Колінько Микола. Варіант 7.
Домашня робота №3

Завдання 1

Маємо стохастичне диференціальне рівняння

$$dX_t = (\mu_1 X_t + \mu_2)dt + (\sigma_1 X_t + \sigma_2)dW_t.$$

Застосуємо формулу Іто для добутку $X_t S_t$, де $S_t = e^{at+bW_t}$. Це багатовимірний випадок з фунцією $g = xe^{at+by}$ та $Y_t = (X_t, W_t)$. Рівняння для dX_t вставимо в рівняння для $X_t S_t$.

$$\begin{aligned}
&dX_t = (\mu_1 X_t + \mu_2)dt + (\sigma_1 X_t + \sigma_2)dW_t, \\
&dX_t S_t = (a + \tfrac{b^2}{2})X_t S_t dt + S_t dX_t + bX_t S_t dW_t + bS_t(\sigma_1 X_t + \sigma_2)dt = \\
&= S_t\left(\left((a + \tfrac{b^2+2b\sigma_1}{2} + \mu_1)X_t + \mu_2 + b\sigma_2\right)dt + ((b+\sigma_1)X_t + \sigma_2)dW_t\right),
\end{aligned}$$

Покладемо $b = -\sigma_1,\ a = -\mu_1 + \frac{\sigma_1^2}{2}$, тоді деякі доданки скоротяться і отримаємо

$$\begin{aligned}
&dX_t e^{-(\mu_1 - \frac{\sigma_1^2}{2})t - \sigma_1 W_t} = e^{-(\mu_1 - \frac{\sigma_1^2}{2})t - \sigma_1 W_t}\left((\mu_2 - \sigma_1\sigma_2)dt + \sigma_2 dW_t\right), \\
&X_t = e^{(\mu_1 - \frac{\sigma_1^2}{2})t + \sigma_1 W_t}\left(X_0 + \int_0^t e^{-(\mu_1 - \frac{\sigma_1^2}{2})s - \sigma_1 W_s}(\mu_2 - \sigma_1\sigma_2)\,ds + \int_0^t \sigma_2 e^{-(\mu_1 - \frac{\sigma_1^2}{2})s - \sigma_1 W_s}\,dW_s\right), \\
&E[X_t] = e^{\mu_1 t}(x_0 - \tfrac{\mu_2}{\mu_1}(e^{-\mu_1 t} - 1)).
\end{aligned}$$

$$\begin{aligned}
&case\ \mu_1 = \sigma_2 = 0 : X_t = x_0 e^{(\mu_1 - \frac{\sigma_1^2}{2})t + \sigma_1 W_t}, E[X_t] = x_0 e^{\mu_1 t} \\
&case\ \mu_1 = \sigma_1 = 0 : X_t = e^{\mu_1 t}\left(x_0 + \sigma_2 \int_0^t e^{-\mu_1 s} dW_s\right), E[X_t] = x_0 e^{\mu_1 t}
\end{aligned}$$

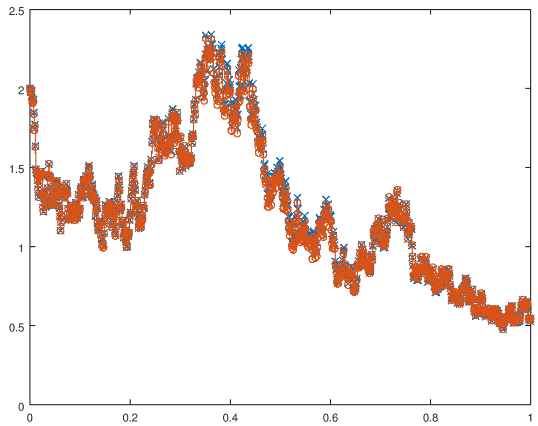
Підставимо коефіцієнти 19 варіанту: $x_0 = 2, \mu_1 = -1, \mu_2 = 0, \sigma_1 = -1.3, \sigma_2 = 0$

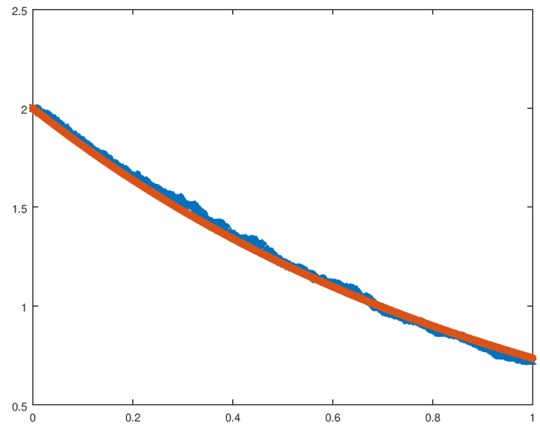
$$X_t = 2e^{(-1 - \frac{(-1.3)^2}{2})t - 1.3W_t},$$

Вигляд наближення Ейлера для наших коефіцієнтів

$$X_{t+dt} = -X_t dt - 1.3X_t(W_{t+dt} - W_t), \quad X_0 = 2.$$

Просимулюємо наближений та точний розв'язок для різних діаметрів розбиття dt та часів зупинки T. Нижче наведено графік однієї траекторії наближеного і точного розв'язку та графік усередненого і матсподівання точного розвязку.





		Діаметр розбиття		
		$\frac{1}{10}$	$\frac{1}{100}$	$\frac{1}{1000}$
T	1	Середнє: 1.9835 Медіана: 1.3105 σ: 2.2039	Середнє: 2.1729 Медіана: 1.4052 σ: 2.4065	Середнє: 2.2706 Медіана: 1.5615 σ: 2.2741
	10	Середнє: 2.1688 Медіана: 1.4153 σ: 2.3235	Середнє: 2.3485 Медіана: 1.4248 σ: 2.8303	Середнє: 2.5618 Медіана: 1.5966 σ: 2.9760
	50	Середнє: 2.1162 Медіана: 1.1300 σ: 2.5750	Середнє: 2.9602 Медіана: 1.7926 σ: 4.1215	Середнє: 2.0632 Медіана: 1.4073 σ: 2.0051

Результати осаннього рядка для 100 траекторій, інакше програма працює дуже повільно. Всі інші результати для 1000 траекторій. Обчислення виконані за допомогою програми мовою Matlab

```
function MMS = euler()
  load("par.mat")
  for j = 1:n_iter
    #wiener process
    dW = sqrt(dt).*randn(1,N-1);
    W = [0, cumsum(dW)];

    #euler method
    X(1) = x0;
    for n=2:N X(n) = X(n-1) + (a*X(n-1) + b).*dt + (c*X(n-1) + d).*dW(n-1); end;
    #figure(1); plot(t,X,"-x",t,x(W),"-o");

    allX(j,:) = X;
    normvec(j) = max(abs(X-x(t)));
    clear dW W X EXP;
  end

```

```
  #values for table
  Mean = mean(normvec);
  Median = median(normvec);
  Std = std(normvec);
  MMS = [Mean, Median, Std];

  MX = mean(allX);
  figure(2); plot(t,MX,"-x",t,mx,"-o");
end
```

```
function [] = init()
  #T - right endpoint, t - uniform partition of [0, T] with N points and diametr of partition dt
  T=1; N=1001; dt=T/(N-1); t=0:dt:T;
  #number of iterations
  n_iter = 1000;

  # coefficients
  x0 = 2;
  a = -1;
  b = 0;
  c = -1.3;
  d = 0;

  #solution
  if (b==0 & d==0) x = @(w)(x0*exp((a-c^2/2)*t+c*w));
  elseif (b==0 & c==0) x = @(w)(exp(a*t).*(cumsum(exp(-a*t).*[x0,diff(w)])));
  else x = @(w)(exp(a*t).*(x0 + cumsum((b-c*d)*exp(-(a-c^2/2)*t-c*w).*dt) +
      cumsum(d*exp(-(a-c^2/2)*t-c*w).*[0,diff(w)])));
  endif

  #mean solution
  if (b==0 & d==0) mx = x0*exp(a*t);
  elseif (b==0 & c==0) mx = x0*exp(a*t);
  elseif (a==0) mx = (x0 + b*t).*exp(a*t);
  else mx = (x0 - (b/a)*(exp(-a*t)-1)).*exp(a*t);
  endif

    save par.mat
end;
```