ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

High Performance Computing for Science and Engineering II

Spring semester 2017

Dr. P. Hadjidoukas, Prof. C. Papadimitriou
ETH Zentrum, CLT E 13
CH-8092 Zürich

Project # 5

Issued: 15.05.2017

- In these scripts, we outline five computational engineering problems. Choose one and work either individually or in a group of two people. Communication is allowed to the extent that you do not copy the work of other groups.
- You are encouraged to contact one of the TAs in order to arrange a meeting to discuss your chosen project. These meetings are meant for clarifying and detailing the projects, give early feedback on your approach, *not* for correcting your code.
- In evaluating your work, we will consider your ability to analyze the problem and the hardware at your disposal, to appropriately apply the principles taught during the whole HPCSE class, and to report your reasoning and findings.
- The report (including text, code, figures) needs to be emailed before the day of the exam. If working in a group of two, each student has to write an *individual* report.

Optimal sensor placement

Optimal sensor placement: Detection of vortex locations using sensors along a wall

Fish have the ability to detect the flow field around them using a specialized sensory organ called *lateral line* [BZ09]. The lateral line consists of cells that measure the pressure and the tangential velocity on the fish skin. Using this information, fish are able to detect objects, find prey and school with other fish.

In this project, you will work with a simplified model of the lateral line. A number of sensors for tangential velocity detection is arranged along a wall bounding a region of fluid. At some location inside the fluid there is a vortex, i.e., a region of rotating fluid. This can be thought of as a simplification of the disturbance that a swimming body, e.g. a small prey fish generates. By combining prior information about possible vortex locations with the recorded sensor measurements it is possible to find an estimate for the location of the vortex. Your task is to optimize the arrangement of sensors along the wall so that you maximize the expected gain of information about the location of the vortex (information is gained by looking at the sensor measurements).

Setting

The flow model is the following. We use a two-dimensional setting. Along the x-axis, there is an infinitely long straight wall. The region above the wall is filled with incompressible, inviscid fluid. The vortex has strength Γ and coordinates (x_v, y_v) and is located somewhere inside the box $[x_l, x_u] \times [y_l, y_u]$. There are M equispaced sensors on the wall, where the first one has coordinates $(x_s, 0)$ and the distance between sensors is h. See Figure 1 for an illustration.

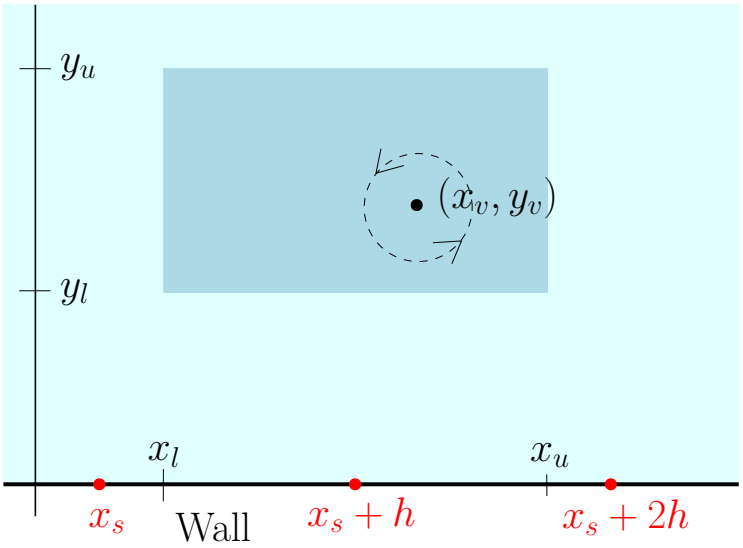

Figure 1: The problem setting with three sensors along the wall

In an incompressible inviscid fluid (potential flow model), the time-stationary flow induced by a point vortex at $(0, 0)$ is radially symmetric and in polar coordinates given by

$$u(r, \alpha) = \begin{pmatrix} 0 \\ \frac{\Gamma}{2\pi r} \end{pmatrix},$$

i.e., no velocity in radial direction and velocity $u_{\text{cr}} = \frac{\Gamma}{2\pi r}$ in cross-radial direction (tangential to a circle around the origin).

Now let the vortex have Cartesian coordinates (x_v, y_v). A sensor at $(x_s, 0)$ measures only the velocity tangential to the wall. Let α denote the angle between the shortest line from the vortex to the wall and a line connecting vortex and sensor, as detailed in Figure 2. Then the value recorded by the sensor is

$$z = u_{\text{cr}} \cos(\alpha) = \frac{\Gamma}{2\pi r} \frac{y_v}{r} = \frac{\Gamma y_v}{2\pi((x_s - x_v)^2 + y_v^2)}.$$

For the ith sensor with coordinates $(x_s + (i-1)h, 0)$, $i = 1, \ldots, M$ the recorded value is

$$u_i = \frac{\Gamma y_v}{2\pi((x_s + (i-1)h - x_v)^2 + y_v^2)}. \tag{1}$$

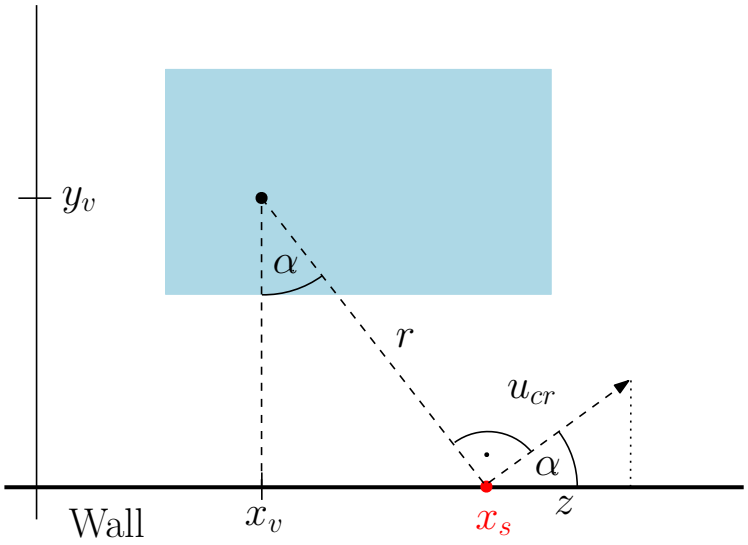

Figure 2: Measuring tangential velocity along the wall

Method description

Optimal sensor placement belongs to the broad topic of Bayesian experimental design. Because conducting experiments (as well as simulations) is expensive and time-consuming[1], the goal is to optimize the experimental conditions beforehand so that the usefulness of the experiment (i.e., the information gained by the experimental results) is maximized.

The aim of this project is to place a number of sensors in an optimal way on a wall in order to gain the most information possible about the location of a nearby disturbance. The method is detailed below. We will use the Nested Sampling technique by Huan and Marzouk [HM13].

Setting. We will look at the following quantities:

- We only consider the case of evenly spaced sensors; therefore the sensor locations are uniquely described by the position of the first sensor along the wall and the spacing between sensors. Define the design parameters $d = \big(x_s, h\big)$ describing the sensor locations along the wall.
- The quantity that we want to infer is the location of the vortex inside the fluid. Define the uncertain parameters of interest $\theta = \big(x_v, y_v\big)$ consisting of the coordinates of the vortex.
- Let M be the number of sensors. The sensors record the tangential velocity along the wall. Denote the measurements with $y \in \mathbb{R}^M$. We will take into account measurement error.

Let $g(\theta; d)$ denote the tangential velocity at sensor locations defined by d for a flow generated by a vortex located at location θ, i.e.

$$g(\theta; d) = \big(u_i(\theta, d)\big)_{i=1,\dots,M}$$

[1] In this project, the model is given by an analytical expression (1) that is cheap to evaluate. However, this is a simplified model for vortex-induced flow and might be replaced by a more expensive and more accurate one, e.g. by a two-dimensional fluid solver.

with u_i as in Equation (1). We assume that the sensor measurements are corrupted by jointly Gaussian noise with zero mean and covariance matrix Σ. This yields the model

$$y = g(\theta; d) + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \Sigma). \tag{2}$$

It follows that $y|\theta, d \sim \mathcal{N}(g(\theta; d), \Sigma)$. Assume we have a prior distribution $p(\theta)$ for the vortex location. Then the posterior for θ is given by

$$\begin{aligned} p(\theta|y,d) &= \frac{p(y|\theta,d)p(\theta|d)}{p(y|d)} \\ &= \frac{p(\theta)}{p(y|d)}(2\pi)^{-\frac{M}{2}} \det(\Sigma)^{-1} \exp\left(-\frac{1}{2}\left(y - g(\theta;d)\right)^T \Sigma^{-1} \left(y - g(\theta;d)\right)\right). \end{aligned}$$

Objective function: Expected information gain. Now we will derive the objective function for optimal experimental design, i.e. the function that must be maximized in order to obtain the best sensor arrangement. We follow the approach detailed in [HM13].

First, we define the *Kullback-Leibler divergence*

$$D_{KL}\big(p\|q\big) = \int p(x) \log\left(\frac{p(x)}{q(x)}\right) \, \mathrm{d}x.$$

This expression is a measure of the difference between two probability distributions p and q. The Kullback-Leibler divergence between prior and posterior distribution $D_{KL}\big(p(\theta|y,d)\|p(\theta)\big)$ measures how much the inclusion of measurements y changes the distribution of θ, i.e. how much information we gain by performing the experiment under conditions d with outcome y. Because we cannot know the outcome of the experiment before we perform it, we define the *expected information gain* by

$$\begin{aligned} U(d) &= \int D_{KL}\big(p(\theta|y,d)\|p(\theta)\big) \; p(y|d) \; \mathrm{d}y \\ &= \int\int \log\left(\frac{p(\theta|y,d)}{p(\theta)}\right) \; p(\theta|y,d) \; p(y|d) \; \mathrm{d}\theta \; \mathrm{d}y. \end{aligned}$$

(This is the standard Bayesian way of dealing with unknown quantities: just integrate them out.) Using Bayes' theorem, we can rewrite the expected information gain into

$$U(d) = \int\int \big(\log p(y|\theta,d) - \log p(y|d)\big) \; p(y|\theta,d) \; p(\theta) \; \mathrm{d}\theta \; \mathrm{d}y.$$

The optimal design d^* is now given by

$$d^* = \arg\min_d U(d).$$

Numerical Approximation: Nested sampling. The function $U(d)$ cannot be computed analytically. We approximate it numerically using Monte Carlo integration:

$$U(d) \approx \frac{1}{n} \sum_{i=1}^{n} \Big(\log p(y^{(i)}|\theta^{(i)},d) - \log p(y^{(i)}|d)\Big),$$

where n denotes the number of samples, $\theta^{(i)}$ are samples from the prior $p(\theta)$ and $y^{(i)}$ are samples from the likelihood $p(y|\theta = \theta^{(i)}, d)$. The evidence $p(y^{(i)}|d)$ must be computed by yet another Monte Carlo integration:

$$\begin{aligned} p(y^{(i)}|d) &= \int p(y^{(i)}, \theta|d) \ \mathrm{d}\theta \\ &= \int p(y^{(i)}|\theta, d)p(\theta) \ \mathrm{d}\theta \\ &\approx \frac{1}{m}\sum_{j=1}^{m} p(y^{(i)}|\theta^{(i,j)}, d), \end{aligned}$$

where $\theta^{(i,j)}$ are again samples from the prior $p(\theta)$.

In principle, we would need to draw $n + nm$ samples from $p(\theta)$ and evaluate the model $g(\theta; d)$ for every such sample. This is very costly. Therefore, we reuse the samples: We set $n = m$, draw n samples $\hat{\theta}^{(k)} \sim p(\theta)$ and set $\theta^{(i)} = \hat{\theta}^{(i)}, \theta^{(i,j)} = \hat{\theta}^{(j)}$.

Optimization: CMA-ES. The expected information gain is given by

$$U(d) \approx \frac{1}{n}\sum_{i=1}^{n}\left(\log p(y^{(i)}|\hat{\theta}^{(i)}, d) - \log\left(\frac{1}{n}\sum_{j=1}^{n} p(y^{(i)}|\hat{\theta}^{(j)}, d)\right)\right)$$

where $\hat{\theta}^{(k)} \sim p(\theta)$, $y^{(k)} \sim p(y|\theta = \hat{\theta}^{(k)}, d)$.

During minimization, this function needs to be evaluated for every design d that we want to try. Of course, it is too expensive to do new sampling for each evaluation. Therefore, we draw the samples before we start the optimization, so that during the optimization procedure we only need to evaluate $U(d)$ using the given samples. This also has the advantage that the resulting $U(d)$ is smooth and does not show random fluctuations caused by different sample sets.

The precise shape of $U(d)$ and in particular its optimum depend on the samples used. To improve robustness, we increase the number of samples in the following way.

Let $\hat{\theta}^{(k)} \sim p(\theta)$, $k = 1, \ldots, n$. To sample y from $p(y|\theta = \hat{\theta}^{(k)}, d)$, we need to evaluate $g(\hat{\theta}^{(k)}; d)$ and add normally distributed error as in Equation (2). Evaluating the model is in general expensive (see above footnote) while generating normally distributed random numbers is cheap. Following this reasoning, we form N_{util} utility functions $U_i(d)$ by using the same samples $\hat{\theta}^{(k)}, k = 1, \ldots, n$ for each i, but redrawing the measurements $y^{(k,i)} \sim p(y|\theta = \hat{\theta}^{(k)}, d)$. This is cheap because we don't need to recompute the model evaluations. Then we define the *extended utility function* by

$$\tilde{U}(d) = \frac{1}{N_{\text{util}}}\sum_{i=1}^{N_{\text{util}}} U_i(d)$$

where

$$U_i(d) = \frac{1}{n}\sum_{k=1}^{n}\left(\log p(y^{(k,i)}|\hat{\theta}^{(k)}, d) - \log\left(\frac{1}{n}\sum_{j=1}^{n} p(y^{(k,i)}|\hat{\theta}^{(j)}, d)\right)\right).$$

For optimization, we use the algorithm CMA-ES that is suited for noisy blackbox objective functions where no gradient information is available.

Problem setup

A prototype serial implementation of this problem in C is provided[2]. Your goal is to develop high-performance implementations of the code by parallelizing the computation of $U(d)$ and the optimization algorithm CMA-ES. The problem size can increase by modifying the number of sample and computations of utility (`N_samples` and `N_u` in `utility/fitfun_extended.c`, respectively) and the population size (`lambda`) of CMA-ES.

References

[BZ09] Horst Bleckmann and Randy Zelick. Lateral line system of fish. *Integrative zoology*, 4(1):13–25, 2009.

[HM13] Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013.

[2]Reference implementation: `http://cse-lab.ethz.ch/images/teaching/HPCSEII_FS2017/OSP.zip`