DTU Compute

02425 Diffusion and SDEs
November 23, 2021
UHT/uht

Exercises for session 7:
Existence and uniqueness, numerics, and Stratonovich SDEs

Conditions for existence and uniqueness

Question 1: Determine which of the following Initial Value Problems satisfy the sufficient condition for existence and uniqueness of a solution.

1. $dX_t = \sin X_t \; dt + \cos X_t \; dB_t$, $X_0 = 0$
2. $dX_t = e^{X_t} \; dt + dB_t$, $X_0 = 0$
3. $dX_t = -X_t^3 \; dt + dB_t$, $X_0 = 0$
4. $dX_t = \lambda(\mu - X_t) \; dt + \sigma\sqrt{|X_t|} \; dB_t$, $X_0 = \mu$, where $\mu > 0$ and $\lambda > 0$.

Optional: For each case, if the conditions are not met, can we expect solutions to fork or explode?

Solution: We use theorem 8.2.2 (page 177) which combines the uniqueness result in theorem 8.1.1 (p. 172) with the growth bound in theorem 8.2.1 (p. 176). Note that a differentiable functions are locally Lipschitz, and even globally Lipschitz if and only if the derivative is bounded. Note also that a globally Lipschitz drift function f automatically satisfies the growth bound in theorem 8.2.1.

1. The drift $f(x) = \sin x$ and the noise intensity $g(x) = \cos x$ are both globally Lipschitz continuous, so a unique solution is guaranteed to exist.
2. The drift $f(x) = e^x$ is locally Lipschitz, so there can be at most one solution. However, it is not globally Lipschitz and does not satisfy the linear growth bound, since the superlinearity is destabilizing. We expect the solution to explode at a finite random time.
3. The drift $f(x) = -x^3$ is not globally Lipschitz continuous, but does satisfy the linear growth bound for any $C > 0$. This is because the superlinearity is stabilizing, i.e. directed towards the origin. We are guaranteed a unique solution.
4. The drift $f(x) = \lambda(\mu - x)$ is globally Lipschitz, but the noise intensity $\sqrt{|x|}$ has a singularity at $x = 0$ where the tangent is vertical. The sufficient conditions of theorem 8.1.1 do not hold.

   We can therefore only guarantee a unique solution up to the (random) time where the process hits the origin. It requires a more careful treatment to determine if the process may in fact hit $x = 0$ (the answer turns out to depend on σ). In case the process may hit $x = 0$, it requires a stronger theorem to prove existence and uniqueness (such a theorem is given in Karatzas & Shreve, chapter 5.2).

The physical pendulum

Consider the two coupled SDE's which model a physical pendulum with friction and noise:

$$dX_t = V_t\ dt, \qquad dV_t = \sin X_t\ dt - \lambda V_t\ dt + \sigma\ dB_t \quad .$$

Question 2: Show that the system satisfies the conditions for existence and uniqueness of solutions, and that the Itō interpretation and the Stratonovich interpretation of the equation are identical.

Solution: The drift function $(x, v) \mapsto (v, \sin x - \lambda v)$ is globally Lipschitz. This is most easily seen by establishing the Jacobian

$$\nabla f = \begin{bmatrix} 0 & 1 \\ \cos x & -\lambda \end{bmatrix}$$

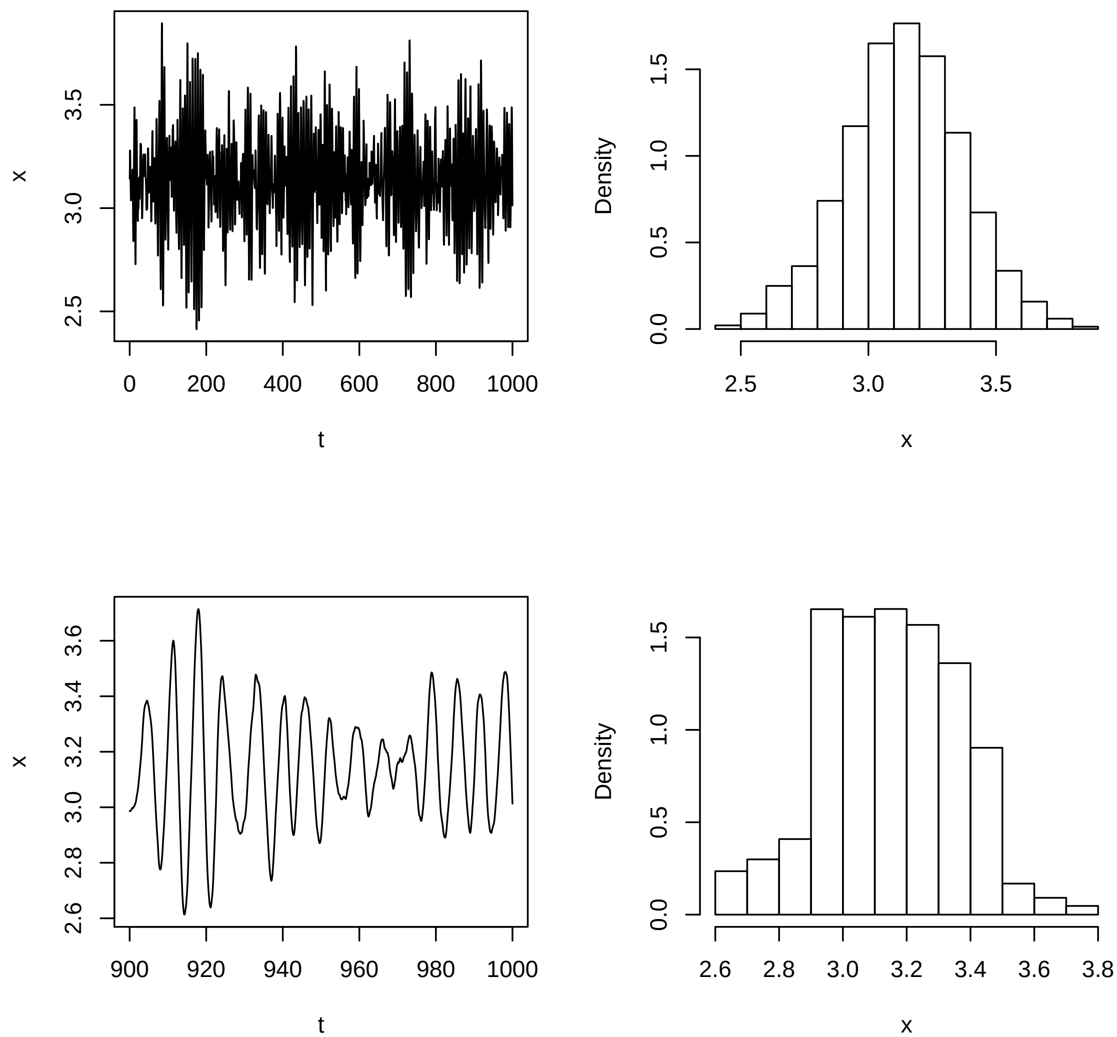
which is bounded element-wise by $\max(1, |\lambda|)$. Since the noise intensity is constant, the model is globally Lipschitz, so that existence and uniqueness is guaranteed.

The noise in the model is additive ($\nabla g = 0$), so the Itō and Stratonovich interpretation coincide.

Question 3: Simulate the system with the Euler method for $t \in [0, 1000]$, taking $\lambda = 0.1$ and $\sigma = 0.01$. Use $dt = 0.01$ and $X_0 = 0$, $V_0 = 0$. Plot the solution as a function of time, and a histogram of the X. Does it appear that the solution reflects the long term behavior of the process? Repeat for larger values of σ, up to $\sqrt{2}$, say.

Solution: The following code performs the simulation

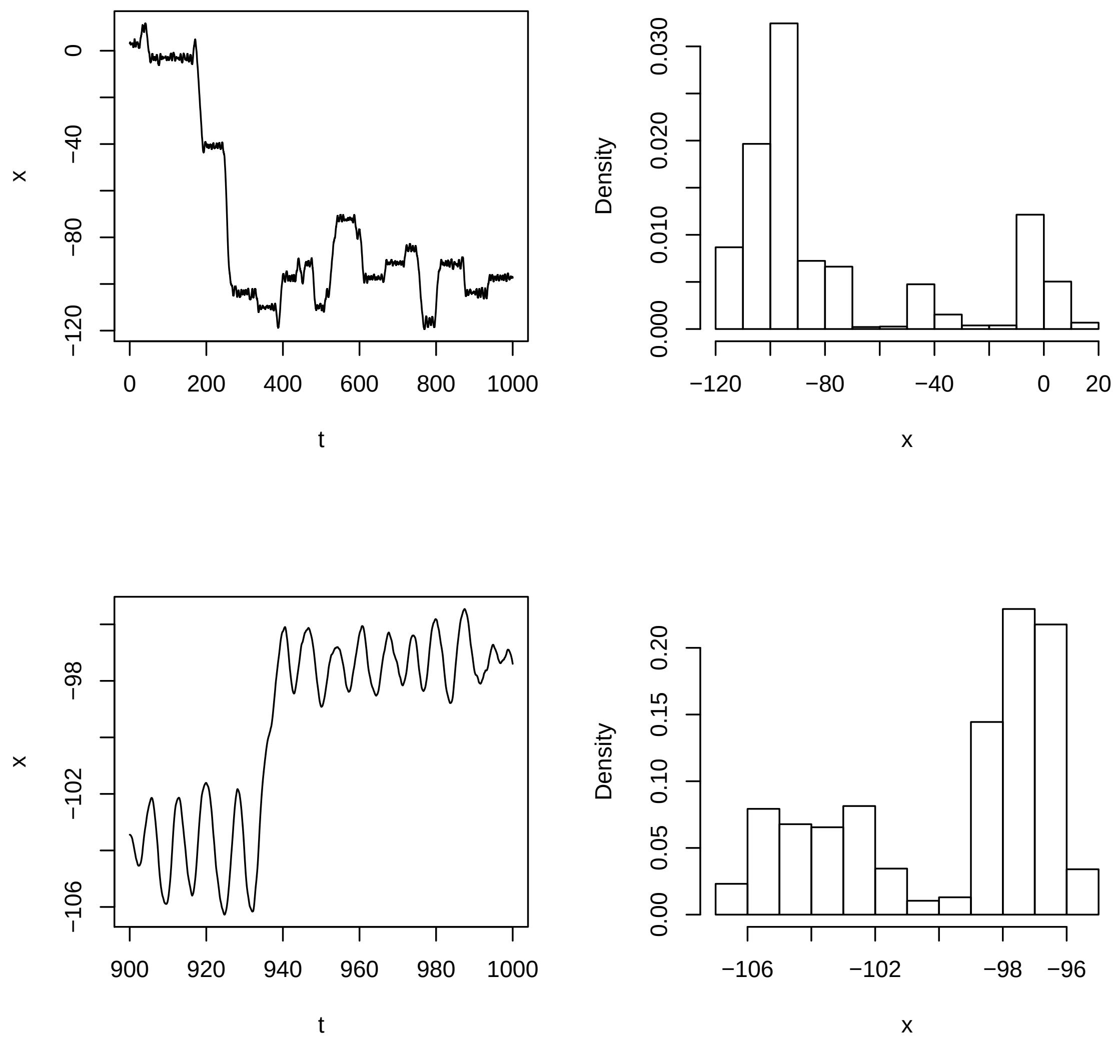
```r
require(SDEtools)

## Loading required package:  SDEtools

## Define drift and noise terms
f <- function(x) c(x[2],sin(x[1]) - lambda*x[2])
g <- function(x) c(0,sigma)

## System parameters
lambda <- 0.1
sigma <- 0.1

## Simulation parameters
dt <- 0.01
x0 <- c(pi,sqrt(0.5)*sigma)
T <- 1000

tv <- seq(0,T,dt)

## Perform the simulation with the Euler method
B <- rBM(tv)
sol <- euler(f,g,tv,x0,B)

## Plot also the last 10 percent
I <- tv>0.9*T

par(mfrow=c(2,2))
plot(sol$times,sol$X[,1],type="l",xlab="t",ylab="x")
hist(sol$X[,1],main="",xlab="x",freq=FALSE)
plot(sol$times[I],sol$X[I,1],type="l",xlab="t",ylab="x")
hist(sol$X[I,1],main="",xlab="x",freq=FALSE)
```



It appears that the simulation is stationary: There is no obvious long-term pattern in the time series. When we compare the histogram based on the entire series, with that based on only the last opart of the series, we see that they are fairly similar, taking the statistical uncertainty into account. When we inspect the short time series, we see the oscillations of the pendulum, as well as the intermittent behavior that the energy (i.e. the amplitude) fluctuates.

We repeat the simulation with $\sigma = 1$:

```
sigma <- 1/2
sol <- euler(f,g,tv,x0,B)
par(mfrow=c(2,2))
plot(sol$times,sol$X[,1],type="l",xlab="t",ylab="x")
hist(sol$X[,1],main="",xlab="x",freq=FALSE)
plot(sol$times[I],sol$X[I,1],type="l",xlab="t",ylab="x")
hist(sol$X[I,1],main="",xlab="x",freq=FALSE)
```



It is now clear that the solution is not stationary. What happens here is that oscillations become so energetic, that the pendulum may do a full circle. This the angle X_t wanders away from the origin in what resembles a random walk. When we inspect the shorter time series, we clearly see the "step" patterns in the angle. Here, the pendulum may spend prolonged periods of time without doing full revolutions, and on this scale it appears as if the angle is roughly constant but subject to noisy fluctuations. Then, occasionally, enough energy is built up that the pendulum does one or more full revolutions, corresponding to steps in the angle which are multiples of $\pm 2\pi$.

If we consider angles x only up to period 2π - i.e., we consider the motion to take place on the unit circle - then the process is stationary. However, we haven't introduced such a formalism.

We must now also reinterpret the results with low noise levels: The process only *appears* to be stationary, because full revolutions are such a rare event that we do not observe them in most simulations. Theoretically, they may happen, and therefore the process is not stationary.

Conversion between Itō and Stratonovich equations

Consider (again) the Itō equation governing the *Cox-Ingersoll-Ross process* (a.k.a. the *square root process*):

$$dX_t = \lambda(\mu - X_t)\ dt + \sigma\sqrt{X_t}\ dB_t \quad .$$

Question 4: Identify the equivalent Stratonovich equation; i.e., the Stratonovich equation that has the same solution.

Solution: We have the Itō equation

$$dX_t = f_I(X_t)\ dt + g(X_t)\ dB_t$$

with $f_I(x) = \lambda(\mu - x)$, $g(x) = \sigma\sqrt{x}$. This $\{X_t\}$ also satisfies the corresponding Stratonovich equation

$$dX_t = f_S(X_t)\ dt + g(X_t) \circ dB_t$$

where

$$f_S(x) = f_I(x) - \frac{1}{2}g'(x)g(x) = \lambda(\mu - x) - \frac{1}{4}\sigma^2$$

Question 5: Simulate the solution: Take $\mu = \lambda = 1$. Start with small values of σ (say, 0.1) and then increase σ gradually, using the same realization of Brownian motion for all values of σ. Plot the resulting trajectory as well as the histogram of the solution. *Note:* Theoretically, the solution $\{X_t\}$ stays non-negative, but the discrete-time approximation may take on negative values which can cause havoc. Take appropriate action to address this if necessary.

Solution: The following code is the main driver. It simulates the process both using the Itō and the Stratonovich formulation.

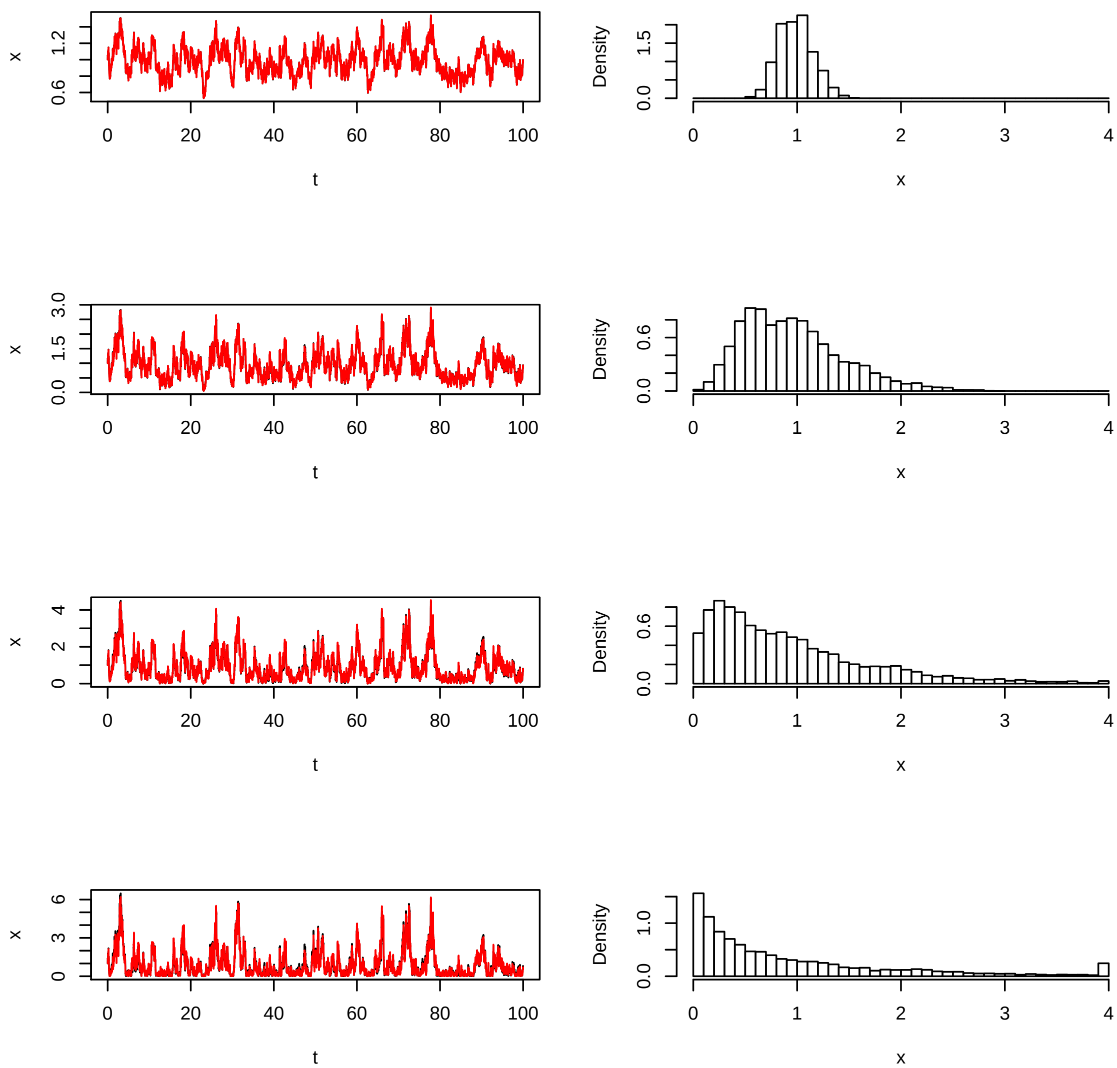
```r
lambda <- mu <- 1

sim <- function(sigma,tv,x0=mu,B=NULL)
    {
        g <- function(x) sigma*sqrt(abs(x))
        fI <- function(x) lambda*(mu-x)
        fS <- function(x) fI(x) - sigma^2/4

        if(is.null(B)) B <- rBM(tv)

        simI <- euler(f=fI,g=g,times=tv,x0=x0,B=B,p=abs)
        simS <- heun (f=fS,g=g,times=tv,x0=x0,B=B,p=abs)

        return(cbind(XI = simI$X,XS = simS$X))
    }
```

We run the code with various values of σ:

```r
sigmas <- seq(0.25,1.75,length=4)
tv <- seq(0,100,0.01)
B <- rBM(tv)
sols <- lapply(sigmas,function(s)sim(s,tv,B=B))
```

This produces the following trajectories:

```
par(mfrow=c(4,2))
xmax <- 4
xi <- seq(0,xmax,0.1)
for(sol in sols)
    {
        plot(tv,sol[,1],type="l",xlab="t",ylab="x")
        lines(tv,sol[,2],col="red")
        hist(pmin(sol[,2],xmax),breaks=xi,freq=FALSE,xlab="x",main="")
    }
```

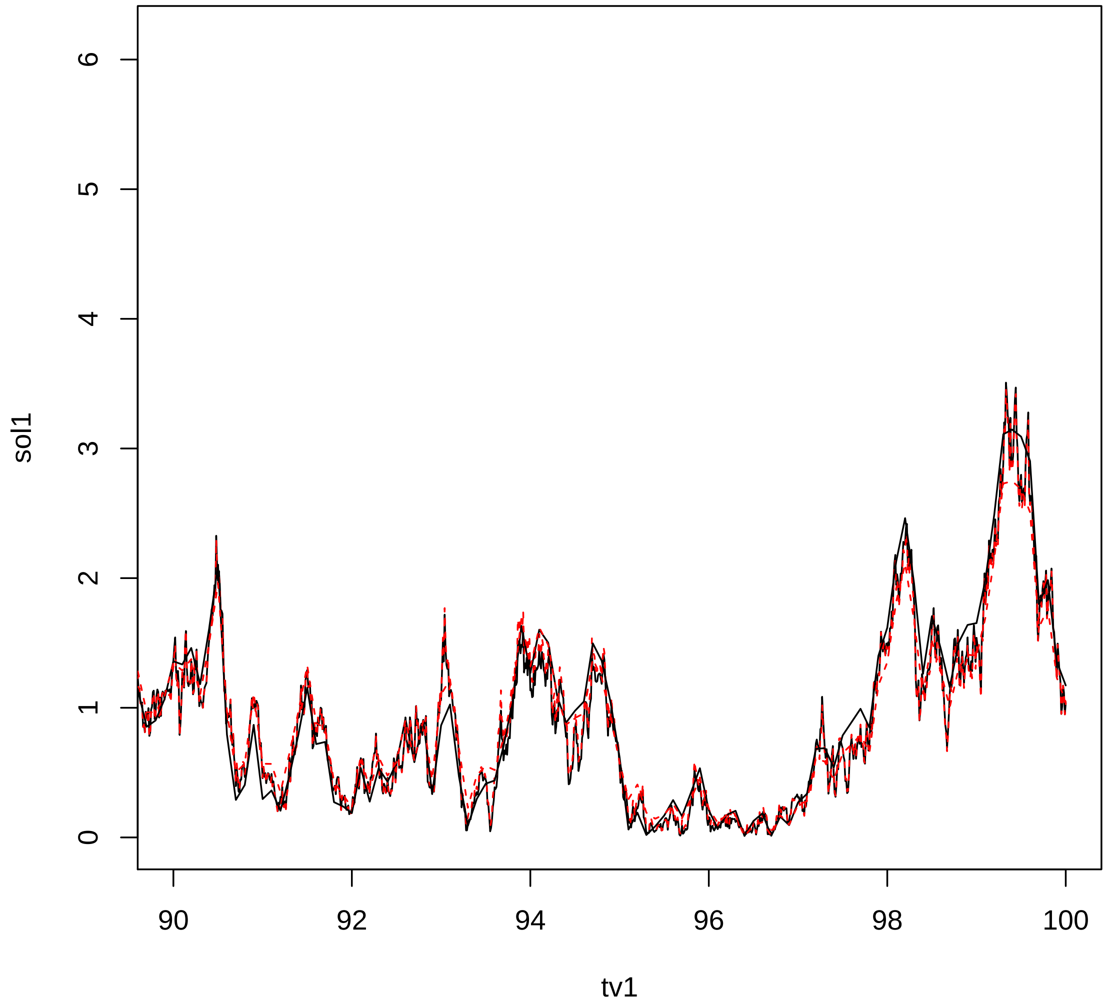


For low values of the noise intensity σ, the distribution resembles a Gaussian around the mean value μ. As the noise intensity σ increases, the distribution becomes more skewed and the right tail becomes more pronounced. The trajectories have an increased tendency to showing "spikes" or "bursts".

Question 6 Euler-Maruyama vs. Heun; the effect of time step: Take $\mu = \lambda = 1$ and $\sigma = 1.25$. Generate a single realization of Brownian motion for $t \in [0, 100]$ with a time step of 0.01. Simulate the process with both the Euler-Maruyama method (for the Itō formulation) and the Heun method (for the Stratonovich formulation). Check that the two solutions are reasonably close to eachother. Then subsample the Brownian motion to a time step of e.g. 0.1 and repeat the two simulations. Assess the sensitivity of the two schemes to the time step.

Solution:

```
sigma <- 1.25
tv1 <- seq(0,100,0.01)
B1 <- rBM(tv1)
sol1 <- sim(sigma,tv1,B=B1)

I <- seq(1,length(B1),10)
B2 <- B1[I]
tv2 <- tv1[I]

sol2 <- sim(sigma,tv2,B=B2)

## Plot only the tail, for increased clarity
matplot(tv1,sol1,type="l",xlim=c(90,100))
matplot(tv2,sol2,type="l",add=TRUE)
```



We compute the differences between the various solutions:

```r
## Difference between Ito and Stratonovich on the fine grid
print(mean(abs(sol1[,1]-sol1[,2])))

## [1] 0.03478116

## Difference between Ito and Stratonovich on the coarse grid
print(mean(abs(sol2[,1]-sol2[,2])))

## [1] 0.1355285

## Difference between fine and coarse grid for Ito
print(mean(abs(sol1[I,1]-sol2[,1])))

## [1] 0.109559

## Difference between fine and coarse grid for Ito
print(mean(abs(sol1[I,2]-sol2[,2])))

## [1] 0.05293363
```

With the fine grid, the Itō and Stratonovich solutions differ less than with the course grid. This is to be expected, because with the fine grid, both approximations are closer to the true solutions. Notice that a factor 10 in the step size leads to a factor of a bit more than 3 in the difference. This is consistent with an understanding that the error scales with the square root of the step size.

Changing step size has a more pronounced effect on the Itō results than on the Stratonovich results. This is to be expected, because the Euler-Maruyama method has lower order than the Heun method.