5. Design space exploration and synthesis

Design space exploration - an interesting history

- Sir Ronald A. Fisher pioneered the concept of "design of experiments" in 1920s in agriculture which resulted in Response Surface Modelling (RSM).
  - Now used in many areas of science and engineering

Limited number of experiments (or simulations!)
Polynomial model (Response Surface)

FAST DESIGN SPACE EXPLORATION!

This process is carried out on several levels of abstraction

Design Space Exploration and Hardware Synthesis

Design Space Exploration
This process is carried out on several levels of abstraction.

Performance analysis is difficult

- Simulation is difficult due to complexity
- Several levels of abstraction
  - Low-fidelity (less accurate) and high-fidelity (more accurate) models
- Behaviour is complex, data dependent
- Constraints
  - Limited resources
  - Limited time to evaluate performance accurately
- Multiple applications
  - Resources even more limited
  - Mapping/scheduling/arbitration very hard

Design space exploration response surface models (RSM)

1. Let the designer select a best design by exploring the design space quickly.
2. RSM approach only needs a limited number of expensive simulations to build the response surface model.
3. RSM uses low-order polynomials, hence it is fast.
4. Need to build a separate RSM for each performance metric — but one set of simulations is sufficient.

Limited number of experiments (or simulations!)  Polynomial model (Response Surface)
**RSM concept will be explained using an example**

- Challenge: find an optimal design of a complex energy-harvester-powered sensor node
  - High-speed vibration microgenerator: small-simulation time step (0.1ms)
  - Low-speed storage: supercapacitor can take tens of hours to charge

---

**Scenario 1: tuning by 1Hz**

Recovery time/duty cycle - 600 seconds

Output power from microgenerator: Supercapacitor voltage:

---

**Scenario 2: tuning by 14Hz**

Maximum tuning range (64~78Hz)

Recovery time/duty cycle - approx. 2 hours

Supercapacitor voltage:

---

**Simulation complexity**

Comparison of CPU times

Two orders of magnitude acceleration was needed to make RSM approach feasible

<table>
<thead>
<tr>
<th></th>
<th>Existing technique</th>
<th>New accelerated technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDL</td>
<td>VHDL-AMS</td>
<td>SystemC-A</td>
</tr>
<tr>
<td>Integration method</td>
<td>Newton-Raphson based</td>
<td>Newton-Raphson based</td>
</tr>
<tr>
<td>CPU time for Scenario 1</td>
<td>2185 sec</td>
<td>2386</td>
</tr>
</tbody>
</table>
| CPU time for Scenario 2 | 7 hours            | 8 hours                   | 228 sec
Component | Type | Make
--- | --- | ---
Microcontroller | PIC16F884 | Microchip
Accelerometer | LIS3L06AL | STMicroelectronics
Linear actuator | 21000 Series Size 8 stepper motor | Haydon
Radio transceiver | eZ430-RF2500 | Texas Instruments

### Response surface model (1)
- To relate the simulation result (the response) with a number of system parameters, a system function (the RSM) can be approximated as:
\[
y = \hat{y}(a_1, a_2, \ldots, a_k) + \epsilon
\]
- Coded variables with zero means are required to build RSM:
\[
x = \frac{a - [a_{\max} + a_{\min}]/2}{[a_{\max} + a_{\min}]/2}
\]
- A low order polynomial equation is often used as system function:
\[
\hat{y} = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_i x_i^2 + \sum_{i<j} \beta_{ij} x_i x_j
\]

### Response surface model (2)
- Low order polynomial equation:
\[
\hat{y} = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_i x_i^2 + \sum_{i<j} \beta_{ij} x_i x_j
\]
- To characterise the RSM is to find coefficients $\beta$ that give the minimum sum of squared residuals (SSE), i.e. minimise the approximation error:
\[
SSE = \sum_{i=1}^{n} \epsilon^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2
\]
- A number of simulations with different parameter values within the design space are carried out to calculate coefficients $\beta$ using least squares method (LSM):
\[
\frac{\partial SSE}{\partial \beta} = 0
\]
- Optimum design (matrix $X$ that maximises the polynomial) can then be found quickly.
- The best design is that which maximises the polynomial.

- Supply voltage is $2.8V$, total energy of 116 $\mu$J during $2.75$ ms of transmission
- Power and energy consumption can be calculated.
Design points choice for simulations - D-optimal design

- How to choose the predefined design points to cover the design space efficiently is the essence of the Design Of Experiment (DOE) methodology:
  \[ \hat{y} = X\beta \]  
  (matrix form of the polynomial, \( X \) = information matrix)
- Different types of DOE method: full factorial, central composite design (CCD), Box Behnken design (BBD) and computer generated designs such as D-optimal design
- D-optimal design is chosen because it requires minimum number of simulation runs
- D-optimal design is obtained by maximising the determinant of information matrix:
  \[ \max(X'X) \]

RSM model of the sensor node system

- 3 system configuration parameters are chosen for optimisation

<table>
<thead>
<tr>
<th>Description</th>
<th>Value range</th>
<th>Coded variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microcontroller clock frequency(Hz)</td>
<td>125k ~ 8M</td>
<td>( x_1 )</td>
</tr>
<tr>
<td>Watchdog timer wakeup time(sec)</td>
<td>160 ~ 600</td>
<td>( x_2 )</td>
</tr>
<tr>
<td>Transmission time interval(sec)</td>
<td>0.005 ~10</td>
<td>( x_3 )</td>
</tr>
</tbody>
</table>

- Test scenario: vibration level 60mg, input frequency changed by 5Hz every 25 minutes
- System performance: total no of transmissions made during 1 hour (related to harvested energy)
- 27 simulations were carried out using accelerated simulation technique developed before
- RSM model (full quadratic equation):
  \[ \hat{y}(x_1, x_2, x_3) = 469.167 - 108.833 x_1 - 18.833 x_2 - 209.5 x_3 \\
  + 71.833 x_1^2 + 90.5 x_2^2 - 39.0 x_3^2 - 32.333 x_1 x_2 - 71.333 x_1 x_3 + 43.333 x_2 x_3 \]

Optimisation within RSM

- Two algorithms from MATLAB toolbox are used for optimisation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Original design</th>
<th>Simulated annealing</th>
<th>Genetic algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microcontroller clock frequency(Hz)</td>
<td>4M</td>
<td>8M</td>
<td>125k</td>
</tr>
<tr>
<td>Watchdog timer wakeup time(sec)</td>
<td>320</td>
<td>60</td>
<td>600</td>
</tr>
<tr>
<td>Transmission time interval(sec)</td>
<td>5</td>
<td>0.005</td>
<td>3.065</td>
</tr>
<tr>
<td>Number of transmissions</td>
<td>405</td>
<td>899</td>
<td>894</td>
</tr>
</tbody>
</table>

- It is possible to investigate the effect of each parameter on system performance using RSM

Demonstrator tool – 6 RSMs

- Performance estimator of wireless sensor powered by kinetic energy harvester
- User parameters and 6 performance indicators (6 RSMs were built)
RSM model of the sensor node system

- 11 continuous parameters and 3 circuit topologies – treated as a discrete parameter
- 6 system performance indicators
- Simulation scenario: 10 hours continuous operation

<table>
<thead>
<tr>
<th>Supercapacitor voltage</th>
<th>Node behaviour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Below 2.0V</td>
<td>Wake up every 1 minute, no transmission</td>
</tr>
<tr>
<td>Between 2.0 and Vth</td>
<td>Wake up and transmit every 1 minute</td>
</tr>
<tr>
<td>Between Vth and 3.5V</td>
<td>Wake up and transmit every 1 second</td>
</tr>
<tr>
<td>Above 3.5V</td>
<td>Wake up and transmit every 3ms, preventing from over-charge</td>
</tr>
</tbody>
</table>

- For each circuit topology with \( n \) continuous parameters, the number of simulations \( (N) \) required by the D-optimal choice is given by:

\[
N = 1 \text{(constant item)} + n \text{(linear items)} + n^2 \text{(squared items)} + C^2_n \text{(interaction items)}
\]

- Total of 200+ simulations needed to build the RSM

Simulations were done on a supercomputer

- Southampton’s Iridis 3 supercomputer
  - 1008 computing nodes each with 12 processing cores
  - Ranked 74 in the world (http://cmg.soton.ac.uk/iridis)
- Message Passing Interface (MPI) library for programming
  - OpenMPI: Open Source High Performance Computing

GUI was created for design space exploration

- User can set the design parameters and the resulting system performance will update instantly
  - Initial GUI, simplified diagram, energy harvester as black box

GUI was created for design space exploration (2)

- Expanded GUI, more detail and more controls for the user
Response surface models in high-level synthesis

- A new concept has been investigated in recent years:
  - Supervised high-level synthesis (SHLS)
- In SHLS, exploration strategies play the role of a supervisor for tuning an HLS engine.
- SHLS tools include not only architectural alternatives but also compiler transformations – high complexity problems
- Recent example published in 2015: a new exploration approach, called spectral-aware Pareto iterative refinement, that exploits response surface models (RSMs) and spectral analysis for predicting the quality of the design points without resorting to costly architectural synthesis procedures: