Optimal sensor location for distributed parameter system identification (Part 1)

Dariusz Uciński

Institute of Control and Computation Engineering
University of Zielona Góra
Structure of the minicourse

- **Tuesday, 27 October, 13:30**: introduction to OED; sensor location problem for parameter estimation of PDEs; adaptation of modern OED theory to the sensor location setting
- **Friday, 30 October, 11:30**: sensor selection for large-scale monitoring networks; overcoming the curse of dimensionality; design for scanning observations
- **Friday, 6 November, 11:30**: design of mobile sensor trajectories; robust design; design with correlated observations; Bayesian approach; design for ill-posed problems
Schedule of Part 1

1. Introduction to optimum design
   - Linear regression
   - Optimum design

2. OED and optimal sensor location
   - Problem and its specificities
   - Adaptation of optimum design methodology
**Example:** For the following observations

<table>
<thead>
<tr>
<th>$x_i$</th>
<th>8</th>
<th>4</th>
<th>5</th>
<th>−1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y_i$</td>
<td>−2</td>
<td>0</td>
<td>2</td>
<td>6</td>
</tr>
</tbody>
</table>

we wish to find an equation describing best the linear trend:
We approximate the response $y$ by the linear model

$$\hat{y} = a_1 x + a_0$$

so as to minimize the individual prediction errors (residuals)

$$e_i = y_i - \hat{y}_i = y_i - a_0 - a_1 x_i$$
Least-squares criterion

\[ S_r = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2 \rightarrow \min \]
Least-squares criterion

The optimality conditions

\[
\frac{\partial S_r}{\partial a_0} = -2 \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i) = 0
\]

\[
\frac{\partial S_r}{\partial a_1} = -2 \sum_{i=1}^{n} [(y_i - a_0 - a_1 x_i)x_i] = 0
\]

yield the system of linear equations

\[
\begin{cases}
0 = \sum y_i - \sum a_0 - \sum a_1 x_i \\
0 = \sum y_i x_i - \sum a_0 x_i - \sum a_1 x_i^2
\end{cases}
\]
After rearranging, we get the so-called normal equations:

\[
\begin{align*}
na_0 + \left( \sum x_i \right) a_1 &= \sum y_i \\
\left( \sum x_i \right) a_0 + \left( \sum x_i^2 \right) a_1 &= \sum x_i y_i
\end{align*}
\]
After rearranging, we get the so-called *normal equations*:

\[
\begin{align*}
na_0 + \left( \sum x_i \right) a_1 &= \sum y_i \\
\left( \sum x_i \right) a_0 + \left( \sum x_i^2 \right) a_1 &= \sum x_i y_i
\end{align*}
\]

or, in matrix form,

\[
F^T F \theta = F^T y
\]

where

\[
F = \begin{bmatrix}
1 & x_1 \\
\vdots & \vdots \\
1 & x_n
\end{bmatrix}, \quad \theta = \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}
\]
Linear regression

Its solution is $\hat{\theta} = (F^T F)^{-1} F^T y$. Specifically

$$\hat{a}_1 = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2}$$

$$\hat{a}_0 = \bar{y} - \hat{a}_1 \bar{x}$$

Example (ctd\text{'}):

$$a_1 = \frac{4 \cdot (-12) - 16 \cdot 6}{4 \cdot 106 - 16^2} = -0.857$$

$$a_0 = 1.5 - (-0.857)(4) = 4.929$$

$$\hat{y} = -0.857x + 4.929$$
Polynomial regression

Given data, fit a parabola:

\[ \hat{y} = a_0 + a_1 x + a_2 x^2 \]
Polynomial regression

Given data, fit a parabola:

$$\hat{y} = a_0 + a_1x + a_2x^2$$

Sum of squared residuals:

$$S_r(\theta) = \sum_{i=1}^{n} (y_i - a_0 - a_1x_i - a_2x_i^2)^2 = \|y - F\theta\|^2$$

where

$$F = \begin{bmatrix} 1 & x_1 & x_1^2 \\
1 & x_2 & x_2^2 \\
\vdots & \vdots & \vdots \\
1 & x_n & x_n^2 \end{bmatrix}, \quad \theta = \begin{bmatrix} a_0 \\
a_1 \\
a_2 \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\
y_2 \\
\vdots \\
y_n \end{bmatrix}$$
Polynomial regression

Given data, fit a parabola:

\[ \hat{y} = a_0 + a_1 x + a_2 x^2 \]

Sum of squared residuals:

\[ S_r(\theta) = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i - a_2 x_i^2)^2 = \| y - F \theta \|^2 \]

where

\[ F = \begin{bmatrix} 1 & x_1 & x_1^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}, \quad \theta = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \]

Optimality conditions are the normal equations again:

\[ (F^T F) \theta = F^T y \]
Exlicitly, the normal equations are

\[
\begin{align*}
(n)a_0 + \left( \sum x_i \right) a_1 + \left( \sum x_i^2 \right) a_2 &= \sum y_i \\
\left( \sum x_i \right) a_0 + \left( \sum x_i^2 \right) a_1 + \left( \sum x_i^3 \right) a_2 &= \sum x_i y_i \\
\left( \sum x_i^2 \right) a_0 + \left( \sum x_i^3 \right) a_1 + \left( \sum x_i^4 \right) a_2 &= \sum x_i^2 y_i
\end{align*}
\]

It is easy to generalize the approach to

\[\hat{y} = a_0 f_0(x) + a_1 f_1(x) + a_2 f_2(x) + \cdots + a_m f_m(x)\]

where \(f_0, f_1, \ldots, f_m\) are linearly independent functions.

**Question**

How much does measurement noise influence the accuracy of the estimates? Is it possible to reduce this influence?
The weights of objects $A$ and $B$ are to be measured using a pan balance and a set of standard weights. Each weighing measures the weight difference between objects placed in the left pan vs. any objects placed in the right pan by adding calibrated weights to the lighter pan until the balance is in equilibrium.
Each measurement has a random error. The average error is zero; the standard deviations of the probability distribution of the errors is the same number $\sigma$ on different weighings; and errors on different weighings are independent. Denote by $m_A$ and $m_B$ the true weights.
Weighing two objects

First, we object $A$ in one pan, with the other pan empty.
Weighing two objects

Let $y_1$ be its measured weight. As the error in this result is $\sigma$, we have

$$m_A = y_1 \pm \sigma$$

For object $B$, we perform the second measurement and get

$$m_B = y_2 \pm \sigma$$

For example, if $y_1$ and $y_2$ are 10 and 25 grammes, respectively, and $\sigma$ is 0.1 gramme, we have

$$m_A = 10 \pm 0.1 \text{ grammes}$$
$$m_B = 25 \pm 0.1 \text{ grammes}$$
We can express these results using the matrix notation

\[ y = \mathbf{F}\theta + \epsilon \]

where

\[ y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} +1 & 0 \\ 0 & +1 \end{bmatrix}, \quad \theta = \begin{bmatrix} m_A \\ m_B \end{bmatrix}, \quad \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix} \]

Note that we have \( \det(\mathbf{F}^T\mathbf{F}) = 1 \).
Weighing two objects

But consider another strategy with the same experimental effort (i.e., two measurements):

![Diagram showing two weighing objects and two sets of measurements](image)
Weighing two objects

Neglecting measurement errors, we get

\[\begin{align*}
+ m_A + m_B &= y_1 \\
- m_A + m_B &= y_2
\end{align*}\]

\[\Rightarrow \begin{align*}
m_A &= \frac{1}{2} (y_1 - y_2) \\
m_B &= \frac{1}{2} (y_1 + y_2)
\end{align*}\]
Weighing two objects

Neglecting measurement errors, we get

\[ \begin{cases} 
+ m_A + m_B = y_1 \\
- m_A + m_B = y_2 \end{cases} \Rightarrow \begin{cases} 
m_A = \frac{1}{2} (y_1 - y_2) \\
m_B = \frac{1}{2} (y_1 + y_2) \end{cases} \]

What is the error for these estimates?

\[
\text{var}(m_A) = \frac{1}{4} \left[ \text{var}(y_1) + \text{var}(y_2) \right] = \frac{1}{4} \left[ \sigma^2 + \sigma^2 \right] = \frac{1}{2} \sigma^2
\]

It was reduced from \( \sigma \) to \( \sigma/\sqrt{2} \):

\[
m_A = 10 \pm 0.07 \text{ grammes} \\
m_B = 25 \pm 0.07 \text{ grammes}
\]
Additionally, observe that now

$$F = \begin{bmatrix} +1 & +1 \\ -1 & +1 \end{bmatrix}$$

and $\det(F^T F) = 4$. 
Weighing four objects: Strategy 1

Consider now weighing four objects $A$, $B$, $C$ and $D$. First, we weigh each of them individually and get

\[
m_A = y_1 \pm \sigma \\
m_B = y_2 \pm \sigma \\
m_C = y_3 \pm \sigma \\
m_D = y_4 \pm \sigma
\]

Suppose that $y_1 = 10$, $y_2 = 25$, $y_3 = 15$ and $y_4 = 30$. Observe that

\[
F = \begin{bmatrix} +1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & +1 \end{bmatrix}
\]

and $\det(F^T F) = 1$. 
Weighing four objects: Strategy 2

But consider another strategy:

- Strategy 2:
  1. Weigh A and B, C and D.
  2. If equal, weigh B and D.
  3. If equal, weigh A and C.
  4. If equal, C is the heaviest.
  5. If B and D are not equal, C is equal to the lighter of B and D.

Neglecting measurement errors, we get

\[
\begin{align*}
+m_A + m_B &= y_1 \\
-m_A + m_B &= y_2 \\
+m_C + m_D &= y_3 \\
-m_C + m_D &= y_4
\end{align*}
\]

\[
\Rightarrow \quad \begin{align*}
m_A &= \frac{1}{2} (y_1 - y_2) \\
m_B &= \frac{1}{2} (y_1 + y_2) \\
m_C &= \frac{1}{2} (y_3 - y_4) \\
m_D &= \frac{1}{2} (y_3 + y_4)
\end{align*}
\]
Weighing four objects: Strategy 2

What is the error for these estimates?

\[
\text{var}(m_A) = \frac{1}{4} \left[ \text{var}(y_1) + \text{var}(y_2) \right] \\
= \frac{1}{4} \left[ \sigma^2 + \sigma^2 \right] = \frac{1}{2} \sigma^2
\]

It was reduced from \( \sigma \) to \( \sigma / \sqrt{2} \):

- \( m_A = 10 \pm 0.07 \) grammes
- \( m_B = 25 \pm 0.07 \) grammes
- \( m_C = 15 \pm 0.07 \) grammes
- \( m_D = 30 \pm 0.07 \) grammes
We have

\[
\begin{bmatrix}
  y_1 \\
  y_2 \\
  y_3 \\
  y_4
\end{bmatrix} = \begin{bmatrix}
  +1 & +1 & 0 & 0 \\
  -1 & +1 & 0 & 0 \\
  0 & 0 & +1 & +1 \\
  0 & 0 & -1 & +1
\end{bmatrix} \begin{bmatrix}
  m_A \\
  m_B \\
  m_C \\
  m_D
\end{bmatrix} + \begin{bmatrix}
  \epsilon_1 \\
  \epsilon_2 \\
  \epsilon_3 \\
  \epsilon_4
\end{bmatrix}
\]

Additionally, observe that \( \det(F^T F) = 16 \).
But consider yet another strategy:

A B C D

80

A B

10

C D

A C

30

B D

A D

B C
Weighing four objects: Strategy 3

Neglecting measurement errors, we get

\[ + m_A + m_B + m_C + m_D = y_1 \]
\[ - m_A + m_B - m_C + m_D = y_2 \]
\[ - m_A - m_B + m_C + m_D = y_3 \]
\[ - m_A + m_B + m_C - m_D = y_4 \]

Consequently,

\[ m_A = \frac{1}{4} (y_1 - y_2 - y_3 - y_4) \]
\[ m_B = \frac{1}{4} (y_1 + y_2 - y_3 + y_4) \]
\[ m_C = \frac{1}{4} (y_1 - y_2 + y_3 + y_4) \]
\[ m_D = \frac{1}{4} (y_1 + y_2 + y_3 - y_4) \]
Weighing four objects: Strategy 3

What is the error for these estimates?

\[
\text{var}(m_A) = \frac{1}{16} \left[ \text{var}(y_1) + \text{var}(y_2) + \text{var}(y_3) + \text{var}(y_4) \right] \\
= \frac{1}{16} \left[ \sigma^2 + \sigma^2 + \sigma^2 + \sigma^2 \right] = \frac{1}{4} \sigma^2
\]

It was further reduced from \( \sigma/\sqrt{2} \) to \( \sigma/2 \):

\[
\begin{align*}
m_A &= 10 \pm 0.05 \text{ grammes} \\
m_B &= 25 \pm 0.05 \text{ grammes} \\
m_C &= 15 \pm 0.05 \text{ grammes} \\
m_D &= 30 \pm 0.05 \text{ grammes}
\end{align*}
\]

It can be shown that this cannot be improved.
We have

\[
\begin{bmatrix}
  y_1 \\
  y_2 \\
  y_3 \\
  y_4
\end{bmatrix}
= \begin{bmatrix}
  +1 & +1 & +1 & +1 \\
  -1 & +1 & -1 & +1 \\
  -1 & -1 & +1 & +1 \\
  -1 & +1 & +1 & -1
\end{bmatrix}
\begin{bmatrix}
  m_A \\
  m_B \\
  m_C \\
  m_D
\end{bmatrix}
+ \begin{bmatrix}
  \epsilon_1 \\
  \epsilon_2 \\
  \epsilon_3 \\
  \epsilon_4
\end{bmatrix}
\]

Additionally, observe that \( \det(F^TF) = 256 \).
We have

\[
\begin{bmatrix}
  y_1 \\
y_2 \\
y_3 \\
y_4
\end{bmatrix}
= \begin{bmatrix}
  +1 & +1 & +1 & +1 \\
  -1 & +1 & -1 & +1 \\
  -1 & -1 & +1 & +1 \\
  -1 & +1 & +1 & -1
\end{bmatrix}
\begin{bmatrix}
m_A \\
m_B \\
m_C \\
m_D
\end{bmatrix}
+ \begin{bmatrix}
  \epsilon_1 \\
  \epsilon_2 \\
  \epsilon_3 \\
  \epsilon_4
\end{bmatrix}
\]

Additionally, observe that \(\text{det}(F^T F) = 256\).

Better precision coincides with higher values of \(\text{det}(F^T F)\). The matrix \(M = F^T F\) is called the Fisher information matrix and experimental conditions maximizing its determinant are called D-optimal.
D-optimum designs for simple linear regression

Assume that the response is the form

\[ y = a_0 + a_1 x \]

We would like to make two measurements to estimate \( a_0 \) and \( a_1 \), but we can select the corresponding settings of \( x = x_1 \) and \( x = x_2 \) before the experiment. If they are to belong to the interval \([-1, 1]\), how can we choose them best?
D-optimum designs for simple linear regression

Assume that the response is the form

\[ y = a_0 + a_1 x \]

We would like to make two measurements to estimate \( a_0 \) and \( a_1 \), but we can select the corresponding settings of \( x = x_1 \) and \( x = x_2 \) before the experiment. If they are to belong to the interval \([-1, 1]\), how can we choose them best?

**Answer:** We are going to get observations

\[ y_1 \approx a_0 + a_1 x_1 \]
\[ y_2 \approx a_0 + a_1 x_2 \]
D-optimum designs for simple linear regression

In matrix form, this is

\[
\begin{bmatrix}
  y_1 \\
  y_2
\end{bmatrix} =
\underbrace{egin{bmatrix}
  1 & x_1 \\
  1 & x_2
\end{bmatrix}}_{F} \begin{bmatrix}
  a_0 \\
  a_1
\end{bmatrix} +
\begin{bmatrix}
  \epsilon_1 \\
  \epsilon_2
\end{bmatrix}
\]
D-optimum designs for simple linear regression

In matrix form, this is

\[
\begin{bmatrix}
y_1 \\
y_2
\end{bmatrix} = \begin{bmatrix}
1 & x_1 \\
1 & x_2
\end{bmatrix} \begin{bmatrix}
a_0 \\
a_1
\end{bmatrix} + \begin{bmatrix}
\epsilon_1 \\
\epsilon_2
\end{bmatrix}
\]

Adopting the D-optimality criterion as was for the weighing example, we get

\[
\det(F^T F) = \det\left( \begin{bmatrix}
1 & 1 \\
x_1 & x_2
\end{bmatrix} \begin{bmatrix}
1 & x_1 \\
1 & x_2
\end{bmatrix} \right)
\]

\[
= \det\left( \begin{bmatrix}
2 & x_1 + x_2 \\
x_1 + x_2 & x_1^2 + x_2^2
\end{bmatrix} \right) = (x_1 - x_2)^2
\]

It is maximized for \( x_1 = -1 \) and \( x_2 = 1 \) (or \( x_1 = 1 \) and \( x_2 = -1 \)), i.e., when the measurements are at the opposite ends of the design interval.
We wish to estimate resistance \( R \) based on 10 measurements of the current (it may vary between 10 A and 20 A) and the corresponding voltage.
Why is the information matrix so important?

Assume that the observations are modelled by

$$y = F\theta + \varepsilon$$

where $\varepsilon$ is measurement noise satisfying

$$E[\varepsilon] = 0, \quad \text{cov}(\varepsilon) = E[\varepsilon\varepsilon^T] = \sigma^2 I$$

and $\sigma^2$ is a constant noise variance.
Why is the information matrix so important?

Assume that the observations are modelled by

\[ y = F\theta + \varepsilon \]

where \( \varepsilon \) is measurement noise satisfying

\[ E[\varepsilon] = 0, \quad \text{cov}(\varepsilon) = E[\varepsilon\varepsilon^T] = \sigma^2 I \]

and \( \sigma^2 \) is a constant noise variance.

The parameter estimate \( \hat{\theta} = (F^TF)^{-1}F^Ty \) is then a random vector and

\[ E[\hat{\theta}] = \theta \quad \text{(unbiasedness)} \]
\[ \text{cov}(\hat{\theta}) = \sigma^2 (F^TF)^{-1} \quad \text{(dispersion around } \theta \text{)} \]
Uncertainty ellipsoid
Uncertainty ellipsoid
Uncertainty ellipsoid
Uncertainty ellipsoid
Uncertainty ellipsoid
Uncertainty ellipsoid
Setting $\mathbf{M} = \mathbf{F}^T \mathbf{F}$, we define

- **D-optimality** criterion:
  
  $$\Psi(\mathbf{M}) = \log \det(\mathbf{M}^{-1}) = -\log \det(\mathbf{M})$$
  (the volume of the uncertainty ellipsoid is minimized)
Setting $\mathbf{M} = \mathbf{F}^T \mathbf{F}$, we define

- **D-optimality** criterion:
  $$\Psi(\mathbf{M}) = \log \det(\mathbf{M}^{-1}) = -\log \det(\mathbf{M})$$
  (the volume of the uncertainty ellipsoid is minimized)

- **A-optimality** criterion:
  $$\Psi(\mathbf{M}) = \text{trace}(\mathbf{M}^{-1})$$
  (the mean axis length of the uncertainty ellipsoid is minimized)
Optimality criteria

Setting $\mathbf{M} = \mathbf{F}^T \mathbf{F}$, we define

- **D-optimality** criterion:
  \[
  \psi(\mathbf{M}) = \log \det(\mathbf{M}^{-1}) = -\log \det(\mathbf{M}) \quad \text{(the volume of the uncertainty ellipsoid is minimized)}
  \]

- **A-optimality** criterion: $\psi(\mathbf{M}) = \text{trace}(\mathbf{M}^{-1}) \quad \text{(the mean axis length of the uncertainty ellipsoid is minimized)}$

- **E-optimality** criterion: $\psi(\mathbf{M}) = \lambda_{\max}(\mathbf{M}^{-1}) \quad \text{(the length of the largest axis of the uncertainty ellipsoid is minimized)}$

*Notation*: $\lambda_{\max}(\cdot)$ is the maximum eigenvalue of its matrix argument.

Most often, they yield different solutions!
1940–1950: development of factorial experiments and their applications in industry (mainly chemical engineering)

1950s: beginning of optimal experiment design (Kiefer-Wolfowitz theorem)

1969–1970: first algorithms for numerical construction of optimum designs (V. Fedorov and H. Wynn)

1970 up to now: optimal input signals for parameter estimation in systems described by ODEs (R. Mehra)

1980s up to now: optimal sensor placement (1981) and input signal design (1983) for PDEs

All the above research directions are still far from being complete.
Workshop on *Experiments for Processes With Time or Space Dynamics*, Isaac Newton Institute for Mathematical Sciences, Cambridge, UK, 18–22 June 2011, organizers: Dariusz Uciński and Andrew Curtis (University of Edinburgh)

[www.newton.ac.uk/programmes/DAE/daew01.html](http://www.newton.ac.uk/programmes/DAE/daew01.html)
Introduction to optimum design
OED and optimal sensor location

Workshops

Contributions to Statistics
Dariusz Uciński
Anthony C. Atkinson
Maciej Patan *Editors*

mODa 10 – Advances in Model-Oriented Design and Analysis
Proceedings of the 10th International Workshop in Model-Oriented Design and Analysis
Held in Łagów Lubuski, Poland,
June 10–14, 2013
Complements to this minicourse

Optimal Measurement Methods for Distributed Parameter System Identification
Dariusz Ucinski
CRC Press, 2005

Optimal Sensor Networks Scheduling in Identification of Distributed Parameter Systems
Maciej Patan
Springer-Verlag, 2012
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example: Heating a metal block with a crack
Example ctd': Mathematical model

Let $y(x, t)$ be the temperature at spatial point $x$ and time $t$. Its evolution is governed by the partial differential equation (PDE)

$$\frac{\partial y}{\partial t} = \theta \Delta y \quad \text{for } x \in \Omega \text{ and } t \in [0, t_f]$$

$\theta$ being a parameter, subject to the initial condition

$$y = 0 \quad \text{in } \Omega \text{ at } t = 0$$

and the boundary conditions

$$y = 100 \quad \text{on the left side of } \Omega \quad \text{(Dirichlet condition)}$$

$$\frac{\partial y}{\partial n} = -10 \quad \text{on the right side of } \Omega \quad \text{(Neumann condition)}$$

$$\frac{\partial y}{\partial n} = 0 \quad \text{on all other boundaries} \quad \text{(Neumann condition)}$$
If thermal conductivity $\theta$ is unknown, which is rather typical, we could observe the temperature evolution at a fixed point and tune $\theta$ so that our model fits the data as best as it can.
If thermal conductivity $\theta$ is unknown, which is rather typical, we could observe the temperature evolution at a fixed point and tune $\theta$ so that our model fits the data as best as it can.

**Problem**

Where to place the pyrometer so as to get the most valuable information about $\theta$?

Such situations (PDEs and related sensor location problems) are common in engineering practice.
Other motivating examples

- environment observation and forecasting systems
  - air pollution monitoring,
  - groundwater resources management and river monitoring,
  - military applications: surveillance and inspection in hazardous environments,
- fault detection and isolation,
- emerging smart materials,
- intelligent building monitoring,
- habitat monitoring,
- intelligent traffic systems,
- computer-assisted tomography,
- recovery of valuable minerals and hydrocarbon from underground permeable reservoirs,
- and many more . . .
Spatiotemporal dynamics

**Distributed parameter system**—dynamic system whose state depends on both time and space; its model (a PDE) is known up to a vector of unknown parameters $\theta$.

**Observations**—using sensors in order to estimate $\theta$. 
Problem: How to choose optimal sensor locations?
Strategies of taking measurements (Option 1)

Stationary sensors

Spatial area $\Omega$

Sensors

$x^1$

$x^2$

$x^3$

How to determine the best sensor configuration?
Strategies of taking measurements (Option 2)

Scanning

Sensors activated at time $t$

Area $\Omega$

How to select optimal locations at given time instants?
Strategies of taking measurements (Option 3)

Mobile sensors

Spatial area $\Omega$

Starting positions

Trajectories

How to design optimal trajectories?
Enabling technology: Sensor Networks (SNs)

Collection of spatially distributed inexpensive, small devices equipped with sensors and interconnected by a communication network, exchanging and processing data to cooperatively monitor physical or environmental conditions (e.g., temperature, sound, vibration, pressure, etc.).
Motes must be **low cost, low power** (for long-term operation), **automated** (maintenance free), **robust** (to withstand errors and failures), and **non-intrusive**.

- **Hardware**: a microprocessor, data storage, sensors, AD-converters, a data transceiver (Bluetooth), controllers, and an energy source
- **Software**: TinyOS

They are already manufactured (Crossbow, Intel).
Applications: from battle field...
Applications: ... to our life

Smart Homes/Spaces
Automated Highways
All this and more...
Existing approaches to optimal sensor location

- Conversion to a non-linear problem of state estimation (Malebranche, 1988; Korbicz et al., 1988);
- Employing random fields analysis (Kazimierczyk, 1989; Sun, 1994);
Conversion to a non-linear problem of state estimation (Malebranche, 1988; Korbicz et al., 1988);

Employing random fields analysis (Kazimierczyk, 1989; Sun, 1994);

Consider a spatial domain $\Omega \subset \mathbb{R}^2$ and the PDE

$$\frac{\partial y}{\partial t} = \mathcal{F}\left(x, t, y, \theta\right), \quad x \in \Omega, \ t \in T$$

subject to the appropriate initial and boundary conditions.

Notation:
- $x$ – spatial variable, $t$ – time, $T = [0, t_f]$;
- $y = y(x, t)$ – state vector;
- $\mathcal{F}$ – given operator including spatial derivatives;
- $\theta \in \mathbb{R}^m$ – vector of unknown (constant) parameters.
Consider a domain $\Omega \subset \mathbb{R}^2$ with sufficiently smooth boundary $\Gamma$.

**Advection-diffusion equation**

\[
\frac{\partial}{\partial t} y(x, t) = \nabla \cdot \left[ \kappa(x, \theta) \nabla y(x, t) \right] + \nu(x) \cdot \nabla y(x, t) + f(x, t, \theta)
\]

\[
y(x, 0) = y_0(x), \quad y(x, t) \big|_{\Gamma} = 0
\]

Notation:
- $y(x, t)$ – contaminant concentration at point $x$ and time $t$
- $\kappa(x, \theta)$ – diffusivity coefficient, parameterized by $\theta$
- $f(x, t, \theta)$ – source term
- $\nu(x)$ – wind velocity field
Observation

For pointwise stationary sensors we have

\[ \mathbf{z}(t) = \mathbf{y}_m(t) + \mathbf{\varepsilon}_m(t), \quad t \in T \]

where

\[ \mathbf{y}_m(t) = \text{col}[y(x^1, t), \ldots, y(x^N, t)] \]
\[ \mathbf{\varepsilon}_m(t) = \text{col}[\varepsilon(x^1, t), \ldots, \varepsilon(x^N, t)] \]

\( x^j \in \bar{\Omega}, j = 1, \ldots, N \) – sensor locations, \( \varepsilon(x, t) \) – Gaussian measurement noise such that

\[ \mathbb{E}\{\varepsilon(x, t)\} = 0, \quad \mathbb{E}\{\varepsilon(x, t)\varepsilon(x', t')\} = q(x, x', t)\delta(t - t') \]

\( \delta \) – Dirac delta function.
The least-squares estimates of $\theta$ are the ones which minimize

$$J(\theta) = \frac{1}{2} \int_{T} \|z(t) - \hat{y}_m(t; \theta)\|_Q^{-1}(t)^2 \, dt$$

where $Q(t) = \left[q(x^i, x^j, t)\right]_{i,j=1}^N \in \mathbb{R}^{N \times N}$ is positive-definite,

$$\|a\|_Q^{-1}(t)^2 = a^T Q^{-1}(t) a, \quad \forall a \in \mathbb{R}^N$$

$$\hat{y}_m(t; \theta) = \text{col} [\hat{y}(x^1, t; \theta), \ldots, \hat{y}(x^N, t; \theta)]$$

and $\hat{y}(\cdot, \cdot; \theta)$ stands for the solution to the state equation corresponding to a given value of $\theta$. 
Optimal measurement problem

Formulation

Determine $x^j$, $j = 1, \ldots, N$ so as to maximize the identification accuracy in a sense.

Problem

How to quantify the identification accuracy?
Optimal measurement problem

Formulation

Determine $x^j, j = 1, \ldots, N$ so as to maximize the identification accuracy in a sense.

Problem

How to quantify the identification accuracy?

Answer: Make use of the Cramér-Rao inequality:

$$ \text{cov}(\hat{\theta}) = E\left\{ (\hat{\theta} - \theta)(\hat{\theta} - \theta)^T \right\} \succeq M^{-1} $$

We have $\text{cov}(\hat{\theta}) = M^{-1}$ provided that an estimator is efficient!

But what is $M$?
Towards a criterion

**Spatially independent measurements**

\[
E\{\varepsilon(x^i, t)\varepsilon(x^j, t')\} = \sigma^2 \delta_{ij} \delta(t - t')
\]

where \(\delta_{ij}\) is the Kronecker delta and \(\sigma > 0\) signifies the standard deviation of the measurement noise.

**Average Fisher Information Matrix (FIM)**

\[
\bar{M} = \frac{1}{Nt_f} \sum_{j=1}^{N} \int_{t}^{t_f} g(x^j, t) g^T(x^j, t) \, dt
\]

where \(g(x^j, t) = \left(\frac{\partial y(x^j, t; \theta)}{\partial \theta}\right)^T_{\theta=\theta^0}\) is the vector of sensitivity coefficients, \(\theta^0\) stands for a preliminary estimate of \(\theta\).

\(\bar{M} = M\) up to a constant multiplier.
Calculation of the sensitivity vector $g$

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Finite-difference method</td>
</tr>
<tr>
<td>2</td>
<td>Sensitivity-equation method (direct approach)</td>
</tr>
<tr>
<td>3</td>
<td>Variational method (indirect approach)</td>
</tr>
</tbody>
</table>
Calculation of the sensitivity vector $g$

Example (direct approach)

Consider

$$\frac{\partial y}{\partial t} = \nabla \cdot (\kappa \nabla y) \quad \text{in } \Omega \times T$$

where $\Omega \subset \mathbb{R}^2$ is the spatial domain with boundary $\Gamma$, $T = (0, t_f)$,

$$\kappa(x) = a + b\psi(x), \quad \psi(x) = x_1 + x_2$$

subject to

$$y(x, 0) = 5 \quad \text{in } \Omega$$

$$y(x, t) = 5(1 - t) \quad \text{on } \Gamma \times T$$
Example (ctd’)

The sensitivities to $a$ and $b$ satisfy

\[
\frac{\partial y_a}{\partial t} = \nabla \cdot \nabla y + \nabla \cdot (\kappa \nabla y_a)
\]
\[
\frac{\partial y_b}{\partial t} = \nabla \cdot (\psi \nabla y) + \nabla \cdot (\kappa \nabla y_b)
\]

supplemented with homogeneous initial and Dirichlet boundary conditions. We have to solve them simultaneously with the state equation and to store the solution in memory for its further use in a sensor location algorithm.
Ultimate formulation

Reformulated sensor location problem

Choose \( x^j, j = 1, \ldots, N \) corresponding to an extreme of some scalar measure \( \Psi \) of the FIM, e.g. minimize

\[
\Psi^\gamma(M) = \begin{cases} 
\frac{1}{m} \text{trace}(QM^{-1}Q^T)^\gamma \left[ \frac{1}{m} \text{trace}(QM^{-1}Q^T)^\gamma \right]^{1/\gamma} & \text{if } \det(M) \neq 0 \\
\infty & \text{if } \det(M) = 0
\end{cases}
\]
Most popular choices:

- \( \gamma = 1 \) and \( Q = I_m \) ⇒ **A-optimal design**

\[
J_A(x^1, \ldots, x^N) = \text{trace} M^{-1}(x^1, \ldots, x^N) \rightarrow \min
\]

- \( \gamma \rightarrow \infty \) and \( Q = I_m \) ⇒ **E-optimal design**

\[
J_E(x^1, \ldots, x^N) = \lambda_{\max}(M^{-1}(x^1, \ldots, x^N)) \rightarrow \min
\]

- \( \gamma \rightarrow 0 \) and \( Q = I_m \) ⇒ **D-optimal design**

\[
J_D(x^1, \ldots, x^N) = \det M(x^1, \ldots, x^N) \rightarrow \max
\]

- Sensitivity criterion (may yield very poor solutions!)

\[
J_S(x^1, \ldots, x^N) = \text{trace} M(x^1, \ldots, x^N) \rightarrow \max
\]
Well, there are some drawbacks...

- The problem dimensionality may be high and many local minima interfere with the optimization process.
- When $t_f < \infty$, the estimator is neither unbiased nor characterized by a normal distribution.
- Sensors sometimes tend to cluster.
- The optimal solution usually depends on the parameters to be identified.
Consider the heat equation

\[ \frac{\partial y(x, t)}{\partial t} = \frac{\partial}{\partial x_1} \left( \kappa(x) \frac{\partial y(x, t)}{\partial x_1} \right) + \frac{\partial}{\partial x_2} \left( \kappa(x) \frac{\partial y(x, t)}{\partial x_2} \right) \quad \text{in } \Omega \times T \]

supplemented with

\[ y(x, 0) = 5 \quad \text{in } \Omega \]
\[ y(x, t) = 5(1 - t) \quad \text{on } \partial \Omega \times T \]

where \( \Omega = [0, 1]^2, \ T = (0, 1) \),

\[ \kappa(x) = \theta_1 + \theta_2 x_1 + \theta_3 x_2 \]

\[ \theta_1 = 0.1, \quad \theta_2 = \theta_3 = 0.3 \]
Figure: Locating six sensors through the direct use of non-linear programming techniques: (a) independent measurements, (b) correlated measurements.
Sensor clusterization

Example

Consider

\[ \frac{\partial y(x, t)}{\partial t} = \theta_1 \frac{\partial^2 y(x, t)}{\partial x^2}, \quad x \in (0, \pi), \quad t \in (0, 1) \]

supplemented by

\[
\begin{aligned}
  y(0, t) &= y(\pi, t) = 0, \quad t \in (0, 1) \\
  y(x, 0) &= \theta_2 \sin(x), \quad x \in (0, \pi)
\end{aligned}
\]

Find D-optimal sensor locations for estimation of \( \theta_1 \) and \( \theta_2 \).
Example (ctd’)

Solution The closed-form solution is

\[ y(x, t) = \theta_2 \exp(-\theta_1 t) \sin(x) \]

Assuming \( \sigma = 1 \), on some easy calculations, we get

\[
\det(M(x^1, x^2)) = \frac{\theta_2^2}{16\theta_1^4} \left( -4\theta_1^2 \exp(-2\theta_1) - 2 \exp(-2\theta_1) + \exp(-4\theta_1) + 1 \right) \]

\[
\cdot (2 - \cos^2(x^1) - \cos^2(x^2))^2 \implies x_1^* = x_2^* = \frac{\pi}{2}
\]
Sensor clusterization

Example (ctd’)

Figure 2.2 shows the corresponding surface and contour plots. Clearly, the maximum of the above criterion is attained for \( x_1^\star = x_2^\star = \pi/2 \), but this means that both the sensors should be placed at the same point at the centre of the interval \((0, \pi)\).

In the literature on stationary sensors, a common remedy for such a predicament is to guess a priori as \( e \) to \( f \) \( N' \) possible locations, where \( N' > N \), and then to seek the best set of \( N \) locations from among the \( N' \) possible, so that the problem is then reduced to a combinatorial one [300]. (If the system is solved numerically, the maximum value of \( N' \) is the number of grid points in the domain \( \bar{\Omega} \).)
Example

Consider the heat process

\[
\frac{\partial y(x, t)}{\partial t} = \theta \frac{\partial^2 y(x, t)}{\partial x^2}, \quad x \in (0, \pi), \quad t \in (0, t_f)
\]

complemented with

\[
\begin{cases}
y(0, t) = y(\pi, t) = 0, & t \in (0, t_f) \\
y(x, 0) = \sin(x) + \frac{1}{2} \sin(2x), & x \in (0, \pi)
\end{cases}
\]

Its solution can be easily found in explicit form as

\[
y(x, t) = \exp(-\theta t) \sin(x) + \frac{1}{2} \exp(-4\theta t) \sin(2x)
\]
Dependence of designs on estimated parameters

Example

The FIM (here scalar) is

\[
M(x^1) = \int_0^{t_f} \left( \frac{\partial y(x^1, t; \theta)}{\partial \theta} \right)^2 dt
= \frac{1}{4\theta^3} \left\{ -\sin^2(x^1) \left( 2\theta^2 t_f^2 + 2t_f \theta + 1 \right) \exp(-2t_f \theta) \\
- \frac{32}{125} \cos(x^1) \sin^2(x^1) \left( 10t_f \theta + 2 + 25\theta^2 t_f^2 \right) \exp(-5t_f \theta) \\
- \frac{1}{4} \cos^2(x^1) \sin^2(x^1) \left( 1 + 32\theta^2 t_f^2 + 8t_f \theta \right) \exp(-8t_f \theta) \\
+ \frac{1}{500} \sin^2(x^1) \left( 500 + 256 \cos(x^1) + 125 \cos^2(x^1) \right) \right\}
\]
Dependence of designs on estimated parameters

Example (ctd’)

Figure 2.3: Surface and contour plots for $M(x_1)$ of Example 2.4 as a function of the parameter $\theta$ (the dashed line represents the best measurement points).

It leads to the same solution due to the monotonicity of $\Psi$. Figure 2.3 shows the corresponding graphs after setting $t_f = 1$, from which two important features can be deduced. First of all, there may exist local minima of $\Psi$ which interfere with numerical minimization of the adopted performance criterion. Apart from that, the optimal sensor position which corresponds to the global minimum of $\Psi(M(x_1))$ depends on the true value of $\theta$ (cf. the dashed line joining points $(0,0.87)$ and $(1,1.1)$ on the contour plot).

This dependence on $\theta$ is an unappealing characteristic of nonlinear optimum-experimental design and was most appropriately depicted by Cochran [131]: 'You tell me the value of $\theta$ and I promise to design the best experiment for estimating $\theta$.' This predicament can be partially circumvented by relying on a nominal value of $\theta$, the results of a preliminary experiment or a sequential design which consists of multiple alternation of experimentation and estimation steps. Unfortunately, such strategies are often impractical, because the required experimental time may be too long and the experimental cost may be too high. An alternative is to exploit the so-called robust-design strategies [282,349] which allow us to make optimal solutions independent of the parameters to be identified. The approach (called the average-optimality approach) relies on a probabilistic description of the prior uncertainty in $\theta$, characterized by a prior distribution $\pi_p(\theta)$ (this distribution may have been inferred, e.g., from previous observations collected in similar processes). In the same spirit, the minimax optimality produces the best sensor positions in the worst circumstances. Both the approaches are treated in detail in Chapter 6.
1st option: Adaptive design

Since $\theta_{\text{true}}$ is unknown, a common approach is to use some prior estimate $\theta^0$ of $\theta_{\text{true}}$ instead. But $\theta^0$ may be far from $\theta_{\text{true}}$ and properties of locally optimal designs can be very sensitive to changes in $\theta$.

**Remedy**

A first way out to obtain a robust design procedure is to provide adaptivity through application of a sequential design.
Experimentation and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
Experimentation and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
Experimention and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
Experimentation and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
Experimentation and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
Experimentation and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
Experimentation and estimation steps are alternated. At each stage, a locally optimal sensor location is determined based on the current parameter estimates (nominal parameter values can be assumed as initial guesses for the first stage), measurements are taken at the newly calculated sensor positions, and the data obtained are then analysed and used to update the parameter estimates.
The approach adopted here

Adaptation of the notion of continuous designs (the approach set forth by Rafajłowicz in the 1980s) for a wide class of optimality criteria, as is common in modern optimum experimental design.
Given \( N \) sensors and a finite set \( X = \{ x^1, \ldots, x^\ell \} \subset \Omega \cup \partial\Omega \) of points at which they can be placed, consider their fixed configuration.
Given \( N \) sensors and a finite set \( X = \{ \mathbf{x}^1, \ldots, \mathbf{x}^\ell \} \subset \Omega \cup \partial \Omega \) of points at which they can be placed, consider their fixed configuration.
Output equation

For pointwise stationary sensors we have

\[ z_{ij}(t) = y(x^i, t; \theta) + \varepsilon_{ij}(t), \quad t \in T \]

for \( i = 1, \ldots, \ell \) and \( j = 1, \ldots, r_i \), where

\[ \sum_{i=1}^{\ell} r_i = N \]

\( \varepsilon_{ij}(\cdot) \) – white Gaussian measurement noise.
Output equation

For pointwise stationary sensors we have

\[ z_{ij}(t) = y(x^i, t; \theta) + \varepsilon_{ij}(t), \quad t \in T \]

for \( i = 1, \ldots, \ell \) and \( j = 1, \ldots, r_i \), where

\[ \sum_{i=1}^{\ell} r_i = N \]

\( \varepsilon_{ij}(\cdot) \) – white Gaussian measurement noise.

Average Fisher Information Matrix (FIM)

Up to a constant multiplier, we have

\[ M = \frac{1}{Nt_f} \sum_{i=1}^{\ell} r_i \int_0^{t_f} g(x^i, t) g^T(x^i, t) \, dt \]
Minimize some measure $\Psi$ of the FIM, e.g.,

$$\Psi(M) = \ln \det(M^{-1}) \quad \text{or} \quad \Psi(M) = \text{trace}(M^{-1})$$

by finding a solution (called design) of the form

$$\xi = \left\{ x^1, \ldots, x^\ell, p_1, \ldots, p_\ell \right\}$$

where $p_i = r_i/N$, i.e., the weight $p_i$ defining the proportion of sensors placed at $x^i$, $i = 1, \ldots, N$. 
Problem of weight optimization

Minimize some measure $\Psi$ of the FIM, e.g.,

$$\Psi(M) = \ln \det(M^{-1}) \quad \text{or} \quad \Psi(M) = \text{trace}(M^{-1})$$

by finding a solution (called design) of the form

$$\xi = \left\{ x^1, \ldots, x^\ell \right\}$$

$$\left\{ p_1, \ldots, p_\ell \right\}$$

where $p_i = r_i / N$, i.e., the weight $p_i$ defining the proportion of sensors placed at $x^i$, $i = 1, \ldots, N$. 
The problem still has a discrete nature, as the sought weights $p_i$ are integer multiples of $1/N$ which sum up to unity. Consequently, calculus techniques cannot be exploited in the solution.
The problem still has a discrete nature, as the sought weights $p_i$ are integer multiples of $1/N$ which sum up to unity. Consequently, calculus techniques cannot be exploited in the solution.

Thus we extend the definition of the design: when $N$ is large, the feasible $p_i$’s can be considered as any real numbers in the interval $[0, 1]$ which sum up to unity, and not necessarily integer multiples of $1/N$, i.e., elements of the canonical simplex.
Ultimate formulation

Find \( \xi^* = \{ x_1^*, ..., x_\ell^* \} \) such that

\[
p^* = \arg \min_{p \in \mathcal{S}} \Psi[M(\xi)]
\]

where \( \mathcal{S} \) is the canonical simplex, subject to

\[
M(\xi) = \sum_{i=1}^{\ell} p_i \Upsilon(x^i)
\]

Here \( \Upsilon(x) = \frac{1}{t_f} \int_0^{t_f} g(x, t) g^T(x, t) \, dt \) is the fixed contribution of the point \( x \) to the FIM.
Ultimate formulation

Find \( \xi^* = \left\{ x_1, ..., x_\ell \right\} \) such that

\[
p^* = \arg \min_{p \in S} \psi[M(\xi)]
\]

where \( S \) is the canonical simplex, subject to

\[
M(\xi) = \sum_{i=1}^{\ell} p_i \Upsilon(x^i)
\]

Here \( \Upsilon(x) = \frac{1}{t_f} \int_0^{t_f} g(x, t)g^T(x, t) \, dt \) is the fixed contribution of the point \( x \) to the FIM.
Ultimate formulation

Find \( \xi^* = \{ x_1^*, \ldots, x_\ell^* \} \) such that

\[
p^* = \arg \min_{p \in S} \Psi[M(\xi)]
\]

where \( S \) is the canonical simplex, subject to

\[
M(\xi) = \sum_{i=1}^\ell p_i \Upsilon(x^i)
\]

Here \( \Upsilon(x) = \frac{1}{t_f} \int_0^{t_f} g(x, t)g^T(x, t) \, dt \) is the fixed contribution of the point \( x \) to the FIM.
Find $\xi^* = \left\{ x_1^*, \ldots, x_\ell^* \right\}$ such that

$$p^* = \arg\min_{p \in \mathbb{S}} \Psi[M(\xi)]$$

where $\mathbb{S}$ is the canonical simplex, subject to

$$M(\xi) = \sum_{i=1}^{\ell} p_i \Upsilon(x^i)$$

Here $\Upsilon(x) = \frac{1}{t_f} \int_0^{t_f} g(x, t)g^T(x, t) \, dt$ is the fixed contribution of the point $x$ to the FIM.
Define

\[ \hat{\psi}(\xi) = \left. \frac{\partial \psi(M)}{\partial M} \right|_{M = M(\xi)} = \left. \left[ \frac{\partial \psi(M)}{\partial M_{ij}} \right] \right|_{M = M(\xi)} \]

Note that we have

\[ \frac{\partial}{\partial M} \log \det(M^{-1}) = -(M^{-1})^T \]
\[ \frac{\partial}{\partial M} \text{trace}(M^{-1}) = -(M^{-2})^T \]
\[ \frac{\partial}{\partial M} \left( - \text{trace}(M) \right) = -I \]
Differentiability of the design criterion

We introduce crucial functions

\[
    c(\xi) = - \text{trace} \left[ \Phi(\xi) M(\xi) \right]
\]

\[
    \phi(x, \xi) = - \text{trace} \left[ \Phi(\xi) \Upsilon(x) \right]
\]

<table>
<thead>
<tr>
<th>\psi[M(\xi)]</th>
<th>\phi(x, \xi)</th>
<th>\phi(x, \xi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ln det(M^{-1}(\xi))</td>
<td>\frac{1}{t_f} \int_0^{t_f} g^T(x, t) M^{-1}(\xi) g(x, t) , dt</td>
<td>\frac{1}{t_f} \int_0^{t_f} g^T(x, t) M^{-1}(\xi) g(x, t) , dt</td>
</tr>
<tr>
<td>trace(M^{-1}(\xi))</td>
<td>trace(M^{-1}(\xi))</td>
<td>trace(M^{-1}(\xi))</td>
</tr>
<tr>
<td>- trace(M(\xi))</td>
<td>trace(M(\xi))</td>
<td>trace(M(\xi))</td>
</tr>
</tbody>
</table>
Theorem (Existence of solution)

An optimal design exists comprising no more than \( m(m + 1)/2 \) support points. Moreover, the set of optimal designs is convex.
Characterizations of solutions

Theorem (Existence of solution)

An optimal design exists comprising no more than $m(m + 1)/2$ support points. Moreover, the set of optimal designs is convex.

Theorem (Equivalence theorem)

The following characterizations of an optimal design $\xi^*$ are equivalent:

1. the design $\xi^*$ minimizes $\Psi[M(\xi)]$,
2. the design $\xi^*$ minimizes $\max_{x \in X} \phi(x, \xi) - c(\xi)$,
3. $\max_{x \in X} \phi(x, \xi^*) = c(\xi^*)$.

All the designs satisfying 1–3 and their convex combinations have the same $M(\xi^*)$. 
For any designs

$$\xi_1 = \left\{ \begin{array}{c} x_1^{(1)}, \ldots, x_{\ell_1}^{(1)} \\ p_1^{(1)}, \ldots, p_{\ell_1}^{(1)} \end{array} \right\}, \quad \xi_2 = \left\{ \begin{array}{c} x_1^{(2)}, \ldots, x_{\ell_1}^{(2)} \\ p_1^{(2)}, \ldots, p_{\ell_1}^{(2)} \end{array} \right\}$$

we define

$$\xi = (1 - \lambda)\xi_1 + \lambda\xi_2$$

$$= \left\{ \begin{array}{c} x_1^{(1)}, \ldots, x_{\ell_1}^{(1)} \\ (1 - \lambda)p_1^{(1)} + \lambda p_1^{(2)}, \ldots, (1 - \lambda)p_{\ell_1}^{(1)} + \lambda p_{\ell_1}^{(2)} \end{array} \right\}$$
For $\Psi[M(\xi)] = \text{trace}(M^{-1}(\xi))$, we have to find $p$ and the additional variable $q \in \mathbb{R}^m$ to minimize

$$J(p, q) = 1^Tq$$

subject to $1^Tp = 1$, $p \succeq 0$ and the Linear Matrix Inequalities (LMIs)

$$\begin{bmatrix} M(\xi) & e_j \\ e_j^T & q_j \end{bmatrix} \succeq 0, \quad j = 1, \ldots, m.$$

where $e_j$ is $j$-th coordinate versor.
For $\psi[M(\xi)] = \lambda_{\text{max}}[M^{-1}(\xi)]$ we have to find $p$ and $q \in \mathbb{R}$ to minimize

$$\tilde{J}(p, q) = -q$$

subject to

$$1^T p = 1, \quad p \succeq 0$$

and the LMI $M(\xi) \succeq qI$. 

Note that $\lambda_{\text{max}}[M^{-1}(\xi)]$ is non-differentiable!
Numerical approach: E-optimality

For $\Psi[M(\xi)] = \lambda_{\text{max}}[M^{-1}(\xi)]$ we have to find $p$ and $q \in \mathbb{R}$ to minimize

$$\tilde{J}(p, q) = -q$$

subject to

$$1^T p = 1, \quad p \succeq 0$$

and the LMI $M(\xi) \succeq qI$.

Note that $\lambda_{\text{max}}[M^{-1}(\xi)]$ is non-differentiable!
Numerical approach: Torsney’s algorithm for D-optimality

For $\Psi[M(\xi)] = \ln \det[M^{-1}(\xi)]$ apply

1. Guess $p_i^{(0)}$, $i = 1, \ldots, \ell$. Set $k = 0$.

2. If $\phi(x^i, \xi^{(k)}) = \text{trace}(\Psi(x^i)M^{-1}(\xi^{(k)})) \leq m$, $i = 1, \ldots, \ell$, then STOP.

3. Evaluate

$$p_i^{(k+1)} = p_i^{(k)} \frac{\phi(x^i, \xi^{(k)})}{m}, \quad i = 1, \ldots, \ell.$$ 

Form $\xi^{(k+1)}$, increment $k$ and go to Step 2.
Numerical approach: Torsney’s algorithm for D-optimality

For $\Psi[M(\xi)] = \ln \det[M^{-1}(\xi)]$ apply

1. Guess $p_i^{(0)}$, $i = 1, \ldots, \ell$. Set $k = 0$.
2. If $\phi(x^i, \xi^{(k)}) = \text{trace}(\Upsilon(x^i)M^{-1}(\xi^{(k)})) \leq m$, $i = 1, \ldots, \ell$, then STOP.
3. Evaluate
   
   $$p_i^{(k+1)} = p_i^{(k)} \frac{\phi(x^i, \xi^{(k)})}{m}, \quad i = 1, \ldots, \ell.$$ 

Form $\xi^{(k+1)}$, increment $k$ and go to Step 2.
Numerical approach: Torsney’s algorithm for D-optimality

For \( \Psi[M(\xi)] = \ln \det[M^{-1}(\xi)] \) apply

1. Guess \( p_i^{(0)}, i = 1, \ldots, \ell \). Set \( k = 0 \).
2. If \( \phi(x^i, \xi^{(k)}) = \text{trace}(\Upsilon(x^i)M^{-1}(\xi^{(k)})) \leq m, i = 1, \ldots, \ell \), then STOP.
3. Evaluate

\[
p_i^{(k+1)} = p_i^{(k)} \frac{\phi(x^i, \xi^{(k)})}{m}, \quad i = 1, \ldots, \ell.
\]

Form \( \xi^{(k+1)} \), increment \( k \) and go to Step 2.
The idea is reminiscent of the EM algorithm used for maximum likelihood estimation.

**Theorem**

Assume that $\{\xi^{(k)}\}$ is a sequence of iterates constructed by the outlined algorithm. Then the sequence $\{\det(M(\xi^{(k)}))\}$ is nondecreasing and

$$
\lim_{k \to \infty} \det(M(\xi^{(k)})) = \max_{\xi} \det(M(\xi))
$$
Consider the nonlinear system

\[
\frac{\partial y}{\partial t} = a \frac{\partial^2 y}{\partial x^2} - b y^3, \quad x \in (0, 1), \quad t \in (0, t_f)
\]

subject to the conditions

\[
y(x, 0) = c \psi(x), \quad x \in (0, 1)
\]
\[
y(0, t) = y(1, t) = 0, \quad t \in (0, t_f)
\]

where \( t_f = 0.8 \), \( \psi(x) = \sin(\pi x) + \sin(3\pi x) \).
Determine a D-optimum design of the form
\[ \xi^* = \left\{ x^1, \ldots, x^\ell \right\} \]
for recovering \( \theta = (a, b, c) \), where the support points are
\[ x^i = 0.05(i - 1), \quad i = 1, \ldots, \ell = 21 \]
Apply the multiplicative algorithm just mentioned. (MATLAB’s routine pdepe comes here in handy.)
Results

Figure: Optimal weights
Results

Figure: Derivative function $\phi(x, \xi^*)$
Results

Figure: Monotonic increase in $\det[M(\xi^{(k)})]$
Consider a metallic core with a rectangular crack inside.

The left and right boundaries $\Gamma_1$ and $\Gamma_2$ are subjected to constant voltages forcing the flow of a low density steady current.
Computer-assisted impedance tomography

Electrical voltage distribution $y$ is described by the Laplace equation

$$-\text{div}(\gamma(x)\nabla y(x)) = 0, \quad x \in \Omega,$$

with boundary conditions

$$\begin{cases} 
  y(x) = 10 & x \in \Gamma_1, \\
  y(x) = -10 & x \in \Gamma_2, \\
  \frac{\partial y(x)}{\partial n} = 0 & x \in \Gamma_i, \ i = 3, 4, 5.
\end{cases}$$

Conductivity coefficient

$$\gamma(x) = 0.1 + 0.2 \exp \left(-\theta_1(x_1 - \theta_2)^2 - \theta_3(x_2 - \theta_4)^2\right)$$

$$\theta^0 = (30.0, 0.3, 80.0, 0.7)$$
Results

D-optimal

\begin{align*}
&\Gamma_1 \quad 0.25 \\
&\Gamma_2 \quad 0.23 \\
&\Gamma_3 \quad 0.03 \\
&\Gamma_4 \quad 0.24 \\
&\Gamma_5 \quad 0.25 \\
\end{align*}

A-optimal

\begin{align*}
&\Gamma_1 \quad 0.05 \\
&\Gamma_2 \quad 0.23 \\
&\Gamma_3 \quad 0.11 \\
&\Gamma_4 \quad 0.24 \\
&\Gamma_5 \quad 0.37 \\
\end{align*}
The approach is useful as
The approach is useful as

- It indicates spatial regions which provide the most important information about the parameters, while quantifying their influence.
The approach is useful as:

- It indicates spatial regions which provide the most important information about the parameters, while quantifying their influence.
- Instead of several sensors placed at a point, we can use a more precise measurement device at that point.
The approach is useful as

- It indicates spatial regions which provide the most important information about the parameters, while quantifying their influence.
- Instead of several sensors placed at a point, we can use a more precise measurement device at that point.
- Design weights can be interpreted as measurement frequencies.
The approach is useful as

- It indicates spatial regions which provide the most important information about the parameters, while quantifying their influence.
- Instead of several sensors placed at a point, we can use a more precise measurement device at that point.
- Design weights can be interpreted as measurement frequencies.
- With a little turning up, the results can be employed when attacking the setting with purely binary weights ($r_i = 0$ or $r_i = 1$), see Part 2.
Question

So far, we have focused on optimizing weights assigned to spatial points fixed *a priori*. Is it possible to simultaneously optimize these spatial points and weights, possibly with selecting their ‘optimal’ number?
Passage from discrete to continuous designs

\[ \xi_N = \left\{ \frac{x^1}{p_1}, \ldots, \frac{x^\ell}{p_\ell} \right\} \]

\[ \xi \twoheadrightarrow M(\xi_N) = \sum_{i=1}^{\ell} p_i \Upsilon(x^i) \]

\[ \xi \rightarrow M(\xi) = \int_X \Upsilon(x) \xi(dx) \]

Notation: \( X \) is a compact set of feasible sensor locations,

\[ \Upsilon(x) = \frac{1}{t_f} \int_0^{t_f} g(x^i, t)g^T(x^i, t) \, dt \]
Introduction to optimum design
OED and optimal sensor location

Passage from discrete to continuous designs

$$\xi_N = \left\{ x^1, \ldots, x^\ell \right\} \quad \begin{array}{c} p_1, \ldots, p_\ell \\ \downarrow p_i = r_i / N, \sum_{i=1}^{\ell} p_i = 1 \end{array} \rightarrow M(\xi_N) = \sum_{i=1}^{\ell} p_i \Upsilon(x^i)$$

$$\xi - \text{any probability measure on } X$$
$$\int_X \xi(dx) = 1$$

$$\xi(x) = \frac{1}{t_f} \int_0^{t_f} g(x^i, t) g^T(x^i, t) dt$$

Notation: $X$ is a compact set of feasible sensor locations,
Let $\Xi(X)$ denote the set of all probability measures on $X$.

**Lemma (Symmetry and nonnegativity)**

For any $\xi \in \Xi(X)$ the information matrix $M(\xi)$ is symmetric and non-negative definite.

**Lemma (Set of all FIMs)**

$\mathcal{M}(X) = \{ M(\xi) : \xi \in \Xi(X) \}$ is compact and convex.

**Lemma (Designs as probability mass functions)**

For any $M_0 \in \mathcal{M}(X)$ there always exists a purely discrete design $\xi$ with no more than $m(m + 1)/2 + 1$ support points such that $M(\xi) = M_0$. 
Consequently, we may think of designs as discrete probability mass functions of the form

$$\xi = \left\{ x^1, x^2, \ldots, x^\ell; \quad p_1, p_2, \ldots, p_\ell; \quad \sum_{i=1}^{\ell} p_i = 1 \right\}$$

where the number of support points $\ell$ is not fixed and constitutes an additional parameter to be determined, subject to the condition $\ell \leq m(m+1)/2 + 1$. 
Our main assumption is that

\[
\frac{\partial \psi}{\partial \lambda} \bigg|_{\lambda=0^+} = \int_X \psi(x, \xi) \bar{\xi}(dx)
\]

But it is not particularly restrictive, since we have

\[
\psi(x, \xi) = c(\xi) - \phi(x, \xi)
\]

where

\[
c(\xi) = - \text{trace} \left[ \dot{\Psi}(\xi) M(\xi) \right], \quad \phi(x, \xi) = - \text{trace} \left[ \dot{\Psi}(\xi) \Upsilon(x) \right]
\]

\[
\dot{\Psi}(\xi) = \left. \frac{\partial \psi(M)}{\partial M} \right|_{M=M(\xi)}
\]
Differentiability of the design criterion

Recall that

\[
\begin{align*}
\psi[M(\xi)] &= \ln \det(M^{-1}(\xi)) \\
\phi(x, \xi) &= \frac{1}{t_f} \int_0^{t_f} \mathbf{g}^T(x, t) M^{-1}(\xi) \mathbf{g}(x, t) \, dt \\
\frac{1}{t_f} \int_0^{t_f} \mathbf{g}^T(x, t) M^{-2}(\xi) \mathbf{g}(x, t) \, dt \\
- \text{trace}(M(\xi)) &= \frac{1}{t_f} \int_0^{t_f} \mathbf{g}^T(x, t) \mathbf{g}(x, t) \, dt \\
\end{align*}
\]

<table>
<thead>
<tr>
<th>$\psi[M(\xi)]$</th>
<th>$\phi(x, \xi)$</th>
<th>$c(\xi)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\ln \det(M^{-1}(\xi))$</td>
<td>$\frac{1}{t_f} \int_0^{t_f} \mathbf{g}^T(x, t) M^{-1}(\xi) \mathbf{g}(x, t) , dt$</td>
<td>$m$</td>
</tr>
<tr>
<td>$\text{trace}(M^{-1}(\xi))$</td>
<td>$\frac{1}{t_f} \int_0^{t_f} \mathbf{g}^T(x, t) M^{-2}(\xi) \mathbf{g}(x, t) , dt$</td>
<td>$\text{trace}(M^{-1}(\xi))$</td>
</tr>
<tr>
<td>$- \text{trace}(M(\xi))$</td>
<td>$\frac{1}{t_f} \int_0^{t_f} \mathbf{g}^T(x, t) \mathbf{g}(x, t) , dt$</td>
<td>$\text{trace}(M(\xi))$</td>
</tr>
</tbody>
</table>
Theorem

Under some conditions (not particularly restrictive anyway; cf. the Monograph) we have the following results:

1. An optimal design exists comprising not more than \( m(m + 1)/2 \) points (i.e. one less than predicted previously).
2. The set of optimal designs is convex.
Assume that the support points of the designs $\xi_1$ and $\xi_2$ coincide (this situation is easy to achieve, since if a support point is included in only one design, we can formally add it to the other design and assign it the zero weight). Thus we have

$$\xi_1 = \left\{ \begin{array}{c} x_1^{(1)}, \ldots, x_\ell^{(1)} \\ p_1^{(1)}, \ldots, p_\ell^{(1)} \end{array} \right\}, \quad \xi_2 = \left\{ \begin{array}{c} x_1^{(2)}, \ldots, x_\ell^{(2)} \\ p_1^{(2)}, \ldots, p_\ell^{(2)} \end{array} \right\}$$

and, in consequence, we define

$$\xi = (1 - \lambda)\xi_1 + \lambda\xi_2 = \left\{ \begin{array}{c} x_1^{(1)}, \ldots, x_\ell^{(1)} \\ (1 - \lambda)p_1^{(1)} + \lambda p_1^{(2)}, \ldots, (1 - \lambda)p_\ell^{(1)} + \lambda p_\ell^{(2)} \end{array} \right\}.$$
Main result: Equivalence Theorem

**Theorem**

The following characterizations of an optimal design $\xi^*$ are equivalent in the sense that each implies the other two:

1. the design $\xi^*$ minimizes $\Psi[M(\xi)]$,
2. the design $\xi^*$ minimizes $\max_{x \in X} \phi(x, \xi) - c(\xi)$, and
3. $\max_{x \in X} \phi(x, \xi^*) = c(\xi^*)$.

All the designs satisfying 1–3 and their convex combinations have the same information matrix $M(\xi^*)$. 
Illustration of Equivalence Theorem

Figure: Derivative function for a D-optimal design (solid) and an exemplary nonoptimal design (dashed).
Illustration of Equivalence Theorem

Example

Consider

$$\frac{\partial y(x, t)}{\partial t} = \alpha \frac{\partial^2 y(x, t)}{\partial x^2} + \beta y(x, t), \quad x \in \Omega = (0, 1), \quad t \in T = (0, 1)$$

subject to

$$y(x, 0) = \sin (\pi x) + \sin (2 \pi x) \quad \text{in } \Omega$$
$$y(0, t) = y(1, t) = 0 \quad \text{on } T$$

The purpose is to determine a D-optimum design for $\theta = (\alpha, \beta)$. 
Illustration of Equivalence Theorem


d\textbf{Example (ctd')} \\

Exploiting the closed-form solution

\[ y(x, t) = e^{(\beta - \alpha \pi^2)t} \sin(\pi x) + e^{(\beta - 4\alpha \pi^2)t} \sin(2\pi x) \]

we get

\[ \xi^* = \begin{cases} \frac{1}{\pi} \arctan(\sqrt{2}), & \frac{1}{2}, \frac{1}{2} \\ 1 - \frac{1}{\pi} \arctan(\sqrt{2}), & 1 - \frac{1}{\pi} \arctan(\sqrt{2}) \end{cases} \]
Illustration of Equivalence Theorem

Example (ctd’)

Figure: Optimal derivative function (solid)
Introduction to optimum design

OED and optimal sensor location

Problem and its specificities

Adaptation of optimum design methodology

Towards numerical algorithms

Step 1. Guess a discrete non-degenerate starting design measure $\xi^{(0)}$ (we must have $\det(\mathbf{M}(\xi^{(0)})) \neq 0$). Choose some positive tolerance $\eta \ll 1$. Set $k = 0$.

Step 2. Determine $x_0^{(k)} = \arg \max_{x \in \mathcal{X}} \phi(x, \xi_k)$. If

$$\phi(x_0^{(k)}, \xi^{(k)}) < c(\xi^{(k)}) + \eta,$$

then STOP.

Step 3. For an appropriate value of $0 < \lambda_k < 1$, set

$$\xi^{(k+1)} = (1 - \lambda_k)\xi^{(k)} + \lambda_k \delta_{x_0^{(k)}}$$

where $\delta_x$ stands for the one-point (Dirac) design measure $\{x \}$, increment $k$ by one and go to Step 2.

Dariusz Uciński

Optimal sensor location for distributed parameter system identification (Part 1)
Example

Consider the heat equation

\[
\frac{\partial y(x, t)}{\partial t} = \frac{\partial}{\partial x_1} \left( \kappa(x) \frac{\partial y(x, t)}{\partial x_1} \right) + \frac{\partial}{\partial x_2} \left( \kappa(x) \frac{\partial y(x, t)}{\partial x_2} \right) \quad \text{in } \Omega \times T
\]

supplemented with

\[
y(x, 0) = 5 \quad \text{in } \Omega
\]
\[
y(x, t) = 5(1 - t) \quad \text{on } \partial \Omega \times T
\]

where \( \Omega = [0, 1]^2 \), \( T = (0, 1) \),

\[
\kappa(x) = \theta_1 + \theta_2 x_1 + \theta_3 x_2
\]
\[
\theta_1 = 0.1, \quad \theta_2 = \theta_3 = 0.3
\]
Results

Figure: Locating six sensors through the direct use of non-linear programming techniques: (a) independent measurements, (b) correlated measurements.
Figure: Support points of an optimal continuous design (equal weights) (left) and the result of a clusterization-free design for $N = 97$ (right).
Conclusions

The following is a concise summary of the contributions provided by this work to the state-of-the-art in optimal sensor location for parameter estimation in DPS’s:

1. Dramatically reduces the problem dimensionality.
2. Provides characterizations of continuous designs for stationary sensors and a wide class of optimality criteria.
3. Clarifies how to adapt well-known algorithms of optimum experimental design for finding numerical approximations to the solutions.