, h, \sqrt{h}N_i)$, where $h = T/n$, and $N_i \sim \mathscr{N}(0,1)$.
2. If $\sigma^2 > 4a$:
   — If $\hat{X}^x_{t_n^i} \geq \mathbf{K}_2(h)$: $\hat{X}^x_{t_n^{i+1}} = \phi(\hat{X}^x_{t_n^i}, h, \sqrt{h}Y_i)$, where $Y_i \overset{\text{law}}{=} Y^{(5)}$.
   — If $\hat{X}^x_{t_n^i} < \mathbf{K}_2(h)$: $\hat{X}^x_{t_n^{i+1}} = g(h, \hat{X}^x_{t_n^i})$.

2.2.2 Weak 3rd order discretization

Similar to the 2nd order scheme, we define a bounded random variable $Y^{(7)}$ which matches the first seven moments of standard normal variables, whose law is : $\mathbb{P}[Y = \sqrt{3+\sqrt{6}}] = \mathbb{P}[Y = -\sqrt{3+\sqrt{6}}] = \frac{\sqrt{6}-2}{4\sqrt{6}}$, and $\mathbb{P}[Y = \sqrt{3-\sqrt{6}}] = \mathbb{P}[Y = -\sqrt{3-\sqrt{6}}] = \frac{1}{2} - \frac{\sqrt{6}-2}{4\sqrt{6}}$. The 3rd order boundary function $\mathbf{K}_3$ is defined as :

$$\mathbf{K}_3(t) := \psi_{-k}(t) \left[\mathbf{1}_{4a/3<\sigma^2<4a} \left(\sqrt{\frac{\sigma^2}{4} - a + \frac{\sigma}{\sqrt{2}}\sqrt{a - \frac{\sigma^2}{4}}} + \frac{\sigma}{2}\sqrt{3+\sqrt{6}} \right)^2 \right.$$
$$\left. + \mathbf{1}_{\sigma^2 \leq 4a/3} \frac{\sigma}{\sqrt{2}}\sqrt{a - \frac{\sigma^2}{4}} + \mathbf{1}_{4a<\sigma^2} \left(\frac{\sigma^2}{4} - a + \left(\sqrt{\frac{\sigma}{\sqrt{2}}\sqrt{\frac{\sigma^2}{4} - a}} + \frac{\sigma}{2}\sqrt{3+\sqrt{6}} \right)^2 \right) \right] \text{[2]}.$$

For $x \geq \mathbf{K}_3(t)$, we firstly consider the case that $k = 0$. Let ϵ and ζ be independent uniform random variables on $\{-1,1\}$ and $\{1,2,3\}$ respectively, we define that for $\sigma\text{^}2 \leq 4a$ (resp. $\sigma\hat{a}2 > 4a$),

$$\hat{X}_t^{x,k=0} :=$$
$$\begin{cases} \tilde{X}(\epsilon t, X_0(t, X_1(\sqrt{t}Y, x))) & (\text{resp. } \tilde{X}(\epsilon t, X_1(\sqrt{t}Y, X_0(t,x)))) \quad \text{if } \zeta = 1, \\ X_0(t, \tilde{X}(\epsilon t, X_1(\sqrt{t}Y, x))) & (\text{resp. } X_1(\sqrt{t}Y, \tilde{X}(\epsilon t, X_0(t,x)))) \quad \text{if } \zeta = 2, \\ X_0(t, X_1(\sqrt{t}Y, \tilde{X}(\epsilon t, x))) & (\text{resp. } X_1(\sqrt{t}Y, X_0(t, \tilde{X}(\epsilon t, x)))) \quad \text{if } \zeta = 3. \end{cases}$$

Proposition 2.4. *For $x \geq \boldsymbol{K}_3(t)$, the scheme*

$$\hat{X}_t^x = e^{-kt} \hat{X}^{x,k=0}_{\psi_{-k}(t)}$$

is well defined and is a potential 3rd order scheme[2].

For $x \in [0, \mathbf{K}_3(t)]$, set $m_i := \mathbb{E}[X^i]$. Let $s = \frac{m_3 - m_1 m_3}{m_2 - m_1^2}$, $p = \frac{m_1 m_3 - m_2^2}{m_2 - m_1^2}$, and $= s^2 - 4p$. We have $\Delta > 0$, and therefore we define $x_\pm = \frac{s \pm \sqrt{\Delta}}{2}$ and $\pi = \frac{m_1 - x_-}{x_+ - x_-}$.

Proposition 2.5. *For $x \in [0, \boldsymbol{K}_3(t)]$, let $U \sim \mathscr{U}([0,1])$. Then*

$$\hat{X}_t^x := \boldsymbol{1}_{U \le \pi(t,x)} x_+(t,x) + \boldsymbol{1}_{U > \pi(t,x)} x_-(t,x)$$

is well defined and is a potential 3rd order scheme[2].

3 Simulation of Wishart Process

Based on our work on CIR process, we moved on to simulate the more general Wishart process on several dimension. Here, we will first study some important properties with which the simulation can be transformed to simpler form. Then the exact simulation method will be given with help of those properties. Ind in the end we will implement the discretisation scheme.

3.1 Important properties of Wishart Process

In this section, we present the properties of Wishart processes that are necessary for developing the simulation methods.

Some identities in law for Wishart processes. We start by introducing three important properties of affine processes which allow us to simplify our simulation.

Proposition 3.1. *For each Wishart process, its infinitesimal generator depends on a through $a^T a$, i.e.,*

$$AFF_d(x, \bar{\alpha}, B, a) \overset{law}{=} AFF(x, \bar{\alpha}, B, \sqrt{a^T a}).[1]$$

This proposition makes it possible to modify a as far as $a^T a$ is kept unchanged.

Proposition 3.2. *Let $n = RK(a)$ be the rank of $a^T a$. Then there exists a diagonal matrix δ and a non-singular matrix $u \in \mathcal{G}_d(\mathbb{R})$ (the general linear group on $\mathbb{R}$ of dimension d), s.t. $\bar{\alpha} = u^T \delta u$ $a^T a = u^T I_d^n u$, and we have*

$$AFF_d(x, \bar{\alpha}, B, a) \overset{law}{=} u^T AFF_d((u^{-1})^T x u^{-1}, \delta, B_u, I_d^n) u,$$

where $B_u(x) := (u^T)^{-1} B(u^T x u) u^{-1}$, and I_d^n is the matrix of which the first n diagonal entries are 1 and all the other entries are 0[1].

Using this property, we could simulate a general Wishart process by simulating a Wishart process whose a matrix equals to I_d^n.

Considering the marginal law of Wishart process at a fixed time point t, we have the following property :

Proposition 3.3. *Let $t > 0$, and $\alpha \geq d - 1$. Let $m_t := \exp(tb)$, $q_t := \int_0^t \exp(sb) a^T a \exp sb^T ds$ and $n = RK(q_t)$. Then there exists $\theta_t \in \mathcal{G}_d(\mathbb{R})$, s.t. $q_t = t\theta_t I_d^n \theta_t^T$, and we have*

$$WIS_d(x, \alpha, b, a; t) \overset{law}{=} \theta_t WIS_d(\theta_t^{-1} m_t x m_t^T (\theta_t^{-1})^T, \alpha, 0, I_d^n; t) \theta_t^T [1].$$

This proposition gives us the capacity of simulating the distribution $\mathrm{WIS}_d(x, \alpha, b, a; t)$ of a general Wishart process at a specific time t as soon as we are abel to simulate $\mathrm{WIS}_d(x, \alpha, 0, I_d^n; t)$.

3.2 EXACT SIMULATION

As we have mentioned in the first section, with 3.1, 3.2 and 3.1, the simulation of a Wishart process $WIS_d(x,\alpha,b,a;t)$ can be simplified to simulating $\mathrm{WIS}_d(x,\alpha,0,I_d^n;t)$.

In order to simulate this process, one splitting operator has been found as follows :

Theorem 3.1. *Let L be the generator of the Wishart process $WIS_d(x,\alpha,0,I_d^n)$, and $L_{e_d^i}$ the generator of $WIS_d(x,\alpha,0,e_d^i)$, then we have*

$$L=\sum_{i=1}^{n} L_{e_d^i}, \tag{3.1}$$

and $\forall i,j,\ L_{e_d^i}L_{e_d^j}=L_{e_d^j}L_{e_d^i}$ [1].

This theorem contributes to the essential part of this simulation : we can simulate the exact Wishart process for any admissble parameters by following decomposition :

$$\begin{aligned} X_t^{1,x} &\sim \mathrm{WIS}_d(x,\ \alpha,\ 0,\ e_d^1;\ t)\\ X_t^{2,X_t^{1,x}} &\sim \mathrm{WIS}_d(X_t^{1,x},\ \alpha,\ 0,\ e_d^2;\ t)\\ &\cdots\\ X_t^{n,\ldots^{X_t^{1,x}}} &\sim \mathrm{WIS}_d(X_t^{n-1,\ldots^{X_t^{1,x}}},\ \alpha,\ 0,\ e_d^2;\ t) \end{aligned}$$

In this way one import proposition can be made :

Proposition 3.4. *Let $X_t^{n-1,\ldots^{X_t^{1,x}}}$ be defined as above. Then*

$$X_t^{n-1,\ldots^{X_t^{1,x}}} \sim WIS_d(x,\alpha,0,I_d^n;t) \tag{3.2}$$

The first observation is that the step i with e_d^i could be calculated in the same way of e_d^1 with a permutation. Thus, the rest of this simulation is how to generate the first step of this decomposition.

In fact, with the help of CIR process who can be simulated explicitly as mentioned in section 2, we are able to write the explicit formula for $X_t^{1,x}$.

Precisely, Let $(Z_t^l)_{1\le l\le r+1}$ be a vector of independent standard Brownien motions and define :

$$\begin{aligned} d(U_t^u)_{1,1} &= (\alpha - r)dt + 2\sqrt{(U_t^u)_{1,1}}dZ_t^1\\ d((U_t^u)_{\{1,l+1\}})_{1\le l\le r} &= (dZ_t^{l+1})_{1\le l\le r} \end{aligned} \tag{3.3}$$

We remark that $(U_t^u)_{1,1}$ is a CIR process, and (p,k_r,c_r) the Cholesky decomposition of

$x_{i,j2\le i,j\le d}$. The solution of X_t^x can be given explicitly by :

$$
\begin{aligned}
X_t^x = & \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_r & 0 \\ 0 & k_r & I_{d-r-1} \end{pmatrix} \\
\times & \begin{pmatrix} (U_t^u)_{1,1} + \sum_{k=1}^{r} ((U_t^u)_{\{1,k+1\}}))^2 & ((U_t^u)_{\{1,l+1\}}))_{1\le l\le r}^T & 0 \\ ((U_t^u)_{\{1,l+1\}}))_{1\le l\le r} & I_r & 0 \\ 0 & k_r & I_{d-r-1} \end{pmatrix} \\
\times & \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_r & 0 \\ 0 & k_r & I_{d-r-1} \end{pmatrix}
\end{aligned}
$$

In this way, the exact simulation and corresponding algorithms are given.

To make it more clear, we show this calculation by following steps.

— Use exact CIR process to simulate $U_{1,1}$ and define $U_{\{1,l+1\}}$ as in 3.2.
— Define $X_t^{1,x}$ with U and the Cholesky decomposition[4] of $x_{i,j2\le i,j\le d}$ as in 3.2, where $X_t^{1,x}$ is in fact WIS$(x,\alpha,0,e_d^1;t)$.
— Depending on 3.2 and 3.2, simulate WIS$(x,\alpha,0,I_d^n;t)$ step by step.
— Depending en 3.1, simulate WIS$(x,\alpha,b,a;t)$ exactly.

3.3 Discretisation schemes

Given the discretisation schemes of cir process, the corresponding discretisation schemes for Wishart process is simple. Similarly, we show its calculation by step :

— Use 2nd order discretization (resp. 3rd order discretization) CIR process to simulate $U_{1,1}$ and 2nd (resp. 3rd) order bounded Gaussian distribution to define $U_{\{1,l+1\}}$ as in 3.2.
— Define X_t^x with 2nd (resp. 3rd) order Gaussian distribution, 2nd (resp. 3rd) order U, and the Cholesky decomposition of $x_{i,j:2\le i,j\le d}$ as in 3.2, where X_t^x is simulated as WIS$(x,\alpha,0,e_d^1,t)$.
— Based on 3.2 and 3.2, simulate WIS$(x,\alpha,0,I_d^n;t)$ step by step .
— Based en 3.1, simulate 2nd (resp. 3rd) WIS$(x,\alpha,b,a;t)$.
— Let x be the simulated value of $X_t^{1,x}$, repeat from the first step.

3.4 Application example

There are several models based on the Wishart process. Here we take the *Gourieroux and Sufana* model[3] as an exemple.

This model considers d risky assets $S_t = (S_t^1,\dots,S_t^d)$ of which the covariance matrix is

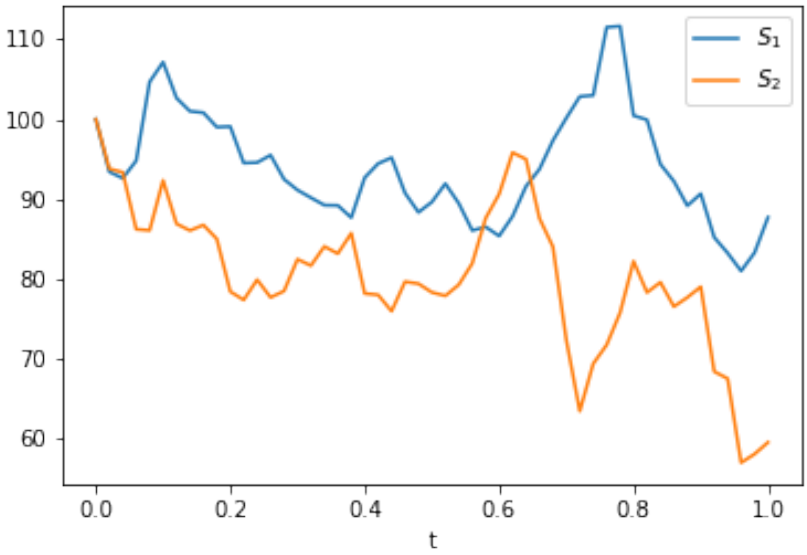

FIGURE 3.1 – Example of GS model, $S_0 = (100, 100)$, $r = 0.02$, $X_0 = 0.04I_2 + 0.02\mathbb{1}_{i\neq j}$, $\alpha = 4.5$, $a = 0.2I_2$, $b = 0.5I_2$, $T = 1$.

stochastic following the Wishart process.

$$dS_t = rS_t + (\sqrt{X_t}dB_t)^T S_t,$$
$$dX_t = (\alpha a^T a + bX_t + X_t b^T)dt + (\sqrt{X_t}dW_t a + a^T dW_t^T \sqrt{X_t}).$$

We could use the methods presented in this article to simulate the Wishart process and then simulate the GS model. We give an example in figure 3.1 of the path of S_t^1 nad S_t^2 for a model of dimension 2.

4 Analysis of Results

In this part, we will verify that the result is correct and then show the result of CIR and Wishart process simulation using exact and discretisaiton method.

As the simulation of Wishart process is based on the simulation of CIR process, the verification of the former simulation is able to confirm the correctness for both implementation.

Proposition 4.1. *Let* $X_t^x \sim WIS_d(x,\alpha,b,a;t)$, $m_t := \exp(tb)$ *and* $q_t := \int_0^t \exp(sb)a^T a \exp sb^T ds$ *and* $\mathscr{D}_{b,a;t} = \{v \in \mathscr{S}_d(\mathbb{R}), \mathbb{E}[exp(Tr(vX_t^x)] \leq \inf\}$. $\mathscr{D}_{b,a;t}$ *can be given explicitly by*

$$\mathscr{D}_{b,a;t} = \{v \in \mathscr{S}_d(\mathbb{R}), \forall s \in [0,t], I_d - 2q_s v \in \mathscr{G}_d(\mathbb{R})\}$$

Then the Laplace transform of X_t^x *is well defined for* $v = v_R + i v_I$ *with* $v_R \in \mathscr{D}_{b,a;t}$, $v_I \in \mathscr{S}_d(\mathbb{R})$ *and is given by*

$$\mathbb{E}[\exp(Tr(vX_t^x))] = \frac{\exp(Tr[v(I_d - 2q_t v)^{-1} m_t x m_t^T])}{\det(I_d - 2q_t v)^{\alpha/2}} \tag{4.1}$$

With this proposition, we implement a method to verify the correctness of the model.

With the configuration of simulations which is $M = 10^7, k = 1/2, a = 1/2, \sigma = 0.8, T = 1, x_0 = 3/2$ as in the paper[2], the test result is given in figure4.1.

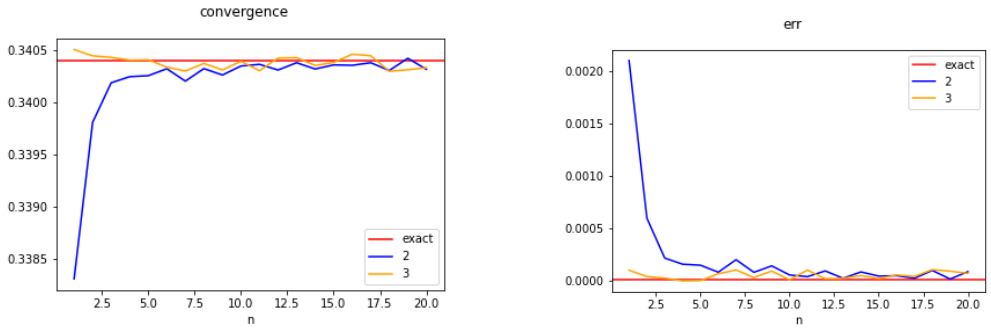

Figure 4.1 – Comparison of precision of CIR process simulation using different method

Similarly, the result of Wishart process simulation is shown in figure4.2, where we use the following parameters $x_0 = 0.4I_3, a = I_3, b = 0, \alpha = 4.5$.

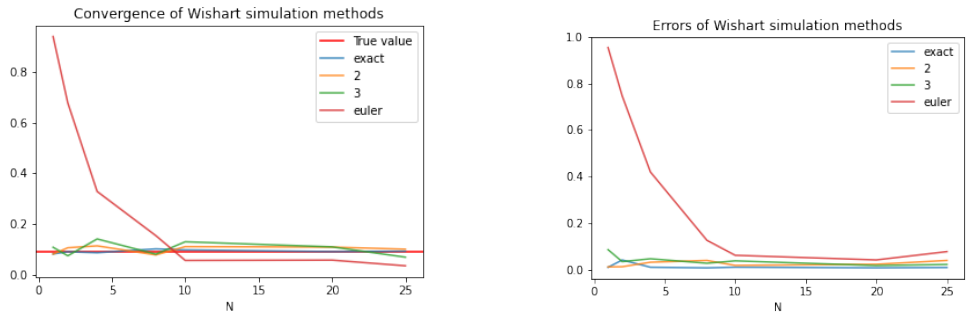

FIGURE 4.2 – Comparison of precision of Wishart process simulation using different method

5 CODE STRUCTURE

In this project, we created a git repository to keep the codes. The codes are organised in four parts :

- `wishart` : The module defining Wishart simulator. In which, `simulation.py` is the core module, where the exact and discretized simulation of Wishart processes are given. `comparison.py` is used to check the validity of the implemented simulation by comparing their result of a certain function. In `utils.py`, some commonly used functions are defined, such like `decompose_cholesky` and `brownian`.
- `cir` : The module defining CIR simulator. In which `simulation.py` defines the exact and discretized simulations of CIR processes.
- `application` : In the second stage of this project, we will study more specific financial applications based on the Wishart processes, this folder is created for future implementations.
- `sampling` : The module defining some important sampling methods in this project like `bounded_gauss`, as they are not provided by existing python package.
- `test_result` : The test file and result obtained of the project, namely the temporary test for Wishart and CIR processes, and the mid-term and final report, will be found in this folder.

6 FUTURE PLAN

Having implemented the simulation methods for CIR processes as well as for Wishart processes, our next object is to apply the simulation to financial mathematics models, e.g. the Gourieroux and Sufana model, one of the applications on option pricing with correlated assets[5].

RÉFÉRENCES

[1] A. Abdelkoddousse and A. Aurélien. Exact and high-order discretization schemes for wishart processes and their affine extensions. *The Annals of Applied Probability*, 23(3) :1025–1073, 2013.

[2] A. Aurélien. High order discretization schemes for the cir process : Application to affine term structure and heston models. *Mathematics of Computation*, 79(269) :209–237, January 2010.

[3] G. Christian and S. Razvan. Derivative pricing with multivariate stochastic volatility : Application to credit risk. *SSRN Electronic Journal*, Februray 2004.

[4] G. H. Goluv and C. F. Van Loan. *Matrix Computations*. 3rn ed. Jogns Hopkins Univ.Press.

[5] F. Jose, Da, G. Martino, and T. Claudio. Option pricing when correlations are stochastic : an analytical framework. *Review of Derivatives Research*, 10 :151–180, January 2008.

[6] B. Marie-France. Wishart process. *Journal of Theoretical Probability*, 4(4“, DOI =), April 1991.

[7] J. A. M. Simon. Chi-square simulation of the cir process and the heston model. *International Journal of Theorical and Applied Finance*, 2008.