**Christian-Albrechts-Universität zu Kiel** **SS 2017**
Mathematisches Seminar
Prof. Dr. Mathias Vetter
Ole Martin, Adrian Theopold Sheet 03

# Computational Finance

Exercises for participants of the programme **'Quantitative Finance'**

**T-Exercise 08 (American put option in the CRR model)**

a) Let $W$ be a standard Brownian motion. Represent the process $X(t)tW(t)$ as an Itô-process, i.e. in the form

$$d(X(t)) = \ldots dt + \ldots dW(t)$$

b) Represent the process

$$Y(t) = \frac{W(t)}{1+t}$$

as an Itô-process.

**T-Exercise 09 (Exchange rates)**
Assume that the exchange rate $D(t)$ of the US-Dollar in Euro at time $t > 0$ follows the equation

$$dD(t) = D(t)\mu \mathrm{d}t + D(t)\sigma \mathrm{d}W(t)$$

with $D(0) > 0$ and $\mu$, $\sigma \in \mathbb{R}$. Hence, the exchange rate of the Euro in US-Dollar at time $t > 0$ is given by $E(t) \coloneqq \frac{1}{D(t)}$.

a) Represent the process $E$ as Itō process, i.e. in the form

$$dE(t) = \ldots \mathrm{d}t + \ldots \mathrm{d}W(t).$$

b) Use the stochastic exponential to compute $D(t)$ and $E(t)$.

c) Use your results from part a) or part b) to interpret the case $\mu = \frac{1}{2}\sigma^2$ economically.

**C-Exercise 10 (Adaptive step size control for the binomial method)**
We extend the algorithm of C-Exercise 06 by choosing the number of periods adaptively in the following way. All other parameters being fixed, we compute the fair price of the American put option for $M$ and $2M$ periods, respectively. The corresponding prices are denoted by $V_M$ and $V_{2M}$. If

$$\frac{|V_M - V_{2M}|}{V_M} < \varepsilon$$

for some fixed accuracy $\varepsilon > 0$, the algorithm accepts $V_{2M}$ as option price and returns this value. In the other case, we double the number of periods and compute the relative deviation of $V_{2M}$ and $V_{4M}$. If these values satisfy the above termination condition, the value $V_{4M}$ is returned. If not, the number of periods is doubled until the termination condition is satisfied, and the corresponding value is returned. Implement this algorithm in a scilab function

```
V0 = CRR_AmPut_Adapt (S_0, r, sigma, T, K, M, epsilon),
```

and test it for

$$S(0) = 100,\ r = 0.03,\ \sigma = 0.24,\ T = 3/4,\ K = 95,\ M = 500,\ \varepsilon = 0.001.$$

Please save your solution of each C-Exercise in a file named `Exercise_##.sce`, where ## denotes the number of the exercise. Please include your name(s) as comment in the beginning of the file.

**Submit until:** Fri, 12.05.2017, 10:00
**Discussion:** 15./17.05.2017,