Ergodic Theory and Transfer Operators
— Question sheet 6 —

Summer 2015

Exercise 1 (Reachability Revisited)
Let $X \subset \mathbb{R}^d$ with $\mu(X) < \infty$, where μ is the Lebesgue measure. Let $T : X \to X$ be ergodic with respect to μ. Show by using the Birkoff Ergodic Theorem that for almost all $x \in X$ the sequence $\{T^n x\}_{n \in \mathbb{N}}$ is dense in X.

Exercise 2 (Mixing)
Let $(X, \mathcal{B}, \mu, T)$ be a mixing ppt. Let ν be a probability measure on $\mathcal{B}$ absolutely continuous with respect to μ. Show that $\lim_{n\to\infty} \nu(T^{-n}A) = \mu(A)$ for $A \in \mathcal{B}$.

Hint: Express $\nu(T^{-n}A)$ by the Radon–Nikodým derivative $\frac{d\nu}{d\mu}$, and use Proposition 1.15B from the lectures.

Exercise 3 (Convergence of the Birkhoff Partial Sums)
We know from the Birkhoff Ergodic Theorem (in particular Corollary 2.5 and Theorem 2.6) that the stationary probability vector p of an irreducible stochastic matrix P, i.e. $p^T P = p^T$, can be approximated by simulating the associated Markov chain for a long run and computing the frequency of visiting each state. Test this[1] for the matrix

$$P = \begin{pmatrix} 0.35-\alpha & 0.17 & 0.48 & 0 & 0 & \alpha \\ 0.46 & 0.47 & 0.07 & 0 & 0 & 0 \\ 0.35 & 0.13 & 0.52 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.34 & 0.48 & 0.18 \\ 0 & 0 & 0 & 0.13 & 0.42 & 0.45 \\ \beta & 0 & 0 & 0.41-\beta & 0.33 & 0.26 \end{pmatrix},$$

where $\alpha = 0.01$, $\beta = 0.01$.

(a) For this, simulate the chain for 1000 steps, and compare the vector of relative frequencies of visits with the stationary probability vector p.

(b) What do you observe by increasing the values of $\alpha \to 0.3$, $\beta \to 0.4$?

Exercise 4 (Coin Flipping Revisited)
Recall Question 4 of Sheet 4.

(a) Write a Matlab function to simulate the coin-flipping experiment with 10 coins by randomly choosing numbers between 1 and 10 and sequentially flipping the appropriately numbered coins. You may use the Matlab function `randi(10)` to generate a random integer between 1 and 10. Simulate the experiment for a large number of steps and estimate the frequency of the "all heads" configuration. Repeat several times to see if you get consistent results.

(b) Extend the above code to estimate the average return time to the "all heads" configuration.

(c) How do you answers to (a) and (b) compare to your theoretical answers from Question 4, Sheet 4?

[1] If you wish, you may use the Matlab program given on the next page to simulate a Markov chain of the desired length.

```
function path = simulate_MC(P,path_len,start)
% function path = simulate_MC(P,path_len,start)
%
% Simulates a path of length 'path_len' of the Markov chain given by the
% transition matrix 'P', starting at node 'start' (an element of
% 1:length(P)), or, if not specified, from a random node. The simulated
% path is returned in 'path'.
%
% (c) 2015 Peter Koltai

% size of state space
n = length(P);

% cumulative row sum and random numbers - precomputed for the simulation
SP = cumsum(P,2);
rnums = rand(path_len,1);

% initialize output
path = zeros(1,path_len);

% starting state
if nargin < 3
    curr = ceil(rand()*n);
else
    curr = start;
end

% loop generating trajectory
for k=1:path_len
    % current state of the process
    path(k) = curr;
    % destination of the next jump
    curr = find(rnums(k) < SP(curr,:),1);
end
```